PostgreSQL 17beta3##

PostgreSQL###########

PostgreSQL 17beta3##

PostgreSQL###########
© 1996–2024 The PostgreSQL Global Development Group, ######PostgreSQL####

####

Copyright © 1996–2024 PostgreSQL#PostgreSQL###################

Copyright © 1994–1995 Postgres95######################

##PostgreSQL####(Japan PostgreSQL User Group)#############

Permission to use, copy, modify, and distribute this software and its documentation for any purpose, without fee, and without a written agreement is hereby granted, provided that the above copyright notice and this paragraph and the following two paragraphs appear in all copies.

[####################] ########### ######2########################### ################################### ##### #############

IN NO EVENT SHALL THE UNIVERSITY OF CALIFORNIA BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, INCLUDING LOST PROFITS, ARISING OUT OF THE USE OF THIS SOFTWARE AND ITS DOCUMENTATION, EVEN IF THE UNIVERSITY OF CALIFORNIA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

################################ ############################ ################################ ############### ########

THE UNIVERSITY OF CALIFORNIA SPECIFICALLY DISCLAIMS ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE SOFTWARE PROVIDED HEREUNDER IS ON AN #AS-IS# BASIS, AND THE UNIVERSITY OF CALIFORNIA HAS NO OBLIGATIONS TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

############################### ############################### ############################ ################# ############## ########

##

..... xxxii
1. PostgreSQL##? xxxii
2. PostgreSQL## xxxii
2.1. ######POSTGRES###### xxxiii
2.2. Postgres95 xxxiii
2.3. PostgreSQL xxxiv
3. ## xxxiv
4. ####### xxxiv
5. ############ xxxiv
5.1. ##### xxxv
5.2. ####### xxxv
5.3. ##### xxxvii
I. ####### 1
1. ######## 3
1.1. ###### 3
1.2. ######## 3
1.3. ######### 3
1.4. ############ 5
2. SQL## 7
2.1. #### 7
2.2. ## 7
2.3. ########## 7
2.4. ######### 8
2.5. ########### 9
2.6. ######## 10
2.7. #### 12
2.8. ## 14
2.9. ## 14
3. ###### 15
3.1. #### 15
3.2. ### 15
3.3. #### 15
3.4. ######## 16
3.5. ####### 17
3.6. ## 19
3.7. ### 21
II. SQL## 22
4. SQL### 30
4.1. ##### 30
4.2. ### 38
4.3. ###### 49
5. ##### 52
5.1. ####### 52
5.2. ###### 53
5.3. Identity Columns 53
5.4. ### 54
5.5. ## 56
5.6. ##### 63
5.7. ####### 64
5.8. ## 66
5.9. ########### 70
5.10. #### 76
5.11. ## 80
5.12. ############## 82
5.13. ##### 94

5.14. ################ 94
5.15. ####### 94
6. ##### 97
6.1. ###### 97
6.2. ###### 98
6.3. ###### 98
6.4. ############# 99
7. ##### 100
7.1. ## 100
7.2. ##### 100
7.3. ##### 113
7.4. ########(UNION, INTERSECT, EXCEPT) 115
7.5. ######(ORDER BY) 116
7.6. LIMIT#OFFSET 116
7.7. VALUES### 117
7.8. WITH############## 118
8. #### 126
8.1. ###### 127
8.2. ### 132
8.3. ### 132
8.4. ######### 134
8.5. ##/###### 136
8.6. ####### 144
8.7. ### 145
8.8. ###### 147
8.9. ########### 149
8.10. ######## 151
8.11. ########### 152
8.12. UUID# 154
8.13. XML# 155
8.14. JSON#### 156
8.15. ## 165
8.16. ### 173
8.17. ### 178
8.18. ##### 183
8.19. ############# 184
8.20. pg_lsn# 186
8.21. ###### 186
9. ###### 188
9.1. ##### 188
9.2. ########## 188
9.3. ######## 192
9.4. ######### 200
9.5. ############# 209
9.6. ############ 212
9.7. ####### 214
9.8. ########## 230
9.9. ##/######## 237
9.10. ######### 252
9.11. ######## 253
9.12. ################ 259
9.13. ############ 262
9.14. UUID## 268
9.15. XML## 268
9.16. JSON###### 281
9.17. ######### 310
9.18. ### 311
9.19. ######## 313

9.20. ############ 316
9.21. #### 322
9.22. ####### 329
9.23. Merge Support Functions 330
9.24. ####### 330
9.25. ####### 333
9.26. ####### 335
9.27. ########### 338
9.28. ######## 356
9.29. ##### 370
9.30. ######### 371
9.31. ###### 373
10. ### 375
10.1. ## 375
10.2. ### 376
10.3. ## 379
10.4. #### 382
10.5. UNION#CASE######## 382
10.6. SELECT### 384
11. ###### 385
11.1. #### 385
11.2. ######### 386
11.3. ######### 387
11.4. #######ORDER BY 388
11.5. ############### 389
11.6. ######## 389
11.7. ########## 390
11.8. ######## 390
11.9. ########################### 393
11.10. ########## 395
11.11. ########## 396
11.12. ############ 396
12. #### 398
12.1. #### 398
12.2. ########## 401
12.3. ######### 403
12.4. #### 409
12.5. ### 413
12.6. ## 415
12.7. ### 423
12.8. ############### 424
12.9. ################### 428
12.10. psql#### 429
12.11. #### 431
13. ###### 433
13.1. #### 433
13.2. ########## 433
13.3. ###### 438
13.4. ######################### 442
13.5. ######## 444
13.6. ## 444
13.7. ########## 444
14. ######### 446
14.1. EXPLAIN### 446
14.2. ############## 457
14.3. ####JOIN########## 461
14.4. ############ 463
14.5. ######## 465

15. ####### 466
15.1. ################# 466
15.2. ###################### 466
15.3. ####### 467
15.4. ###### 469
III. ###### 471
16. ############# 478
17. ############## 479
17.1. #### 479
17.2. ###### 480
17.3. Autoconf#Make########### 481
17.4. Meson############# 492
17.5. ########### 502
17.6. ############### 504
17.7. ############# 504
18. ######### 510
18.1. PostgreSQL######## 510
18.2. ############# 510
18.3. ########### 512
18.4. ########## 515
18.5. ########## 521
18.6. PostgreSQL############## 522
18.7. ########## 524
18.8. ######## 525
18.9. SSL#####TCP/IP## 526
18.10. GSSAPI#####TCP/IP## 529
18.11. SSH##########TCP/IP## 529
18.12. Windows####Event Log### 530
19. ##### 531
19.1. ######## 531
19.2. ####### 534
19.3. ##### 535
19.4. ##### 540
19.5. ########WAL# 548
19.6. ######## 557
19.7. ####### 563
19.8. ########## 568
19.9. ####### 580
19.10. ##Vacuum## 581
19.11. ############# 583
19.12. ##### 592
19.13. ################# 593
19.14. ##### 595
19.15. ########## 595
19.16. ######## 597
19.17. ########## 597
19.18. ####### 601
20. ######## 603
20.1. `pg_hba.conf`#### 603
20.2. ####### 610
20.3. #### 611
20.4. Trust## 612
20.5. ####### 612
20.6. GSSAPI## 613
20.7. SSPI## 614
20.8. Ident## 615
20.9. Peer## 615
20.10. LDAP## 616

20.11. RADIUS## 618
20.12. ##### 619
20.13. PAM## 619
20.14. BSD## 620
20.15. ######### 620
21. ######### 622
21.1. ######### 622
21.2. ###### 623
21.3. ######### 624
21.4. ###### 626
21.5. ####### 626
21.6. ######### 628
22. ######## 629
22.1. ## 629
22.2. ######### 629
22.3. ############ 630
22.4. ######### 631
22.5. ######### 631
22.6. ###### 632
23. ########## 634
23.1. ######### 634
23.2. ######## 638
23.3. ######### 646
24. ############## 656
24.1. ########### 656
24.2. ############## 663
24.3. ######### 663
25. ########### 666
25.1. SQL###### 666
25.2. ################## 668
25.3. ########################PITR# 669
26. ##################### 679
26.1. ######## 679
26.2. ############### 681
26.3. ######## 688
26.4. ######## 689
27. ############# 696
27.1. ####Unix### 696
27.2. ######## 697
27.3. ###### 734
27.4. ######### 735
27.5. #### 742
27.6. Monitoring Disk Usage 749
28. ############ 752
28.1. ### 752
28.2. ######### 753
28.3. ########(WAL) 754
28.4. ####### 754
28.5. WAL### 755
28.6. WAL### 758
29. ########## 759
29.1. ######## 759
29.2. ######### 760
29.3. Logical Replication Failover 765
29.4. ##### 766
29.5. #### 773
29.6. ###### 775
29.7. #### 776

29.8. ####### 777
29.9. ## 777
29.10. ###### 777
29.11. #### 778
29.12. ##### 779
30. ########(JIT) 780
30.1. JIT########## 780
30.2. ######JIT####### 780
30.3. ## 781
30.4. ### 782
31. ########## 783
31.1. ###### 783
31.2. ###### 786
31.3. ########## 788
31.4. TAP### 789
31.5. ############# 790
IV. ############# 791
32. libpq — C ##### 796
32.1. ############ 796
32.2. ###### 813
32.3. ######## 819
32.4. ########## 833
32.5. ######### 838
32.6. Retrieving Query Results in Chunks 842
32.7. ############### 843
32.8. ######### 848
32.9. ##### 849
32.10. COPY######## 849
32.11. #### 853
32.12. ##### 855
32.13. #### 858
32.14. ######## 859
32.15. #### 865
32.16. ######### 867
32.17. ########## 867
32.18. ########LDAP## 868
32.19. SSL#### 868
32.20. ############## 872
32.21. libpq######## 872
32.22. ######### 874
33. ######### 884
33.1. #### 884
33.2. #### 884
33.3. ############# 884
33.4. ####### 888
33.5. ######### 889
34. ECPG — C#########SQL 894
34.1. ## 894
34.2. ########### 894
34.3. SQL####### 897
34.4. ######## 900
34.5. ##SQL 913
34.6. pgtypes ##### 914
34.7. ######## 927
34.8. ##### 938
34.9. ########## 945
34.10. ####SQL######## 947
34.11. ####### 947

34.12. ######### 948
34.13. C++######## 949
34.14. ####SQL#### 952
34.15. Informix##### 974
34.16. Oracle##### 987
34.17. ## 987
35. ###### 990
35.1. #### 990
35.2. #### 990
35.3. information_schema_catalog_name 991
35.4. administrable_role_authorizations 991
35.5. applicable_roles 991
35.6. attributes 991
35.7. character_sets 993
35.8. check_constraint_routine_usage 994
35.9. check_constraints 994
35.10. collations 995
35.11. collation_character_set_applicability 995
35.12. column_column_usage 996
35.13. column_domain_usage 996
35.14. column_options 997
35.15. column_privileges 997
35.16. column_udt_usage 998
35.17. columns 998
35.18. constraint_column_usage 1001
35.19. constraint_table_usage 1001
35.20. data_type_privileges 1002
35.21. domain_constraints 1002
35.22. domain_udt_usage 1003
35.23. domains 1003
35.24. element_types 1005
35.25. enabled_roles 1006
35.26. foreign_data_wrapper_options 1006
35.27. foreign_data_wrappers 1007
35.28. foreign_server_options 1007
35.29. foreign_servers 1008
35.30. foreign_table_options 1008
35.31. foreign_tables 1008
35.32. key_column_usage 1009
35.33. parameters 1009
35.34. referential_constraints 1011
35.35. role_column_grants 1012
35.36. role_routine_grants 1012
35.37. role_table_grants 1013
35.38. role_udt_grants 1013
35.39. role_usage_grants 1014
35.40. routine_column_usage 1015
35.41. routine_privileges 1015
35.42. routine_routine_usage 1016
35.43. routine_sequence_usage 1016
35.44. routine_table_usage 1017
35.45. routines 1017
35.46. schemata 1021
35.47. sequences 1022
35.48. sql_features 1023
35.49. sql_implementation_info 1023
35.50. sql_parts 1023
35.51. sql_sizing 1024

35.52. table_constraints 1024
35.53. table_privileges 1025
35.54. tables 1025
35.55. transforms 1026
35.56. triggered_update_columns 1027
35.57. triggers 1027
35.58. udt_privileges 1029
35.59. usage_privileges 1029
35.60. user_defined_types 1030
35.61. user_mapping_options 1031
35.62. user_mappings 1032
35.63. view_column_usage 1032
35.64. view_routine_usage 1033
35.65. view_table_usage 1033
35.66. views 1034
V. ########## 1035
36. SQL### 1041
36.1. ###### 1041
36.2. PostgreSQL###### 1041
36.3. ####### 1043
36.4. ########### 1044
36.5. ########SQL### 1044
36.6. ########## 1058
36.7. ######## 1058
36.8. ######## 1060
36.9. #### 1060
36.10. C#### 1060
36.11. ########### 1080
36.12. ######## 1081
36.13. ####### 1086
36.14. ######### 1090
36.15. ############ 1090
36.16. ################### 1093
36.17. ######################## 1104
36.18. ###### 1110
37. ### 1115
37.1. ######## 1115
37.2. ######### 1117
37.3. C########### 1117
37.4. ######## 1120
38. ####### 1124
38.1. ############ 1124
38.2. ############## 1125
38.3. C################## 1128
38.4. ############ 1129
38.5. ################# 1130
38.6. A Database Login Event Trigger Example 1131
39. ####### 1133
39.1. ########## 1133
39.2. ########### 1134
39.3. ########### 1140
39.4. INSERT#UPDATE#DELETE######## 1143
39.5. ###### 1152
39.6. ############# 1153
39.7. ####### 1154
40. ##### 1156
40.1. ############ 1156
41. PL/pgSQL — SQL##### 1158

41.1. ## 1158
41.2. PL/pgSQL### 1159
41.3. ## 1160
41.4. # 1166
41.5. ##### 1167
41.6. #### 1174
41.7. #### 1187
41.8. ########## 1192
41.9. ######### 1193
41.10. ##### 1195
41.11. PL/pgSQL### 1202
41.12. PL/pgSQL########## 1205
41.13. Oracle PL/SQL##### 1208
42. PL/Tcl — Tcl##### 1217
42.1. ## 1217
42.2. PL/Tcl##### 1217
42.3. PL/Tcl######### 1219
42.4. PL/Tcl########### 1219
42.5. PL/Tcl############# 1219
42.6. PL/Tcl###### 1221
42.7. PL/Tcl############# 1223
42.8. PL/Tcl###### 1223
42.9. PL/Tcl################# 1224
42.10. ########## 1225
42.11. PL/Tcl### 1225
42.12. Tcl####### 1226
43. PL/Perl — Perl##### 1227
43.1. PL/Perl##### 1227
43.2. PL/Perl######## 1231
43.3. ###### 1231
43.4. PL/Perl########### 1236
43.5. #####PL/Perl#########PL/Perl 1236
43.6. PL/Perl### 1237
43.7. PL/Perl####### 1239
43.8. PL/Perl### 1239
44. PL/Python — Python##### 1241
44.1. PL/Python## 1241
44.2. #### 1242
44.3. ###### 1247
44.4. ######### 1247
44.5. ##### 1247
44.6. ########## 1248
44.7. ############# 1251
44.8. ########## 1252
44.9. ######### 1252
44.10. Python 2#Python 3 1253
44.11. #### 1253
45. ################# 1255
45.1. ######### 1255
45.2. ############# 1294
45.3. ##### 1303
45.4. ########## 1313
45.5. ######### 1316
45.6. # 1316
46. ################ 1320
47. ########### 1323
47.1. ############# 1323
47.2. ################# 1326

47.3. ########################## 1328
47.4. ##########SQL###### 1328
47.5. ###################### 1329
47.6. ################## 1329
47.7. ############### 1335
47.8. ############################## 1336
47.9. ################################## 1336
47.10. #############2##### 1337
48. ############ 1339
49. ######### 1340
49.1. ##### 1340
49.2. ############### 1340
VI. ###### 1342
I. SQL#### 1347
ABORT 1351
ALTER AGGREGATE 1352
ALTER COLLATION 1354
ALTER CONVERSION 1356
ALTER DATABASE 1357
ALTER DEFAULT PRIVILEGES 1359
ALTER DOMAIN 1362
ALTER EVENT TRIGGER 1365
ALTER EXTENSION 1366
ALTER FOREIGN DATA WRAPPER 1369
ALTER FOREIGN TABLE 1371
ALTER FUNCTION 1376
ALTER GROUP 1379
ALTER INDEX 1381
ALTER LANGUAGE 1384
ALTER LARGE OBJECT 1385
ALTER MATERIALIZED VIEW 1386
ALTER OPERATOR 1388
ALTER OPERATOR CLASS 1390
ALTER OPERATOR FAMILY 1391
ALTER POLICY 1394
ALTER PROCEDURE 1395
ALTER PUBLICATION 1398
ALTER ROLE 1401
ALTER ROUTINE 1405
ALTER RULE 1406
ALTER SCHEMA 1407
ALTER SEQUENCE 1408
ALTER SERVER 1411
ALTER STATISTICS 1413
ALTER SUBSCRIPTION 1414
ALTER SYSTEM 1417
ALTER TABLE 1419
ALTER TABLESPACE 1436
ALTER TEXT SEARCH CONFIGURATION 1438
ALTER TEXT SEARCH DICTIONARY 1440
ALTER TEXT SEARCH PARSER 1442
ALTER TEXT SEARCH TEMPLATE 1443
ALTER TRIGGER 1444
ALTER TYPE 1446
ALTER USER 1450
ALTER USER MAPPING 1451
ALTER VIEW 1452
ANALYZE 1454

BEGIN 1457
CALL 1459
CHECKPOINT 1460
CLOSE 1461
CLUSTER 1462
COMMENT 1465
COMMIT 1469
COMMIT PREPARED 1470
COPY 1471
CREATE ACCESS METHOD 1480
CREATE AGGREGATE 1481
CREATE CAST 1487
CREATE COLLATION 1491
CREATE CONVERSION 1494
CREATE DATABASE 1496
CREATE DOMAIN 1500
CREATE EVENT TRIGGER 1503
CREATE EXTENSION 1505
CREATE FOREIGN DATA WRAPPER 1507
CREATE FOREIGN TABLE 1509
CREATE FUNCTION 1513
CREATE GROUP 1521
CREATE INDEX 1522
CREATE LANGUAGE 1530
CREATE MATERIALIZED VIEW 1532
CREATE OPERATOR 1534
CREATE OPERATOR CLASS 1537
CREATE OPERATOR FAMILY 1540
CREATE POLICY 1541
CREATE PROCEDURE 1546
CREATE PUBLICATION 1549
CREATE ROLE 1553
CREATE RULE 1557
CREATE SCHEMA 1560
CREATE SEQUENCE 1562
CREATE SERVER 1566
CREATE STATISTICS 1568
CREATE SUBSCRIPTION 1572
CREATE TABLE 1577
CREATE TABLE AS 1596
CREATE TABLESPACE 1599
CREATE TEXT SEARCH CONFIGURATION 1601
CREATE TEXT SEARCH DICTIONARY 1602
CREATE TEXT SEARCH PARSER 1604
CREATE TEXT SEARCH TEMPLATE 1606
CREATE TRANSFORM 1607
CREATE TRIGGER 1609
CREATE TYPE 1615
CREATE USER 1623
CREATE USER MAPPING 1624
CREATE VIEW 1625
DEALLOCATE 1630
DECLARE 1631
DELETE 1634
DISCARD 1637
DO 1638
DROP ACCESS METHOD 1640
DROP AGGREGATE 1641

DROP CAST 1643
DROP COLLATION 1644
DROP CONVERSION 1645
DROP DATABASE 1646
DROP DOMAIN 1647
DROP EVENT TRIGGER 1648
DROP EXTENSION 1649
DROP FOREIGN DATA WRAPPER 1650
DROP FOREIGN TABLE 1651
DROP FUNCTION 1652
DROP GROUP 1654
DROP INDEX 1655
DROP LANGUAGE 1657
DROP MATERIALIZED VIEW 1658
DROP OPERATOR 1659
DROP OPERATOR CLASS 1661
DROP OPERATOR FAMILY 1663
DROP OWNED 1664
DROP POLICY 1665
DROP PROCEDURE 1666
DROP PUBLICATION 1668
DROP ROLE 1669
DROP ROUTINE 1670
DROP RULE 1671
DROP SCHEMA 1672
DROP SEQUENCE 1673
DROP SERVER 1674
DROP STATISTICS 1675
DROP SUBSCRIPTION 1676
DROP TABLE 1678
DROP TABLESPACE 1679
DROP TEXT SEARCH CONFIGURATION 1680
DROP TEXT SEARCH DICTIONARY 1681
DROP TEXT SEARCH PARSER 1682
DROP TEXT SEARCH TEMPLATE 1683
DROP TRANSFORM 1684
DROP TRIGGER 1685
DROP TYPE 1686
DROP USER 1687
DROP USER MAPPING 1688
DROP VIEW 1689
END 1690
EXECUTE 1691
EXPLAIN 1692
FETCH 1698
GRANT 1702
IMPORT FOREIGN SCHEMA 1707
INSERT 1709
LISTEN 1715
LOAD 1717
LOCK 1718
MERGE 1721
MOVE 1727
NOTIFY 1729
PREPARE 1731
PREPARE TRANSACTION 1733
REASSIGN OWNED 1735
REFRESH MATERIALIZED VIEW 1736

REINDEX 1738
RELEASE SAVEPOINT 1743
RESET 1745
REVOKE 1746
ROLLBACK 1750
ROLLBACK PREPARED 1751
ROLLBACK TO SAVEPOINT 1752
SAVEPOINT 1754
SECURITY LABEL 1756
SELECT 1759
SELECT INTO 1777
SET 1779
SET CONSTRAINTS 1782
SET ROLE 1783
SET SESSION AUTHORIZATION 1785
SET TRANSACTION 1787
SHOW 1790
START TRANSACTION 1792
TRUNCATE 1793
UNLISTEN 1795
UPDATE 1797
VACUUM 1802
VALUES 1806
II. PostgreSQL############## 1808
clusterdb 1809
createdb 1812
createuser 1815
dropdb 1819
dropuser 1822
ecpg 1824
pg_amcheck 1826
pg_basebackup 1831
pgbench 1839
pg_combinebackup 1861
pg_config 1864
pg_dump 1867
pg_dumpall 1880
pg_isready 1886
pg_receivewal 1888
pg_recvlogical 1892
pg_restore 1896
pg_verifybackup 1905
psql 1908
reindexdb 1945
vacuumdb 1948
III. PostgreSQL########### 1953
initdb 1954
pg_archivecleanup 1959
pg_checksums 1961
pg_controldata 1963
pg_createsubscriber 1964
pg_ctl 1968
pg_resetwal 1973
pg_rewind 1977
pg_test_fsync 1981
pg_test_timing 1982
pg_upgrade 1985
pg_waldump 1994

pg_walsummary 1997
postgres 1998
VII. #### 2004
50. PostgreSQL##### 2010
50.1. ######## 2010
50.2. ##### 2010
50.3. ###### 2010
50.4. PostgreSQL####### 2011
50.5. ####/####### 2012
50.6. ####### 2013
51. ######## 2014
51.1. ## 2014
51.2. pg_aggregate 2015
51.3. pg_am 2017
51.4. pg_amop 2017
51.5. pg_amproc 2018
51.6. pg_attrdef 2019
51.7. pg_attribute 2019
51.8. pg_authid 2021
51.9. pg_auth_members 2022
51.10. pg_cast 2022
51.11. pg_class 2023
51.12. pg_collation 2025
51.13. pg_constraint 2026
51.14. pg_conversion 2028
51.15. pg_database 2028
51.16. pg_db_role_setting 2029
51.17. pg_default_acl 2030
51.18. pg_depend 2030
51.19. pg_description 2032
51.20. pg_enum 2032
51.21. pg_event_trigger 2033
51.22. pg_extension 2033
51.23. pg_foreign_data_wrapper 2034
51.24. pg_foreign_server 2034
51.25. pg_foreign_table 2035
51.26. pg_index 2035
51.27. pg_inherits 2037
51.28. pg_init_privs 2037
51.29. pg_language 2037
51.30. pg_largeobject 2038
51.31. pg_largeobject_metadata 2038
51.32. pg_namespace 2039
51.33. pg_opclass 2039
51.34. pg_operator 2040
51.35. pg_opfamily 2041
51.36. pg_parameter_acl 2041
51.37. pg_partitioned_table 2041
51.38. pg_policy 2042
51.39. pg_proc 2043
51.40. pg_publication 2045
51.41. pg_publication_namespace 2045
51.42. pg_publication_rel 2046
51.43. pg_range 2046
51.44. pg_replication_origin 2047
51.45. pg_rewrite 2047
51.46. pg_seclabel 2047
51.47. pg_sequence 2048

51.48. pg_shdepend 2048
51.49. pg_shdescription 2049
51.50. pg_shseclabel 2050
51.51. pg_statistic 2050
51.52. pg_statistic_ext 2051
51.53. pg_statistic_ext_data 2052
51.54. pg_subscription 2053
51.55. pg_subscription_rel 2054
51.56. pg_tablespace 2054
51.57. pg_transform 2055
51.58. pg_trigger 2055
51.59. pg_ts_config 2056
51.60. pg_ts_config_map 2057
51.61. pg_ts_dict 2057
51.62. pg_ts_parser 2058
51.63. pg_ts_template 2058
51.64. pg_type 2059
51.65. pg_user_mapping 2062
52. ####### 2063
52.1. ## 2063
52.2. pg_available_extensions 2064
52.3. pg_available_extension_versions 2064
52.4. pg_backend_memory_contexts 2065
52.5. pg_config 2065
52.6. pg_cursors 2066
52.7. pg_file_settings 2066
52.8. pg_group 2067
52.9. pg_hba_file_rules 2067
52.10. pg_ident_file_mappings 2068
52.11. pg_indexes 2069
52.12. pg_locks 2069
52.13. pg_matviews 2071
52.14. pg_policies 2072
52.15. pg_prepared_statements 2072
52.16. pg_prepared_xacts 2073
52.17. pg_publication_tables 2073
52.18. pg_replication_origin_status 2074
52.19. pg_replication_slots 2074
52.20. pg_roles 2076
52.21. pg_rules 2077
52.22. pg_seclabels 2077
52.23. pg_sequences 2077
52.24. pg_settings 2078
52.25. pg_shadow 2080
52.26. pg_shmem_allocations 2081
52.27. pg_stats 2081
52.28. pg_stats_ext 2082
52.29. pg_stats_ext_exprs 2083
52.30. pg_tables 2085
52.31. pg_timezone_abbrevs 2085
52.32. pg_timezone_names 2086
52.33. pg_user 2086
52.34. pg_user_mappings 2087
52.35. pg_views 2087
52.36. pg_wait_events 2088
53. #######/########### 2089
53.1. ## 2089
53.2. ######## 2090

53.3. SASL## 2101
53.4. ################### 2102
53.5. ####################### 2111
53.6. ######### 2112
53.7. ######## 2113
53.8. ################## 2128
53.9. ################## 2130
53.10. #####2.0######### 2138
54. PostgreSQL######## 2140
54.1. ## 2140
54.2. ############## 2140
54.3. ################ 2143
54.4. ############ 2147
55. ####### 2149
55.1. #### 2149
55.2. ###### 2151
56. ############ 2154
57. ############ 2156
57.1. ########### 2156
57.2. ################### 2156
57.3. ############## 2169
57.4. ################### 2170
57.5. ############### 2172
58. ################## 2174
58.1. ################ 2174
59. ################ 2177
59.1. ############# 2177
59.2. ############# 2178
59.3. ########### 2179
60. ############### 2182
60.1. ################### 2182
60.2. ######### 2182
60.3. PostgreSQL############GEQO# 2183
60.4. ######### 2184
61. ###################### 2185
62. ######################## 2186
62.1. #########API## 2186
62.2. ################ 2188
62.3. ######### 2193
62.4. ################## 2194
62.5. ########### 2195
62.6. ############# 2196
63. Write Ahead Logging for Extensions 2199
63.1. ##WAL#### 2199
63.2. ####WAL######## 2200
64. Built-in Index Access Methods 2203
64.1. B-Tree###### 2203
64.2. GiST###### 2208
64.3. SP-GiST###### 2223
64.4. GIN###### 2233
64.5. BRIN###### 2238
64.6. ######### 2250
65. ############# 2252
65.1. ################ 2252
65.2. TOAST 2253
65.3. ####### 2255
65.4. ###### 2256
65.5. ####### 2256

65.6. ############### 2256
65.7. ########(HOT) 2258
66. ######### 2260
66.1. ########### 2260
66.2. ########### 2260
66.3. ######### 2260
66.4. 2######## 2261
67. ################ 2262
67.1. ############## 2262
67.2. ############# 2263
67.3. BKI###### 2267
67.4. BKI#### 2267
67.5. BKI################ 2268
67.6. BKI## 2268
68. ################### 2269
68.1. ###### 2269
68.2. ####### 2273
68.3. ################ 2276
69. ############## 2278
69.1. ######################### 2278
69.2. ######################## 2278
69.3. #############WAL######## 2279
VIII. ## 2280
A. PostgreSQL###### 2286
B. ##/####### 2294
B.1. ##/####### 2294
B.2. ################### 2295
B.3. ##/####### 2295
B.4. ##/######## 2296
B.5. POSIX###### 2297
B.6. ##### 2299
B.7. #####(Julian Date) 2299
C. SQL##### 2301
D. SQL#### 2324
D.1. ########### 2325
D.2. ########### 2336
D.3. XML####SQL/XML#### 2344
E. ####### 2347
E.1. ####17 2347
E.2. ####### 2358
F. ################ 2359
F.1. amcheck — ######################## 2360
F.2. auth_delay — ########### 2365
F.3. auth_explain — ############### 2366
F.4. basebackup_to_shell — pg_basebackup#####"shell"## 2369
F.5. basic_archive — WAL############ 2370
F.6. bloom — ######################## 2371
F.7. btree_gin — GIN#######B-tree## 2375
F.8. btree_gist — GiST#######B-tree## 2376
F.9. citext — ################ 2378
F.10. cube — ######### 2381
F.11. dblink — ##PostgreSQL########### 2385
F.12. dict_int — ################ 2415
F.13. dict_xsyn — ############## 2416
F.14. earthdistance — ######## 2418
F.15. file_fdw — ############################## 2420
F.16. fuzzystrmatch — ############### 2423
F.17. hstore — hstore##/##### 2427

F.18. intagg — ########## 2435
F.19. intarray — ######### 2437
F.20. isn — ######ISBN#EAN#UPC######### 2440
F.21. lo — ############# 2444
F.22. ltree — ############# 2446
F.23. pageinspect — ################ 2453
F.24. passwordcheck — ############ 2463
F.25. pg_buffercache — PostgreSQL################ 2464
F.26. pgcrypto — #### 2468
F.27. pg_freespacemap — ########### 2478
F.28. pg_prewarm — ########################## 2480
F.29. pgrowlocks — ############## 2482
F.30. pg_stat_statements — SQL##################### 2484
F.31. pgstattuple — ############### 2492
F.32. pg_surgery — ####################### 2496
F.33. pg_trgm — ########################## 2498
F.34. pg_visibility — ############### 2503
F.35. pg_walinspect — #####WAL## 2505
F.36. postgres_fdw — ###PostgreSQL################## 2508
F.37. seg — ################### 2517
F.38. sepgsql — SELinux###################MAC########### 2520
F.39. spi — ##################/# 2527
F.40. sslinfo — ######SSL###### 2528
F.41. tablefunc — ########(`crosstab`#) 2530
F.42. tcn — ####################### 2539
F.43. test_decoding — SQL####WAL#################/###### 2540
F.44. tsm_system_rows — `TABLESAMPLE`####`SYSTEM_ROWS`######### 2541
F.45. tsm_system_time — `TABLESAMPLE`####`SYSTEM_TIME`######### 2542
F.46. unaccent — ################ 2543
F.47. uuid-ossp — UUID### 2545
F.48. xml2 — XPath#####XSLT## 2547
G. ############ 2551
G.1. ############# 2551
G.2. ########## 2558
H. ####### 2559
H.1. ############ 2559
H.2. ##### 2559
H.3. ##### 2559
H.4. ## 2559
I. ########## 2560
I.1. Git########### 2560
J. ######### 2561
J.1. DocBook 2561
J.2. ###### 2561
J.3. Make########## 2563
J.4. Meson########## 2564
J.5. ##### 2564
J.6. ###### 2565
K. PostgreSQL### 2567
L. ### 2568
M. ### 2575
N. ### 2587
N.1. ######## 2587
N.2. ###### 2587
O. ############## 2588
O.1. `recovery.conf`#####`postgresql.conf`### 2588
O.2. ##################### 2588
O.3. `pg_xlogdump`####`pg_waldump`### 2588

O.4. pg_resetxlog####pg_resetwal### 2588
O.5. pg_receivexlog####pg_receivewal### 2588
P. ### 2589
..... 2594
..... 2596

####

60.1. ########### 2183
64.1. GIN### 2237
65.1. ######## 2257

####

4.1. ################## 32
4.2. ##################### 37
5.1. ACL### 69
5.2. ########## 69
8.1. #### 126
8.2. ###### 127
8.3. ### 132
8.4. ### 132
8.5. ######### 134
8.6. ######### 134
8.7. #############bytea#### 135
8.8. bytea################ 135
8.9. ##/###### 136
8.10. #### 137
8.11. #### 138
8.12. ##### 138
8.13. #####/#### 140
8.14. ##/####### 140
8.15. ######## 141
8.16. ISO 8601############## 142
8.17. ###### 143
8.18. ########## 144
8.19. ####### 144
8.20. ###### 147
8.21. ############## 149
8.22. cidr####### 149
8.23. JSON########PostgreSQL##### 157
8.24. jsonpath## 165
8.25. jsonpath Accessors 165
8.26. ############# 184
8.27. ###### 186
9.1. ##### 189
9.2. #### 189
9.3. #### 192
9.4. ##### 192
9.5. #### 194
9.6. #### 197
9.7. #### 198
9.8. ##### 199
9.9. SQL######### 200
9.10. ############### 202
9.11. SQL############# 209
9.12. ############# 210
9.13. ####/########### 212
9.14. ######### 213
9.15. ######## 213
9.16. ########## 217
9.17. ######## 222
9.18. ######## 222
9.19. ###### 223
9.20. ############### 224
9.21. ############## 225
9.22. ########### 225
9.23. ######## 226
9.24. ARE########### 226

9.25. ######## 229
9.26. ###### 230
9.27. ##/################ 231
9.28. ##/################### 232
9.29. ################ 234
9.30. ################### 235
9.31. `to_char`## 236
9.32. ##/##### 237
9.33. ##/####### 239
9.34. `AT TIME ZONE` and `AT LOCAL` Variants 249
9.35. ######## 252
9.36. ######## 253
9.37. ######## 256
9.38. ####### 257
9.39. IP####### 259
9.40. IP###### 261
9.41. MAC###### 262
9.42. ######## 262
9.43. ######## 263
9.44. ############ 267
9.45. `json`#`jsonb`### 281
9.46. ##`jsonb`### 282
9.47. JSON#### 284
9.48. SQL/JSON###### 285
9.49. JSON#### 286
9.50. `jsonpath`######## 297
9.51. `jsonpath`######## 301
9.52. SQL/JSON Query Functions 303
9.53. ####### 310
9.54. ##### 313
9.55. #### 314
9.56. ##### 317
9.57. ####### 318
9.58. #### 320
9.59. ###### 321
9.60. ###### 322
9.61. ########## 326
9.62. ######## 327
9.63. ######## 328
9.64. ####### 328
9.65. ######## 329
9.66. Merge Support Functions 330
9.67. ####### 335
9.68. ####### 337
9.69. ######## 339
9.70. ########## 341
9.71. `aclitem`### 343
9.72. `aclitem`## 343
9.73. ########### 344
9.74. ############ 345
9.75. ########## 348
9.76. ######## 348
9.77. ################# 349
9.78. GUC Flags 349
9.79. ################ 349
9.80. ######## 350
9.81. ########## 350
9.82. ########ID############# 351

9.83. ########### 352
9.84. ############ID############ 352
9.85. ###################### 353
9.86. ####### 353
9.87. pg_control_checkpoint#### 353
9.88. pg_control_system#### 354
9.89. pg_control_init Output Columns 354
9.90. pg_control_recovery#### 354
9.91. Version Information Functions 355
9.92. WAL Summarization Information Functions 355
9.93. ###### 356
9.94. ########## 356
9.95. ######### 358
9.96. ######## 359
9.97. ######## 360
9.98. ########### 361
9.99. ########### 361
9.100. ################ 364
9.101. ################ 365
9.102. ######## 366
9.103. ########## 366
9.104. ######### 367
9.105. ########### 367
9.106. ######## 369
9.107. ######## 370
9.108. ########### 373
12.1. ############## 414
13.1. ############# 433
13.2. ######## 440
13.3. ######### 441
18.1. System V IPC##### 515
18.2. SSL########### 527
19.1. synchronous_commit### 549
19.2. ########## 573
19.3. JSON######## 579
19.4. ######## 602
21.1. ####### 626
23.1. ICU####### 643
23.2. ICU###### 644
23.3. PostgreSQL##### 647
23.4. ####################### 651
23.5. ############### 652
26.1. ####################### 680
27.1. ######## 698
27.2. ########## 699
27.3. pg_stat_activity### 701
27.4. ####### 702
27.5. Wait Events of Type Activity 703
27.6. Wait Events of Type Bufferpin 704
27.7. Wait Events of Type Client 704
27.8. Wait Events of Type Extension 704
27.9. Wait Events of Type Io 704
27.10. Wait Events of Type Ipc 707
27.11. Wait Events of Type Lock 709
27.12. Wait Events of Type Lwlock 710
27.13. Wait Events of Type Timeout 713
27.14. pg_stat_replication### 714
27.15. pg_stat_replication_slots### 716

27.16. pg_stat_wal_receiver### 717
27.17. pg_stat_recovery_prefetch### 718
27.18. pg_stat_subscription### 719
27.19. pg_stat_subscription_stats### 719
27.20. pg_stat_ssl### 720
27.21. pg_stat_gssapi### 720
27.22. pg_stat_archiver### 721
27.23. pg_stat_io View 721
27.24. pg_stat_bgwriter### 723
27.25. pg_stat_checkpointer View 723
27.26. pg_stat_wal### 724
27.27. pg_stat_database### 725
27.28. pg_stat_database_conflicts### 726
27.29. pg_stat_all_tables### 727
27.30. pg_stat_all_indexes### 728
27.31. pg_statio_all_tables### 729
27.32. pg_statio_all_indexes### 730
27.33. pg_statio_all_sequences### 730
27.34. pg_stat_user_functions### 731
27.35. pg_stat_slru### 731
27.36. ########## 732
27.37. ############### 734
27.38. pg_stat_progress_analyze### 735
27.39. ANALYZE##### 735
27.40. pg_stat_progress_cluster### 736
27.41. CLUSTER#VACUUM FULL##### 737
27.42. pg_stat_progress_copy### 737
27.43. pg_stat_progress_create_index### 738
27.44. CREATE INDEX##### 738
27.45. pg_stat_progress_vacuum### 739
27.46. VACUUM##### 740
27.47. pg_stat_progress_basebackup### 741
27.48. ############## 741
27.49. #######DTrace#### 742
27.50. ################## 747
29.1. UPDATE#### 767
32.1. SSL###### 870
32.2. libpq/##########SSL######## 871
33.1. SQL############ 888
34.1. PostgreSQL#####C######## 902
34.2. ###PGTYPESdate_from_asc##### 918
34.3. ###PGTYPESdate_fmt_asc##### 920
34.4. ###rdefmtdate##### 920
34.5. ###PGTYPEStimestamp_from_asc##### 921
35.1. information_schema_catalog_name## 991
35.2. administrable_role_authorizations## 991
35.3. applicable_roles## 991
35.4. attributes## 992
35.5. character_sets## 994
35.6. check_constraint_routine_usage## 994
35.7. check_constraints## 995
35.8. collations## 995
35.9. collation_character_set_applicability## 995
35.10. column_column_usage## 996
35.11. column_domain_usage## 996
35.12. column_options## 997
35.13. column_privileges## 997
35.14. column_udt_usage## 998

35.15. columns## 998
35.16. constraint_column_usage## 1001
35.17. constraint_table_usage## 1001
35.18. data_type_privileges## 1002
35.19. domain_constraints## 1002
35.20. domain_udt_usage## 1003
35.21. domains## 1003
35.22. element_types## 1005
35.23. enabled_roles## 1006
35.24. foreign_data_wrapper_options## 1007
35.25. foreign_data_wrappers## 1007
35.26. foreign_server_options## 1007
35.27. foreign_servers## 1008
35.28. foreign_table_options## 1008
35.29. foreign_tables## 1009
35.30. key_column_usage## 1009
35.31. parameters## 1010
35.32. referential_constraints## 1011
35.33. role_column_grants## 1012
35.34. role_routine_grants## 1012
35.35. role_table_grants## 1013
35.36. role_udt_grants## 1014
35.37. role_usage_grants## 1014
35.38. routine_column_usage Columns 1015
35.39. routine_privileges## 1015
35.40. routine_routine_usage Columns 1016
35.41. routine_sequence_usage Columns 1016
35.42. routine_table_usage Columns 1017
35.43. routines## 1018
35.44. schemata## 1021
35.45. sequences## 1022
35.46. sql_features## 1023
35.47. sql_implementation_info## 1023
35.48. sql_parts## 1024
35.49. sql_sizing## 1024
35.50. table_constraints## 1024
35.51. table_privileges## 1025
35.52. tables## 1026
35.53. transforms## 1026
35.54. triggered_update_columns## 1027
35.55. triggers## 1027
35.56. udt_privileges## 1029
35.57. usage_privileges## 1029
35.58. user_defined_types## 1030
35.59. user_mapping_options## 1031
35.60. user_mappings## 1032
35.61. view_column_usage## 1032
35.62. view_routine_usage## 1033
35.63. view_table_usage## 1033
35.64. views## 1034
36.1. ### 1042
36.2. ####SQL######C## 1062
36.3. B-tree##### 1094
36.4. ######## 1094
36.5. GiST###2####R-tree###### 1094
36.6. SP-GiST########### 1095
36.7. GIN ######## 1095
36.8. BRIN Minmax##### 1095

36.9. B-tree###### 1096
36.10. ########## 1096
36.11. GiST###### 1096
36.12. SP-GiST###### 1097
36.13. GIN###### 1097
36.14. BRIN###### 1097
38.1. ######################### 1125
41.1. ############ 1173
41.2. ####### 1185
297. ################### 1543
298. pgbench Automatic Variables 1847
299. pgbench#### 1849
300. pgbench### 1851
51.1. ######## 2014
51.2. pg_aggregate## 2016
51.3. pg_am## 2017
51.4. pg_amop## 2017
51.5. pg_amproc## 2018
51.6. pg_attrdef## 2019
51.7. pg_attribute## 2019
51.8. pg_authid## 2021
51.9. pg_auth_members## 2022
51.10. pg_cast## 2022
51.11. pg_class## 2023
51.12. pg_collation## 2025
51.13. pg_constraint## 2026
51.14. pg_conversion## 2028
51.15. pg_database## 2028
51.16. pg_db_role_setting## 2030
51.17. pg_default_acl## 2030
51.18. pg_depend## 2030
51.19. pg_description## 2032
51.20. pg_enum## 2032
51.21. pg_event_trigger## 2033
51.22. pg_extension## 2033
51.23. pg_foreign_data_wrapper## 2034
51.24. pg_foreign_server## 2035
51.25. pg_foreign_table## 2035
51.26. pg_index## 2035
51.27. pg_inherits## 2037
51.28. pg_init_privs## 2037
51.29. pg_language## 2038
51.30. pg_largeobject## 2038
51.31. pg_largeobject_metadata## 2039
51.32. pg_namespace## 2039
51.33. pg_opclass## 2039
51.34. pg_operator## 2040
51.35. pg_opfamily## 2041
51.36. pg_parameter_acl Columns 2041
51.37. pg_partitioned_table## 2042
51.38. pg_policy## 2042
51.39. pg_proc## 2043
51.40. pg_publication## 2045
51.41. pg_publication_namespace Columns 2045
51.42. pg_publication_rel## 2046
51.43. pg_range## 2046
51.44. pg_replication_origin## 2047
51.45. pg_rewrite## 2047

51.46. pg_seclabel## 2048
51.47. pg_sequence## 2048
51.48. pg_shdepend## 2049
51.49. pg_shdescription## 2050
51.50. pg_shseclabel## 2050
51.51. pg_statistic## 2051
51.52. pg_statistic_ext## 2051
51.53. pg_statistic_ext_data## 2052
51.54. pg_subscription## 2053
51.55. pg_subscription_rel## 2054
51.56. pg_tablespace## 2054
51.57. pg_transform## 2055
51.58. pg_trigger## 2055
51.59. pg_ts_config## 2057
51.60. pg_ts_config_map## 2057
51.61. pg_ts_dict## 2057
51.62. pg_ts_parser## 2058
51.63. pg_ts_template## 2058
51.64. pg_type## 2059
51.65. typcategory#### 2061
51.66. pg_user_mapping## 2062
52.1. ####### 2063
52.2. pg_available_extensions## 2064
52.3. pg_available_extension_versions## 2064
52.4. pg_backend_memory_contexts Columns 2065
52.5. pg_config## 2066
52.6. pg_cursors## 2066
52.7. pg_file_settings## 2067
52.8. pg_group## 2067
52.9. pg_hba_file_rules## 2068
52.10. pg_ident_file_mappings Columns 2068
52.11. pg_indexes## 2069
52.12. pg_locks## 2069
52.13. pg_matviews## 2071
52.14. pg_policies## 2072
52.15. pg_prepared_statements## 2072
52.16. pg_prepared_xacts## 2073
52.17. pg_publication_tables## 2074
52.18. pg_replication_origin_status## 2074
52.19. pg_replication_slots## 2074
52.20. pg_roles## 2076
52.21. pg_rules## 2077
52.22. pg_seclabels## 2077
52.23. pg_sequences## 2078
52.24. pg_settings## 2078
52.25. pg_shadow## 2080
52.26. pg_shmem_allocations## 2081
52.27. pg_stats## 2081
52.28. pg_stats_ext## 2083
52.29. pg_stats_ext_exprs Columns 2084
52.30. pg_tables## 2085
52.31. pg_timezone_abbrevs## 2085
52.32. pg_timezone_names## 2086
52.33. pg_user## 2086
52.34. pg_user_mappings## 2087
52.35. pg_views## 2087
52.36. pg_wait_events Columns 2088
64.1. ####GiST###### 2209

64.2. ####SP-GiST###### 2223
64.3. ####GIN###### 2234
64.4. ####BRIN###### 2240
64.5. Minmax################ 2248
64.6. Inclusion################ 2248
64.7. Bloom#################### 2249
64.8. minmax-multi#################### 2250
65.1. PGDATA### 2252
65.2. ######## 2256
65.3. PageHeaderData###### 2257
65.4. HeapTupleHeaderData###### 2258
A.1. PostgreSQL###### 2286
B.1. ## 2295
B.2. ### 2296
B.3. ##/########### 2296
C.1. SQL##### 2301
F.1. cube##### 2381
F.2. cube#### 2381
F.3. cube### 2382
F.4. cube#####earthdistance### 2418
F.5. point#####earthdistance#### 2419
F.6. hstore#### 2427
F.7. hstore### 2429
F.8. intarray## 2437
F.9. intarray### 2438
F.10. isn#### 2440
F.11. isn Functions 2441
F.12. ltree### 2447
F.13. ltree## 2448
F.14. pg_buffercache## 2464
F.15. pg_buffercache_summary()### 2465
F.16. pg_buffercache_usage_counts()### 2465
F.17. crypt()############ 2469
F.18. crypt()######## 2469
F.19. ############ 2470
F.20. pgrowlocks#### 2482
F.21. pg_stat_statements## 2484
F.22. pg_stat_statements_info## 2488
F.23. pgstattuple#### 2492
F.24. pgstattuple_approx#### 2495
F.25. pg_trgm Functions 2498
F.26. pg_trgm Operators 2499
F.27. seg#### 2517
F.28. ###SEG#### 2517
F.29. Seg GiST### 2518
F.30. sepgsql## 2525
F.31. tablefunc### 2530
F.32. connectby##### 2536
F.33. UUID##### 2545
F.34. UUID####### 2545
F.35. xml2## 2547
F.36. xpath_table###### 2548
K.1. PostgreSQL### 2567

####

8.1. ######## 133
8.2. boolean#### 145
8.3. ########## 151
9.1. SQL/XML###HTML#####XSLT####### 280
10.1. ######### 377
10.2. ########### 377
10.3. ############ 377
10.4. ########## 378
10.5. ########### 378
10.6. ######### 380
10.7. ########## 380
10.8. ########## 381
10.9. character######## 382
10.10. Union############## 383
10.11. ###Union####### 383
10.12. #####UNION####### 383
10.13. ####UNION####### 384
11.1. ###################### 390
11.2. ######################### 391
11.3. ############## 392
11.4. ############################ 392
20.1. pg_hba.conf##### 608
20.2. pg_ident.conf###### 611
32.1. libpq ######## 1 874
32.2. libpq ######## 2 876
32.3. libpq ######## 3 879
33.1. Libpq######################## 889
34.1. SQLDA####### 936
34.2. ################ECPG##### 948
40.1. PL/Perl############# 1157
41.1. ################ 1171
41.2. UPDATE/INSERT### 1184
41.3. PL/pgSQL##### 1196
41.4. PL/pgSQL########### 1197
41.5. ######PL/pgSQL######## 1198
41.6. ###############PL/pgSQL##### 1199
41.7. ######### 1201
41.8. PL/pgSQL######## 1202
41.9. ######PL/SQL##PL/pgSQL#### 1209
41.10. #########PL/SQL#PL/pgSQL### 1209
41.11. #########OUT########PL/SQL#######PL/pgSQL#### 1210
41.12. PL/SQL#######PL/pgSQL#### 1212
F.1. PostgreSQL CSV ########### 2421
F.2. Create a Foreign Table with an Option on a Column 2422

####

###PostgreSQL################ PostgreSQL##############PostgreSQL################ ########## ########PostgreSQL######################

PostgreSQL###################################### ####################PostgreSQL ######################

- ### I####################
- ### II###########SQL## PostgreSQL ##############
- ### III######################### PostgreSQL################################## ##############
- ### IV#PostgreSQL#####################################
- ### V############################## ################################
- ### VI##SQL## ########################## ##################
- ### VII##PostgreSQL########################

1. PostgreSQL##?

PostgreSQL####################################POSTGRES, Version 4.2[1]############ ###################ORDBMS#### POSTGRES################################## ###########

PostgreSQL##SQL######################## ##

- complex queries
- foreign keys
- triggers
- updatable views
- transactional integrity
- multiversion concurrency control

##PostgreSQL##

- data types
- functions
- operators
- aggregate functions
- index methods
- procedural languages

##################PostgreSQL############# ##############################

2. PostgreSQL##

[1] https://dsf.berkeley.edu/postgres.html

##PostgreSQL##POSTGRES ############## ################PostgreSQL#####################################

2.1. ######POSTGRES######

Michael Stonebraker#####POSTGRES################Defense Advanced Research Projects Agency#DARPA## Army Research Office (ARO)#National Science Foundation#NSF#####ESL, Inc.########## POSTGRES####1986######### ##############[ston86]################ ##[rowe87]######## ##############[ston87a]######## ##############[ston87b]######## ##

POSTGRES########################## ###############1987#########1988##ACM-SIGMOD################ [ston90a]############1##1989#6########################## ############[ston89]################[ston90b]########2#1990#6#################### ###### #####3#1991## ## Postgres95##

POSTGRES## ############################ ##GIS######## POSTGRES########## ################ ####Illustra Information Technologies##### Informix[2]###########IBM[3]## ################## POSTGRES#1992########Sequoia 2000 scientific computing project[4]### ################

1993########################## ## ########### ######################POSTGRES############4.2##############

2.2. Postgres95

1994##Andrew Yu#Jolly Chen#POSTGRES#SQL################ #####Postgres95#Web##### ##########POSTGRES##

Postgres95#######ANSI C############################25%################# ######## ################## Postgres95####1.0.x##POSTGRES#####4.2###Wisconsin Benchmark##30# 50%######## #######################

- ##########SQL#######PostQUEL########## #############libpq#PostQUEL######## ###### PostgreSQL##################################Postgres95#######SQL######## #### ########### `GROUP BY`###################
- GNU#Readline#############psql######SQL################# #####monitor########## ###########
- ################`libpgtcl`#Tcl################### #######`pgtclsh`#Tcl#Postgres95 ################Tcl############
- ############################## ############################## ################ #####
- ########################## ########################
- ##SQL####Postgres95##################################
- GNU make##BSD make################ ###Postgres95############GCC########## #double######################

[2] https://www.ibm.com/analytics/informix
[3] https://www.ibm.com/
[4] http://meteora.ucsd.edu/s2k/s2k_home.html

2.3. PostgreSQL

1996######Postgres95################################ ##########POSTGRES#SQL#
#########################PostgreSQL############## ############POSTGRES######
############6.0##################

######################PostgreSQL##Postgres############ ###################### #
###########################

Postgres95## PostgreSQL################
####################################

###PostgreSQL############E##########

3.

####################### ####[###]################# ####{###}#######|######1###
################## ###...###################### ############################
##

############SQL#######=>##############$###### ###################

################################## ######PostgreSQL###########################
######### ######################### ################################

4.

###############PostgreSQL##################

Wiki

PostgreSQL wiki[5]#######FAQ[6]###########TODO[7]##########################

Web###

PostgreSQL#Web###[8]####################PostgreSQL#########################
##

#######

###PostgreSQL#Web
############

######

PostgreSQL################ ############################### PostgreSQL######
#####
################## ###################################### ##################
##########

5.

PostgreSQL########################## ############################PostgreSQL##
############################PostgreSQL##########################

[5] https://wiki.postgresql.org
[6] https://wiki.postgresql.org/wiki/Frequently_Asked_Questions
[7] https://wiki.postgresql.org/wiki/Todo
[8] https://www.postgresql.org

##############################
####

################################## ##
################# ### #####
#
##############################

5.1.

####################
####### ############## ###################################### ###############
######################

- ## #####disk full##
- ##############################
- ##
- ## ####################################
  ########################
- ##################PostgreSQL############################

##

##################################
################## ##SQL##

#######TODO####FAQ########################### ##TODO####################
########## #####TODO##################

5.2.

####################################
#################### ## ############
################################
##&################################
##

################################

- ## ##########################SELECT###
  ################CREATE TABLE#INSERT#################### ################
  ##

  SQL##################psql################################### #~/.psqlrc##
  ############################ ################pg_dump########################
  ############################## ##################################### ########
  ################

  #########PHP## ##############
  Web########################## ## #######
  ### ####################

- ########## ## ######################
  ################ ## ###

#################### ### ############################

##
psql#######\set VERBOSITY verbose######### ##################################### #log_error_verbosity########verbose#########

##
################################

- ############################## ## ############################## ###### SQL###########/Oracle########################### SQL##PostgreSQL################### ######################################
- ### ################### OS# ## ###
- ##########################
- PostgreSQL####### SELECT version();######################## ############# #--version############ #####postgres --version#psql --version######## ## RPM## Git##

  17beta3################################## ################################ ######## #PostgreSQL## ##PostgreSQL## ##

- ########### ############C################################## ##Debian#####################################x86_64################################## ################################make###############

##[9]##

################

########################### ################PostgreSQL###############Postgres##### ################################PostgreSQL######################## 1##postgres################################## 1## ################ ####psql## ##############################

[9] https://www.chiark.greenend.org.uk/~sgtatham/bugs.html

5.3.

###########<pgsql-bugs@lists.postgresql.org>######################### ### ####################################

################Web###[10]########################## ############ ##<pgsql-bugs@lists.postgresql.org>###############

##pgsql-bugs############ ######### <security@postgresql.org>#######

<pgsql-sql@lists.postgresql.org>#<pgsql-general@lists.postgresql.org># ################################## ### ################ ##

#########<pgsql-hackers@lists.postgresql.org>################ ### PostgreSQL#################################### ############################ #pgsql-hackers############

#################################<pgsql-docs@lists.postgresql.org>##### ##### ###########################

#######################################
#<pgsql-hackers@lists.postgresql.org>######### ##########PostgreSQL##### #####################

> ##
>
> ## ###E########### ################# ##################https://lists.postgresql.org/############### ###

[10] https://www.postgresql.org/

I.

#####[68.528864]#PostgreSQL############## ################PostgreSQL##################SQL### ##################### ##################################### Unix###################### ###### PostgreSQL################################### ################################### ###### PostgreSQL############## ###########PostgreSQL####################SQL############### ################################ Unix ################### ###########PostgreSQL################################ ############################

#####[69.367089]###################SQL########################## II##PostgreSQL############ ##################### IV############ ########################## III########## ######## ##################### II#########SQL############PostgreSQL########################### ### PostgreSQL######################## III#########

##

1. ######## 3
    1.1. ###### 3
    1.2. ######## 3
    1.3. ######### 3
    1.4. ############ 5
2. SQL## 7
    2.1. #### 7
    2.2. ## 7
    2.3. ########## 7
    2.4. ######### 8
    2.5. ########### 9
    2.6. ######## 10
    2.7. #### 12
    2.8. ## 14
    2.9. ## 14
3. ###### 15
    3.1. #### 15
    3.2. ### 15
    3.3. #### 15
    3.4. ######## 16
    3.5. ####### 17
    3.6. ## 19
    3.7. ### 21

#1#

1.1.

PostgreSQL#################################### ################################ #######################PostgreSQL########################### ################## #############PostgreSQL#########################

PostgreSQL## ################ ###### ###### PostgreSQL#########################root#########################

###PostgreSQL#######################17########### ############################## # ##################################

##PGHOST################################ ###PGPORT######################## ## ## ##################################

1.2.

#######PostgreSQL################## PostgreSQL################################## ############

#############PostgreSQL#######/############### PostgreSQL######################## ############

- ######## ## ########## ##############postgres########
- ## ################################ ##### ##################################Web########Web############################## ### ####################PostgreSQL## ####

######/## ################TCP/IP## ################ ############### ## ##

PostgreSQL############################### ###############################fork#### ## ###########################postgres###################### ################## ## ################## ############################

1.3.

####PostgreSQL################ #####

#################################### ###########################

################################### #####mydb########

```
$ createdb mydb
```

##

##################PostgreSQL##################

```
createdb: command not found
```

PostgreSQL#################################### ##########################

```
$ /usr/local/pgsql/bin/createdb mydb
```

###################### ###

##############

```
createdb: error: connection to server on socket "/tmp/.s.PGSQL.5432"
 failed: No such file or directory
        Is the server running locally and accepting connections on that
 socket?
```

################createdb############################## ####################
##########

##############

```
createdb: error: connection to server on socket "/tmp/.s.PGSQL.5432"
 failed: FATAL:  role "joe" does not exist
```

###joe########################### ###############PostgreSQL####################
#PostgreSQL#################################### ######################21###
####### ####################PostgreSQL############################postgres#
########## PostgreSQL## ####
##PostgreSQL#############-U##########PGUSER################

###

```
createdb: error: database creation failed: ERROR:  permission denied to
 create database
```

PostgreSQL############################
######################### ######################### ###PostgreSQL##############
[1]

######################## PostgreSQL#################################### #######
####################63################## ################################## ##
#######################
####

```
$ createdb
```

######################## ####mydb####################################

```
$ dropdb mydb
```

####################
############################## #######################

createdb#dropdb########createdb#dropdb######

[1] ##################PostgreSQL#################################### ####################PostgreSQL#########
################################PostgreSQL################
####### ############################## ################-U################PostgreSQL############

1.4.

##########################

- *psql*###PostgreSQL############# ##########SQL#######################
- pgAdmin#########################ODBC####JDBC############################## ###### #########################
- ##########1#################### ########## IV#########

################psql############ ################mydb################

```
$ psql mydb
```

############createdb###########

psql####################

```
psql (17beta3)
Type "help" for help.

mydb=>
```

#######################

```
mydb=#
```

########################### ###PostgreSQL#################################### ################################### ##############################

psql######################## createdb####psql############################### ###

psql################psql############psql#########SQL################### # ##############

```
mydb=> SELECT version();
                                          version
-----------------------------------------------------------------
-----------------------
 PostgreSQL 17beta3 on x86_64-pc-linux-gnu, compiled by gcc (Debian
 4.9.2-10) 4.9.2, 64-bit
(1 row)

mydb=> SELECT current_date;
    date
------------
 2016-01-07
(1 row)

mydb=> SELECT 2 + 2;
 ?column?
----------
        4
(1 row)
```

psql######SQL######################### ##############\######## ###### PostgreSQL SQL################################

```
mydb=> \h
```

psql################

```
mydb=> \q
```

psql################ #############psql######\?########## psql#########
#psql######### ##

#2# SQL##

2.1.

#####SQL########################## ###################SQL##################
[melt93]#[date97]###SQL############## PostgreSQL################################

################mydb################psql#########################

############PostgreSQL#############src/tutorial/######### #PostgreSQL######
########################## ######################################make########
##

```
$ cd .../src/tutorial
$ make
```

################################C############## ##############################
##

```
$ psql -s mydb

...

mydb=> \i basics.sql
```

\i######################## psql#-s##
###########basics.sql##########

2.2.

PostgreSQL###################RDBMS#### ################################
######### ######################### ##
###################################### Unix###################################
######### #########################

#################### ############################## ################ #####
#############SQL################################### ########################
###

##################1##PostgreSQL#######################################

2.3.

##

```
CREATE TABLE weather (
    city            varchar(80),

    temp_lo         int,           -- ####
    temp_hi         int,           -- ####
    prcp            real,          -- ###
    date            date
);
```

##############psql####### psql##############################

SQL################################ ################################### ###1# ############# ####2########--############# ###################### SQL######### ################## ##

varchar(80)##80############################ int############ real########## ######### date################## ##########date#######date######## ############ #################### ############

PostgreSQL###SQL#####int#smallint#real#double precision#char(*N*)#varchar(*N*)#date#time#timestamp#interval######## #### ############################## ####################PostgreSQL########### #### ####SQL###

#####2##########################

```
CREATE TABLE cities (
    name            varchar(80),
    location        point
);
```

point###PostgreSQL############

###

```
DROP TABLE tablename;
```

2.4.

#######INSERT###################

```
INSERT INTO weather VALUES ('San Francisco', 46, 50, 0.25, '1994-11-27');
```

############################'##### ######### date######################

point###########################

```
INSERT INTO cities VALUES ('San Francisco', '(-194.0, 53.0)');
```

############################## ################################

```
INSERT INTO weather (city, temp_lo, temp_hi, prcp, date)
    VALUES ('San Francisco', 43, 57, 0.0, '1994-11-29');
```

################### ################### ##########################

```
INSERT INTO weather (date, city, temp_hi, temp_lo)
    VALUES ('1994-11-29', 'Hayward', 54, 37);
```

###

##################################

###COPY################################### COPY#####INSERT################## ######################## ##########

```
COPY weather FROM '/home/user/weather.txt';
```

COPY##COPY####################

2.5.

  ################################ #####SQL#SELECT######## ################## ## ####weather# ####################

```
SELECT * FROM weather;
```

###*############## [1] ########################

```
SELECT city, temp_lo, temp_hi, prcp, date FROM weather;
```

###############

```
     city      | temp_lo | temp_hi | prcp |    date
---------------+---------+---------+------+------------
 San Francisco |      46 |      50 | 0.25 | 1994-11-27
 San Francisco |      43 |      57 |    0 | 1994-11-29
 Hayward       |      37 |      54 |      | 1994-11-29
(3 rows)
```

################################## #################

```
SELECT city, (temp_hi+temp_lo)/2 AS temp_avg, date FROM weather;
```

###############

```
     city      | temp_avg |    date
---------------+----------+------------
 San Francisco |       48 | 1994-11-27
 San Francisco |       50 | 1994-11-29
 Hayward       |       45 | 1994-11-29
(3 rows)
```

AS############################ #AS##########

############WHERE###################### WHERE################################ ############AND#OR#NOT############## #######San Francisco#################

```
SELECT * FROM weather
    WHERE city = 'San Francisco' AND prcp > 0.0;
```

#############

```
     city      | temp_lo | temp_hi | prcp |    date
---------------+---------+---------+------+------------
 San Francisco |      46 |      50 | 0.25 | 1994-11-27
(1 row)
```

 ##########################

[1] SELECT *##

```
SELECT * FROM weather
    ORDER BY city;
```

```
     city      | temp_lo | temp_hi | prcp |    date
---------------+---------+---------+------+------------
 Hayward       |      37 |      54 |      | 1994-11-29
 San Francisco |      43 |      57 |    0 | 1994-11-29
 San Francisco |      46 |      50 | 0.25 | 1994-11-27
```

####################### #####San Francisco################ ####################### ###

```
SELECT * FROM weather
    ORDER BY city, temp_lo;
```

 ########################

```
SELECT DISTINCT city
    FROM weather;
```

```
     city
---------------
 Hayward
 San Francisco
(2 rows)
```

####################### DISTINCT#ORDER BY############################### [2]

```
SELECT DISTINCT city
    FROM weather
    ORDER BY city;
```

2.6.

###############1#################### ### ##########1#################### ### ########## ####1##########2### ######## ####################################weather#######city###cities######## #name############# [3] ########################

```
SELECT * FROM weather JOIN cities ON city = name;
```

```
     city      | temp_lo | temp_hi | prcp |    date    |     name      |
 location
---------------+---------+---------+------+------------+---------------
+-----------
 San Francisco |      46 |      50 | 0.25 | 1994-11-27 | San Francisco |
 (-194,53)
 San Francisco |      43 |      57 |    0 | 1994-11-29 | San Francisco |
 (-194,53)
(2 rows)
```

########2##############

[2] PostgreSQL#########################DISTINCT##################ORDER BY###### ########SQL############# PostgreSQL##DISTINCT####################

[3] ################ ########1################################ ##################

- Hayward############### ###cities######Hayward##################weather#### ################### ##########################
- ######2######## weather#####cities############################## ######## ###############*##############################

```
SELECT city, temp_lo, temp_hi, prcp, date, location
    FROM weather JOIN cities ON city = name;
```

2##############################

```
SELECT weather.city, weather.temp_lo, weather.temp_hi,
       weather.prcp, weather.date, cities.location
    FROM weather JOIN cities ON weather.city = cities.name;
```

###########

#####################################

```
SELECT *
    FROM weather, cities
    WHERE city = name;
```

#####JOIN/ON########SQL-92######## #####FROM############WHERE####### ##### ###########JOIN/ON########## ### #### ######################################WHERE############

###########Hayward############################## ###########weather######### ####cities################## ############cities############################## ################### ################### ###############

```
SELECT *
    FROM weather LEFT OUTER JOIN cities ON weather.city = cities.name;
```

```
     city      | temp_lo | temp_hi | prcp |    date    |     name      |
 location
---------------+---------+---------+------+------------+---------------
+-----------
 Hayward       |      37 |      54 |      | 1994-11-29 |               |
 San Francisco |      46 |      50 | 0.25 | 1994-11-27 | San Francisco |
 (-194,53)
 San Francisco |      43 |      57 |    0 | 1994-11-29 | San Francisco |
 (-194,53)
(3 rows)
```

################### ## ############# ##NULL########

##: ################## #################

######################### ############# ##################################### ## weather###temp_lo#temp_hi####weather##temp_lo#temp_hi############### #######################

```
SELECT w1.city, w1.temp_lo AS low, w1.temp_hi AS high,
```

```
       w2.city, w2.temp_lo AS low, w2.temp_hi AS high
    FROM weather w1 JOIN weather w2
        ON w1.temp_lo < w2.temp_lo AND w1.temp_hi > w2.temp_hi;
```

```
     city      | low | high |     city      | low | high
---------------+-----+------+---------------+-----+------
 San Francisco |  43 |   57 | San Francisco |  46 |   50
 Hayward       |  37 |   54 | San Francisco |  46 |   50
(2 rows)
```

##################### weather#####w1#w2############## ########################## ######## #########

```
SELECT *
    FROM weather w JOIN cities c ON w.city = c.name;
```

####################

2.7.

######################### PostgreSQL############# ############1########### ###### ######### count#####sum#####avg#####max#####min#################### ######################################

```
SELECT max(temp_lo) FROM weather;
```

```
 max
-----
  46
(1 row)
```

 ##

```
SELECT city FROM weather WHERE temp_lo = max(temp_lo);     ###
```

####max###WHERE############################ #WHERE######################### ################################### ############### ######################### ############### ######################

```
SELECT city FROM weather
    WHERE temp_lo = (SELECT max(temp_lo) FROM weather);
```

```
     city
---------------
 San Francisco
(1 row)
```

##

 ###GROUP BY#################### ##

```
SELECT city, count(*), max(temp_lo)
    FROM weather
    GROUP BY city;
```

```
     city      | count | max
---------------+-------+-----
 Hayward       |     1 |  37
 San Francisco |     2 |  46
(2 rows)
```

#########1######### ################################### #######HAVING#######
####################

```
SELECT city, count(*), max(temp_lo)
    FROM weather
    GROUP BY city
    HAVING max(temp_lo) < 40;
```

```
  city   | count | max
---------+-------+-----
 Hayward |     1 |  37
(1 row)
```

#########################temp_lo###40############# #########S###############
###############

```
SELECT city, count(*), max(temp_lo)
    FROM weather
    WHERE city LIKE 'S%'            -- 1
    GROUP BY city;
```

```
     city      | count | max
---------------+-------+-----
 San Francisco |     2 |  46
(1 row)
```

1 LIKE#######################9.7#######

###SQL#WHERE#HAVING#################### WHERE#HAVING#########WHERE#######
###HAVING################
#################### ######WHERE################# ##########################
############# ####HAVING############ ################HAVING################
############# #####WHERE##################

#####WHERE################ ############### ###HAVING################ ###
#WHERE####################################

#################1###################FILTER##########

```
SELECT city, count(*) FILTER (WHERE temp_lo < 45), max(temp_lo)
    FROM weather
    GROUP BY city;
```

```
     city      | count | max
---------------+-------+-----
 Hayward       |     1 |  37
 San Francisco |     1 |  46
(2 rows)
```

FILTER#WHERE## #####count##
#temp_lo#45#############max#####################46#######

2.8.

UPDATE#################### 11#28##############2################### ######### ###################

```
UPDATE weather
    SET temp_hi = temp_hi - 2,  temp_lo = temp_lo - 2
    WHERE date > '1994-11-28';
```

################

```
SELECT * FROM weather;

     city      | temp_lo | temp_hi | prcp |    date
---------------+---------+---------+------+------------
 San Francisco |      46 |      50 | 0.25 | 1994-11-27
 San Francisco |      41 |      55 |    0 | 1994-11-29
 Hayward       |      35 |      52 |      | 1994-11-29
(3 rows)
```

2.9.

DELETE######################## Hayward################## ######################## ##########

```
DELETE FROM weather WHERE city = 'Hayward';
```

Hayward####################

```
SELECT * FROM weather;

     city      | temp_lo | temp_hi | prcp |    date
---------------+---------+---------+------+------------
 San Francisco |      46 |      50 | 0.25 | 1994-11-27
 San Francisco |      41 |      55 |    0 | 1994-11-29
(2 rows)
```

######################

```
DELETE FROM tablename;
```

########DELETE############################## ##########################

#3#

3.1.

#####PostgreSQL#SQL################################## ######################
##SQL##################### ###PostgreSQL#############

#######2#################################### ############tutorial######
#advanced.sql## ######
####2.1##########

3.2.

2.6################### ##
################################## ##
#########

```
CREATE VIEW myview AS
    SELECT name, temp_lo, temp_hi, prcp, date, location
        FROM weather, cities
        WHERE city = name;

SELECT * FROM myview;
```

###############SQL###################### ####################################
##################################

########################## #########################

3.3.

#2##weather#####cities############## ################# cities#############
#####weather################## ################### ########################
####cities##########################weather##############################
####### ################################PostgreSQL#################

#######################

```
CREATE TABLE cities (
        name     varchar(80) primary key,
        location point
);

CREATE TABLE weather (
        city      varchar(80) references cities(name),
        temp_lo   int,
        temp_hi   int,
        prcp      real,
        date      date
);
```

###################

```
INSERT INTO weather VALUES ('Berkeley', 45, 53, 0.0, '1994-11-28');
```

```
ERROR:  insert or update on table "weather" violates foreign key
 constraint "weather_city_fkey"
DETAIL:  Key (city)=(Berkeley) is not present in table "cities".
```

############################ ##5#########
##

3.4.

################################ ##
################################## ##
################

##############$100.00###
################ ###############SQL###########

```
UPDATE accounts SET balance = balance - 100.00
    WHERE name = 'Alice';
UPDATE branches SET balance = balance - 100.00
    WHERE name = (SELECT branch_name FROM accounts WHERE name = 'Alice');
UPDATE accounts SET balance = balance + 100.00
    WHERE name = 'Bob';
UPDATE branches SET balance = balance + 100.00
    WHERE name = (SELECT branch_name FROM accounts WHERE name = 'Bob');
```

######SQL########################### ##
$100.00################
################################## ###$100.00#################################
################### ## #####
############################## ##
#########

###
############# ##
############ ##
##################################

############################
################################
###
##
####### ## ##################
##############################

PostgreSQL###############SQL#####BEGIN#COMMIT######### ####################
###########

```
BEGIN;
UPDATE accounts SET balance = balance - 100.00
    WHERE name = 'Alice';

-- ##
COMMIT;
```

###COMMIT####ROLLBACK#####
###############

PostgreSQL######SQL######################## BEGIN###################BEGIN##
################COMMIT############## BEGIN#COMMIT##########################
#########

##

####################BEGIN#COMMIT#################################### ## ############################

###########################
############################## SAVEPOINT#######################ROLLBACK TO#
######################### ##
#######################

###################
###################################### ######################################
#####################################

############################
###########1##

############################$100.00##
########### ####################


```
BEGIN;
UPDATE accounts SET balance = balance - 100.00
    WHERE name = 'Alice';
SAVEPOINT my_savepoint;
UPDATE accounts SET balance = balance + 100.00
    WHERE name = 'Bob';

-- ############################
ROLLBACK TO my_savepoint;
UPDATE accounts SET balance = balance + 100.00
    WHERE name = 'Wally';
COMMIT;
```

#############
###ROLLBACK TO##############
##################

3.5.

#######
###
##

##


```
SELECT depname, empno, salary, avg(salary) OVER (PARTITION BY depname)
 FROM empsalary;
```


```
  depname  | empno | salary |          avg
-----------+-------+--------+----------------------
 develop   |    11 |   5200 | 5020.0000000000000000
 develop   |     7 |   4200 | 5020.0000000000000000
 develop   |     9 |   4500 | 5020.0000000000000000
 develop   |     8 |   6000 | 5020.0000000000000000
 develop   |    10 |   5200 | 5020.0000000000000000
 personnel |     5 |   3500 | 3700.0000000000000000
```

```
 personnel |     2 |   3900 | 3700.0000000000000000
 sales     |     3 |   4800 | 4866.6666666666666667
 sales     |     1 |   5000 | 4866.6666666666666667
 sales     |     4 |   4800 | 4866.6666666666666667
(10 rows)
```

###3##########empsalary##################1###########4###########
#depname################################ #############avg##########OVER####
################################

################################OVER####### ################################
OVER## OVER##PARTITION BY####
#PARTITION BY################################ ##############################
################

OVER##ORDER BY#################################### #######ORDER BY#######
################## ##########

```
SELECT depname, empno, salary,
       rank() OVER (PARTITION BY depname ORDER BY salary DESC)
FROM empsalary;
```

```
  depname  | empno | salary | rank
-----------+-------+--------+------
 develop   |     8 |   6000 |    1
 develop   |    10 |   5200 |    2
 develop   |    11 |   5200 |    2
 develop   |     9 |   4500 |    4
 develop   |     7 |   4200 |    5
 personnel |     2 |   3900 |    1
 personnel |     5 |   3500 |    2
 sales     |     1 |   5000 |    1
 sales     |     4 |   4800 |    2
 sales     |     3 |   4800 |    2
(10 rows)
```

##########rank###########ORDER BY#########################ORDER BY#######
########## rank######################OVER################

#################WHERE#GROUP BY####HAVING##################FROM########
############# ####WHERE############################## ####OVER##############
################################## ##################################

ORDER BY################################ PARTITION BY########## ##########
##################

######################### ### ####
########ORDER BY###########
#####################################ORDER BY##############################
ORDER BY### [1] sum###########

```
SELECT salary, sum(salary) OVER () FROM empsalary;
```

```
 salary |  sum
--------+-------
   5200 | 47100
   5000 | 47100
   3500 | 47100
```

[1] ##4.2.8#########

```
   4800 | 47100
   3900 | 47100
   4200 | 47100
   4500 | 47100
   4800 | 47100
   6000 | 47100
   5200 | 47100
(10 rows)
```

####OVER###ORDER BY##PARTITION BY## ## ######## ##ORDER BY###################

```
SELECT salary, sum(salary) OVER (ORDER BY salary) FROM empsalary;
```

```
 salary |  sum
--------+-------
   3500 |  3500
   3900 |  7400
   4200 | 11600
   4500 | 16100
   4800 | 25700
   4800 | 25700
   5000 | 30700
   5200 | 41100
   5200 | 41100
   6000 | 47100
(10 rows)
```

####sum##########salary#####################################salary############## ###

#############SELECT####ORDER BY########### GROUP BY#HAVING####WHERE##### ############## ## ################ ############# ##

##SELECT####### #######

```
SELECT depname, empno, salary, enroll_date
FROM
  (SELECT depname, empno, salary, enroll_date,
          rank() OVER (PARTITION BY depname ORDER BY salary DESC, empno)
 AS pos
     FROM empsalary
  ) AS ss
WHERE pos < 3;
```

########3#####rank#######################

##################################OVER## ############## ####WINDOW####################OVER######### ##########

```
SELECT sum(salary) OVER w, avg(salary) OVER w
  FROM empsalary
  WINDOW w AS (PARTITION BY depname ORDER BY salary DESC);
```

################ 4.2.8# 9.22# 7.2.5#### SELECT ##############

3.6.

###################### ##################################

2######cities########capitals################ ##########################
########################## ######################


```
CREATE TABLE capitals (
  name       text,
  population real,

  elevation  int,    -- ########
  state      char(2)
);

CREATE TABLE non_capitals (
  name       text,
  population real,

  elevation  int     -- ########
);

CREATE VIEW cities AS
  SELECT name, population, elevation FROM capitals
    UNION
  SELECT name, population, elevation FROM non_capitals;
```

##

#################


```
CREATE TABLE cities (
  name       text,
  population real,

  elevation  int     -- ########
);

CREATE TABLE capitals (
  state      char(2) UNIQUE NOT NULL
) INHERITS (cities);
```

######capitals########cities##############name######population######
#elevation########## name################PostgreSQL#########text####
capitals##################state###### PostgreSQL#########################
##########

####################500#######################


```
SELECT name, elevation
  FROM cities
  WHERE elevation > 500;
```

##########


```
   name    | elevation
-----------+-----------
 Las Vegas |      2174
 Mariposa  |      1953
 Madison   |       845
(3 rows)
```

############500######################################

```
SELECT name, elevation
    FROM ONLY cities
    WHERE elevation > 500;
```

```
   name    | elevation
-----------+-----------
 Las Vegas |      2174
 Mariposa  |      1953
(2 rows)
```

###cities######ONLY#########cities##################cities##############
######SELECT#UPDATE###DELETE#############ONLY############

> **##**
>
> ## ###5.11##########

3.7.

SQL################################PostgreSQL####### ######################

##################PostgreSQL#Web###[2]####################

[2] https://www.postgresql.org

II. SQL##

#####[72.811060]######PostgreSQL#SQL############### #####SQL## ##########SQL######################### ### ############PostgreSQL##SQL############### ###SQL## ######SQL########################### ####################################

#####[80.082136]## ## ### VI########## ## ## ################### ################### ###### VI##########

#####[89.782609]#######PostgreSQL############SQL##################### ################### ######## I################ SQL######PostgreSQL#####psql################################### ####### #########PostgreSQL##########SQL######################### ################### I### ########## SQL########PostgreSQL#########psql##################################

##

4. SQL### 30
4.1. ##### 30
4.1.1. ######### 30
4.1.2. ## 31
4.1.3. ### 36
4.1.4. #### 36
4.1.5. #### 36
4.1.6. ######## 37
4.2. ### 38
4.2.1. #### 38
4.2.2. ####### 38
4.2.3. ## 39
4.2.4. ####### 39
4.2.5. ######## 40
4.2.6. ###### 40
4.2.7. ### 40
4.2.8. ########### 42
4.2.9. ##### 44
4.2.10. ##### 45
4.2.11. ######### 45
4.2.12. ######### 46
4.2.13. ######## 47
4.2.14. ###### 48
4.3. ###### 49
4.3.1. ####### 49
4.3.2. ######### 50
4.3.3. ####### 50
5. ##### 52
5.1. ####### 52
5.2. ###### 53
5.3. Identity Columns 53
5.4. ### 54
5.5. ## 56
5.5.1. #### 56
5.5.2. #NULL## 57
5.5.3. ##### 58
5.5.4. ### 59
5.5.5. #### 60
5.5.6. #### 63
5.6. ##### 63
5.7. ####### 64
5.7.1. #### 64
5.7.2. #### 65
5.7.3. ##### 65
5.7.4. ##### 65
5.7.5. ########### 65
5.7.6. ######### 66
5.7.7. ##### 66
5.7.8. ######## 66
5.8. ## 66
5.9. ########### 70
5.10. #### 76
5.10.1. ####### 76
5.10.2. public#### 77
5.10.3. ######## 77

5.10.4. ######## 78
5.10.5. ########### 79
5.10.6. ###### 79
5.10.7. ### 79
5.11. ## 80
5.11.1. ## 82
5.12. ############## 82
5.12.1. ## 82
5.12.2. ########### 83
5.12.3. ############### 88
5.12.4. ######## 92
5.12.5. ################# 93
5.12.6. ###################### 93
5.13. ##### 94
5.14. ################ 94
5.15. ####### 94
6. ##### 97
6.1. ###### 97
6.2. ###### 98
6.3. ###### 98
6.4. ############# 99
7. ##### 100
7.1. ## 100
7.2. ##### 100
7.2.1. FROM# 100
7.2.2. WHERE# 108
7.2.3. GROUP BY##HAVING# 109
7.2.4. GROUPING SETS#CUBE#ROLLUP 110
7.2.5. ######## 113
7.3. ##### 113
7.3.1. ####### 113
7.3.2. #### 114
7.3.3. DISTINCT 114
7.4. #######(UNION, INTERSECT, EXCEPT) 115
7.5. ######(ORDER BY) 116
7.6. LIMIT#OFFSET 116
7.7. VALUES### 117
7.8. WITH############## 118
7.8.1. WITH##SELECT 118
7.8.2. ######## 118
7.8.3. ################### 122
7.8.4. WITH######## 123
8. #### 126
8.1. ###### 127
8.1.1. ###### 128
8.1.2. ######## 128
8.1.3. ######## 130
8.1.4. ### 131
8.2. ### 132
8.3. ### 132
8.4. ######### 134
8.4.1. bytea#hex## 134
8.4.2. bytea######## 135
8.5. ##/###### 136
8.5.1. ##/##### 137
8.5.2. ##/##### 140
8.5.3. ### 141
8.5.4. ####### 142

8.5.5. ###### 143
8.6. ###### 144
8.7. ### 145
8.7.1. ##### 145
8.7.2. ## 145
8.7.3. #### 146
8.7.4. #### 146
8.8. ##### 147
8.8.1. ### 147
8.8.2. ## 147
8.8.3. ## 147
8.8.4. ## 148
8.8.5. ## 148
8.8.6. ###(####) 148
8.8.7. # 149
8.9. ########## 149
8.9.1. `inet` 149
8.9.2. `cidr` 149
8.9.3. `inet#cidr`###### 150
8.9.4. `macaddr` 150
8.9.5. `macaddr8` 150
8.10. ####### 151
8.11. ########## 152
8.11.1. `tsvector` 152
8.11.2. `tsquery` 153
8.12. UUID# 154
8.13. XML# 155
8.13.1. XML#### 155
8.13.2. ######## 156
8.13.3. XML###### 156
8.14. JSON#### 156
8.14.1. JSON##### 157
8.14.2. JSON######## 158
8.14.3. `jsonb`########### 158
8.14.4. `jsonb` ##### 160
8.14.5. `jsonb`### 162
8.14.6. ## 164
8.14.7. jsonpath# 164
8.15. ## 165
8.15.1. ##### 165
8.15.2. ###### 166
8.15.3. ####### 167
8.15.4. #### 169
8.15.5. ##### 171
8.15.6. ####### 172
8.16. ### 173
8.16.1. ##### 173
8.16.2. ####### 174
8.16.3. ######## 175
8.16.4. ##### 175
8.16.5. ########### 176
8.16.6. ######## 177
8.17. ### 178
8.17.1. ########### 178
8.17.2. # 179
8.17.3. ########## 179
8.17.4. ###(#####)## 179
8.17.5. ##### 180

8.17.6. ############## ... 181
8.17.7. ###### ... 181
8.17.8. ######## ... 181
8.17.9. ##### ... 182
8.17.10. #### ... 182
8.18. #### ... 183
8.19. ########### ... 184
8.20. pg_lsn# ... 186
8.21. ##### ... 186
9. ##### ... 188
9.1. #### ... 188
9.2. ######## ... 188
9.3. ####### ... 192
9.4. ######## ... 200
9.4.1. format ... 207
9.5. ########### ... 209
9.6. ########### ... 212
9.7. ###### ... 214
9.7.1. LIKE ... 215
9.7.2. SIMILAR TO#### ... 216
9.7.3. POSIX#### ... 217
9.8. ######### ... 230
9.9. ##/######## ... 237
9.9.1. EXTRACT, date_part ... 243
9.9.2. date_trunc ... 247
9.9.3. date_bin ... 248
9.9.4. AT TIME ZONE and AT LOCAL ... 248
9.9.5. ####/## ... 250
9.9.6. #### ... 251
9.10. ######## ... 252
9.11. ####### ... 253
9.12. ############### ... 259
9.13. ########### ... 262
9.14. UUID## ... 268
9.15. XML## ... 268
9.15.1. XML#### ... 268
9.15.2. XML## ... 272
9.15.3. XML### ... 274
9.15.4. XML######## ... 277
9.16. JSON##### ... 281
9.16.1. JSON######## ... 281
9.16.2. SQL/JSON#### ... 292
9.16.3. SQL/JSON Query Functions ... 303
9.16.4. JSON_TABLE ... 305
9.17. ######## ... 310
9.18. ### ... 311
9.18.1. CASE ... 311
9.18.2. COALESCE ... 312
9.18.3. NULLIF ... 313
9.18.4. GREATEST###LEAST ... 313
9.19. ####### ... 313
9.20. ########### ... 316
9.21. #### ... 322
9.22. ###### ... 329
9.23. Merge Support Functions ... 330
9.24. ###### ... 330
9.24.1. EXISTS ... 331
9.24.2. IN ... 331

9.24.3. NOT IN 331
9.24.4. ANY/SOME 332
9.24.5. ALL 332
9.24.6. ######## 333
9.25. ###### 333
9.25.1. IN 333
9.25.2. NOT IN 333
9.25.3. ANY/SOME (##) 334
9.25.4. ALL (##) 334
9.25.5. ########## 334
9.25.6. ##### 335
9.26. ###### 335
9.27. ########### 338
9.27.1. ######## 338
9.27.2. ######### 341
9.27.3. ############## 343
9.27.4. ########### 344
9.27.5. ################ 349
9.27.6. ####### 350
9.27.7. ######### 350
9.27.8. #######ID############ 351
9.27.9. ################## 353
9.27.10. ########## 353
9.27.11. Version Information Functions 355
9.27.12. WAL Summarization Information Functions 355
9.28. ####### 356
9.28.1. ##### 356
9.28.2. ########## 356
9.28.3. ######### 358
9.28.4. ####### 359
9.28.5. ########### 361
9.28.6. ########### 361
9.28.7. ################ 364
9.28.8. ######### 367
9.28.9. ########### 367
9.28.10. ######## 369
9.29. ##### 370
9.30. ######## 371
9.30.1. ############## 371
9.30.2. DDL################## 372
9.30.3. ############### 373
9.31. ##### 373
9.31.1. MCV##### 373
10. ### 375
10.1. ## 375
10.2. ### 376
10.3. ## 379
10.4. #### 382
10.5. UNION#CASE######## 382
10.6. SELECT### 384
11. ##### 385
11.1. #### 385
11.2. ######## 386
11.2.1. B-Tree 386
11.2.2. Hash 386
11.2.3. GiST 386
11.2.4. SP-GiST 387
11.2.5. GIN 387

11.2.6. BRIN 387
11.3. ######## 387
11.4. #######ORDER BY 388
11.5. ############### 389
11.6. ######## 389
11.7. ########### 390
11.8. ######## 390
11.9. ########################### 393
11.10. ########### 395
11.11. ########### 396
11.12. ############## 396
12. #### 398
12.1. #### 398
12.1.1. ######? 398
12.1.2. ########## 399
12.1.3. ## 401
12.2. ########### 401
12.2.1. ######## 401
12.2.2. ######## 402
12.3. ######## 403
12.3.1. ###### 403
12.3.2. ######## 404
12.3.3. ########## 406
12.3.4. ##### 408
12.4. #### 409
12.4.1. ##### 409
12.4.2. ########## 409
12.4.3. ########### 412
12.4.4. ########## 413
12.5. ### 413
12.6. ## 415
12.6.1. ####### 416
12.6.2. simple## 416
12.6.3. ##### 417
12.6.4. #### 419
12.6.5. Ispell## 421
12.6.6. Snowball## 422
12.7. ### 423
12.8. ############### 424
12.8.1. ###### 424
12.8.2. ####### 426
12.8.3. ###### 427
12.9. ################### 428
12.10. psql#### 429
12.11. #### 431
13. ###### 433
13.1. #### 433
13.2. ########### 433
13.2.1. ############## 434
13.2.2. ############## 435
13.2.3. ############# 436
13.3. ###### 438
13.3.1. ########## 438
13.3.2. ####### 440
13.3.3. ######## 441
13.3.4. ###### 441
13.3.5. ###### 442
13.4. ######################## 442

13.4.1. ########################## 443
13.4.2. ######################## 443
13.5. ######## 444
13.6. ## 444
13.7. ########## 444
14. ######### 446
14.1. EXPLAIN### 446
14.1.1. EXPLAIN### 446
14.1.2. EXPLAIN ANALYZE 453
14.1.3. ## 456
14.2. ############## 457
14.2.1. ####### 457
14.2.2. ###### 459
14.3. ####JOIN########### 461
14.4. ############# 463
14.4.1. ############ 463
14.4.2. COPY### 463
14.4.3. ########### 463
14.4.4. ######### 463
14.4.5. maintenance_work_mem#### 463
14.4.6. max_wal_size#### 464
14.4.7. WAL########################### 464
14.4.8. ###ANALYZE### 464
14.4.9. pg_dump########### 464
14.5. ######## 465
15. ####### 466
15.1. ################# 466
15.2. ######################## 466
15.3. ####### 467
15.3.1. ######## 467
15.3.2. ######## 468
15.3.3. ###### 468
15.3.4. ######## 468
15.3.5. ############## 469
15.4. ###### 469
15.4.1. ################## 469

#4# SQL###

####SQL############ ###################SQL################################
#######

####SQL#####################PostgreSQL#####################SQL##############
#################

4.1.

SQL################### ##############################;######## ############
############ #########################

###########################
##

##################SQL######

```
SELECT * FROM MY_TABLE;
UPDATE MY_TABLE SET A = 5;
INSERT INTO MY_TABLE VALUES (3, 'hi there');
```

####1##1###########3##################1#######1############ 1################
######1##########################

#########SQL################# #########################

SQL### ##################
##############SELECT###UPDATE###INSERT#####################
#UPDATE#######SET#####################################INSERT###########
#VALUES######### ################### VI#########

4.1.1.

#########SELECT#UPDATE#####VALUES################### ########SQL#########
######### MY_TABLE#####A############# ####################################
######### ######################## ##
#################### ##############C######

SQL#############a#z##############Latin###########_############## ##########
######################0#9#########$######## ##SQL#########################
######### ##################################### ##SQL#######################
####################### ######################################

#####NAMEDATALEN-1################ ##############################
######NAMEDATALEN#64##########63##### #############src/include/
pg_config_manual.h##NAMEDATALEN##################

################################ ######

```
UPDATE MY_TABLE SET A = 5;
```

#################

```
uPDaTE my_TabLE SeT a = 5;
```

#################################### ##############

```
UPDATE my_table SET a = 5;
```

################ ################### ################"######### ######### ################### #####"select"##select#################################### #select### ##################### ####

```
UPDATE "my_table" SET "a" = 5;
```

#############0##################### ####################2######## ############ ###&## ####################

######FOO#foo#"foo"# PostgreSQL##################"Foo"#"FOO"#####3#################### #PostgreSQL#########################SQL############SQL################### ########## ######SQL#####foo#"FOO"#########"foo"########### ##############

################################Unicode############ ######U&###########U#### ################################### ####U&"foo"###### ########&############## ####### ########################## ######Unicode################4#16######## #########1##############################6#16########################### ###### #"data"###############

```
U&"d\0061t\+000061"
```

################slon##################

```
U&"\0441\043B\043E\043D"
```

################################UESCAPE###################

```
U&"d!0061t!+000061" UESCAPE '!'
```

##########16## ########UESCAPE### ################################

############################2#########

U+FFFF#####################UTF-16################4##6#####################6# ################## ##

#########UTF-8##################1#######Unicode############################## ##################

4.1.2.

PostgreSQL###3################## ################## ########################### ########################### ####################

4.1.2.1.

SQL################'################ ####'This is a string'### ########## #######2############### ####'Dianne''s horse'### #####(")################

2#############*1*#######################2#############1#################### ### ####

```
SELECT 'foo'
'bar';
```

##

```
SELECT 'foobar';
```

#########

```
SELECT 'foo'      'bar';
```

############ ################SQL###########PostgreSQL###############

4.1.2.2. C##############

PostgreSQL######################### #####SQL###### ###########E################ ########################### ###E'foo'### ################################E#### ####### ########################\####C### ########### 4.1#####################

#4.1

#################	##
\b	##
\f	####
\n	##
\r	##
\t	##
o, *oo*, *ooo* (*o* = 0–7)	8######
\x*h*, \x*hh* (*h* = 0–9, A–F)	16######
\u*xxxx*, \U*xxxxxxxx* (*x* = 0–9, A–F)	16####32####16## Unicode ###

################################ ######################2###########\\######## ## ############################''####\'##############

##8###16###Unicode#######4.1.2.3#########Unicode###

> ##
>
> #######standard_conforming_strings# off####PostgreSQL######################### ################## ####PostgreSQL9.1########on########################### ############### #######SQL### ######################## ##############off################################ ############### ################################E##################
>
> standard_conforming_strings##########escape_string_warning## #backslash_quote#########################

#######################

4.1.2.3. Unicode##############

PostgreSQL################Unicode################################## Unicode######
######U&#########U## ####U&'foo'######
#########&#################### ########################## #######Unicode#######
##########4#16#################1##############################6#16############
################ #######'data'###########

```
U&'d\0061t\+000061'
```

#################slon##################

```
U&'\0441\043B\043E\043D'
```

################################UESCAPE####################

```
U&'d!0061t!+000061' UESCAPE '!'
```

##########16##

############################2##########

U+FFFF#####################UTF-16################4##6#######################6#
################### ##

#########UTF-8###################1#######Unicode################################
###################

################################standard_conforming_strings############## ##########
##SQL####################SQL## ######off##
################################

4.1.2.4.

###
############################PostgreSQL################################ ########
##############$#######0###
###################### #####Dianne's horse##############################2######

```
$$Dianne's horse$$
$SomeTag$Dianne's horse$SomeTag$
```

###########################
################################## ##

########
##

```
$function$
BEGIN
    RETURN ($1 ~ $q$[\t\r\n\v\\]$q$);
END;
$function$
```

####q[\t\r\n\v\\]q###########[\t\r\n\v\\]###########PostgreSQL#######
############## ########################$function$############################
############

#####tagString contenttag########TAGString content tag######

#########

###########SQL########### ### #### ############################## ######################### ###################### #####4###################### ##4#########################2################### #############1########

4.1.2.5.

#########B## ###B'1001'# ######## ##################0#1#####

###############X'1FF'###########X################16################ #######16## ######4##2######################

########################

4.1.2.6.

######################

digits
digits.[*digits*][e[+-]*digits*]
[*digits*].*digits*[e[+-]*digits*]
*digits*e[+-]*digits*

###*digits*#1####10####0#9#### ################1###################### ####e#### ########e########1############### ## ############## ## ################## ########

###################

42
3.5
4.
.001
5e2
1.925e-3

###########10#############

0x*hexdigits*
0o*octdigits*
0b*bindigits*

####*hexdigits*#1####16##(0-9#A-F)#*octdigits*#1####8##(0-7)#*bindigits*#1####2##(0 ###1)### 16############################## ##################10################ ##

########10#############

0b100101
0B10011001
0o273
0O755
0x42f
0XFFFF

################################## ####################### #:

1_500_000_000
0b10001000_00000000
0o_1_755
0xFFFF_FFFF
1.618_034

##
#######################

#######################integer##32########integer########### ######
##bigint##64########bigint######## ##########numeric######## ############
#################numeric###########

###
##############real##float4####
#######

```
REAL '1.23'  -- #####
1.23::REAL   -- ######PostgreSQL##
```

##################################

4.1.2.7.

#############################

type 'string'
'string'::type
CAST (*'string'* AS *type*)

##########*type*#################### ############## ########################
###############

#########SQL###################

###########################

typename (*'string'*)

################################## ###4.2.9#########

::#CAST()##########4.2.9##################################### ##############
##*type 'string'*################################ ###*type 'string'########
################ ################::#CAST()#########

CAST()###SQL####### *type 'string'*################ SQL###################
####PostgreSQL############## ::##########PostgreSQL##########
#################

4.1.3.

####NAMEDATALEN-1#######63###########################

+ - * / < > = ~ ! @ # % ^ & | ` ?

#######################

- --#/*################## ##########################
- ##########################1##########+#-#############

  ~ ! @ # % ^ & | ` ?

  ####@-##################*-########## #######PostgreSQL##SQL################## ##################

#SQL## ###@############## ##X*@Y######## PostgreSQL######1#####2##################X* @Y##########

4.1.4.

###############################

- #############$## ######################### ##################################
- ###( )########################## ##############SQL############################
- ####[]#################### ############8.15#########
- ####,##################################
- ######;###SQL############### ##
- ####:######################## (8.15#########)# #####SQL#######SQL############# #############
- #######*## ######################## ##
- #####.############## ##############################

4.1.5.

################################## ##############

-- #####SQL#######

####C####################

/* #############
 * #### /* ############## */
 */

#####/*########*/###### ############C#########SQL################### ###### ##

##

4.1.6.

4.2##PostgreSQL################### ######################### ################ #########

#4.2 ####################

###/##	###	##
.	#	####/########
::	#	PostgreSQL#######
[ ]	#	######
+ -	#	########
COLLATE	#	######
AT	left	AT TIME ZONE, AT LOCAL
^	#	##
* / %	#	#########
+ -	#	#####
(#######)	#	########################
BETWEEN IN LIKE ILIKE SIMILAR		##################
< > = <= >= <>		#####
IS ISNULL NOTNULL		IS TRUE#IS FALSE#IS NULL#IS DISTINCT FROM####
NOT	#	####
AND	#	###
OR	#	###

######+#################### ##################+#########################

########OPERATOR#######################

```
SELECT 3 OPERATOR(pg_catalog.+) 4;
```

OPERATOR##### 4.2#################################### ####OPERATOR()######## ##############

> **##**
>
> 9.5####PostgreSQL############################## ##<=#>=#<>############## ######IS#########################NOT BETWEEN#################### ##BETWEEN####NOT#################### ##SQL######################### ######################### ## ##no such operator#################################### ##

4.2.

#######SELECT#############INSERT#UPDATE################################## ######### ## ################ ############## ###

################

- ##########
- ####
- ###########################
- ######
- #######
- #######
- ######
- ###
- ##########
- #####
- ####(collation)#
- ########
- ########
- #######
- ##

#9 ############## ######IS NULL#######

4.1.2################ ######################

4.2.1.

##################

correlation.columnname

correlation################################FROM################ correlation### ### ##7###########

4.2.2.

##############SQL################### ######SQL########################## #### ##############SQL#################################### ###################### ######### ###################

$*number*

dept

```
CREATE FUNCTION dept(text) RETURNS dept
    AS $$ SELECT * FROM dept WHERE name = $1 $$
    LANGUAGE SQL;
```

###$1##########################

4.2.3.

##

expression[*subscript*]

##

expression[*lower_subscript*:*upper_subscript*]

#######[ ]##*subscript*############
############

########*expression*#######################expression##########################
####### ########################### ##########

```
mytable.arraycolumn[4]
mytable.two_d_column[17][34]
$1[10:42]
(arrayfunction(a,b))[42]
```

############# ######8.15##########

4.2.4.

##

expression.fieldname

#######*expression*############### ### #
#########

```
mytable.mycolumn
$1.somecolumn
(rowfunction(a,b)).col3
```

################################
########

```
(compositecol).somefield
(mytable.compositecol).somefield
```

compositecol###################2######mytable############################
#####

.*##############################

```
(compositecol).*
```

##############################8.16.5##########

4.2.5.

############2########

expression operator expression ########
operator expression ########

###*operator*######4.1.3##################AND#OR#NOT#########################

```
OPERATOR(schema.operatorname)
```

#9################
####

4.2.6.

###

```
function_name ([expression [, expression ... ]] )
```

##########2###########

```
sqrt(2)
```

###########9####### ###############

##10.3#####################

###################4.3########

##
##col(table)#table.col################## ######SQL###### ##PostgreSQL## ####8.16.5#

4.2.7.

###################

```
aggregate_name (expression [ , ... ] [ order_by_clause ] ) [ FILTER
 ( WHERE filter_clause ) ]
aggregate_name (ALL expression [ , ... ] [ order_by_clause ] ) [ FILTER
 ( WHERE filter_clause ) ]
aggregate_name (DISTINCT expression [ , ... ] [ order_by_clause ] )
 [ FILTER ( WHERE filter_clause ) ]
aggregate_name ( * ) [ FILTER ( WHERE filter_clause ) ]
aggregate_name ( [ expression [ , ... ] ] ) WITHIN GROUP ( order_by_clause
 ) [ FILTER ( WHERE filter_clause ) ]
```

```
####aggregate_name##############################expression##############
######################## #####order_by_clause#filter_clause#######

######################1############ ALL#########2################ 3#########
######################################### 4#################1################
#####################count(*)############# ################################
########

##########NULL############1######NULL########### ##########################
##############

####count(*)############# count#NULL########count(f1)#f1##NULL########
##### count(distinct f1)#f1#######NULL########

###########################   ####################min####################   #
########(array_agg###string_agg##)###################    #################
####order_by_clause###################   order_by_clause##7.5#############
#ORDER BY############################################ ##########


WITH vals (v) AS ( VALUES (1),(3),(4),(3),(2) )
SELECT array_agg(v ORDER BY v DESC) FROM vals;
  array_agg
-------------
 {4,3,3,2,1}

######jsonb##########################################


WITH vals (k, v) AS ( VALUES ('key0','1'), ('key1','3'), ('key1','2') )
SELECT jsonb_object_agg(k, v ORDER BY v) FROM vals;
      jsonb_object_agg
----------------------------
 {"key0": "1", "key1": "3"}

#################ORDER BY############################ ####


SELECT string_agg(a, ',' ORDER BY a) FROM table;

####


SELECT string_agg(a ORDER BY a, ',') FROM table;  -- incorrect

######## ################2##ORDER BY#########################(2##############
######)

######DISTINCT#####_by_###############BY###DISTINCT################### #
###:#


WITH vals (v) AS ( VALUES (1),(3),(4),(3),(2) )
SELECT array_agg(DISTINCT v ORDER BY v DESC) FROM vals;
 array_agg
-----------
 {4,3,2,1}

#################ORDER        BY##########################################
##  ####################################order_by_clause####################
#######  ############################  ########order_by_clause############
##WITHIN    GROUP    (...)########  order_by_clause##################1
###########order_by_clause############################  (#WITHIN    GROUP
```

order_by_clause################################) WITHIN GROUP#########
##*order_by_clause*########################### ################################
#######1########### #######GROUP BY#######################################
########################### ##########1#######1########################### ####
##################(*)####()######### (PostgreSQL##########################SQL##
#######)

 ##################

```
SELECT percentile_cont(0.5) WITHIN GROUP (ORDER BY income) FROM
 households;
 percentile_cont
-----------------
           50489
```

########households##income##50################### ###0.5###################
##############

FILTER#########*filter_clause*#################################### ####

```
SELECT
    count(*) AS unfiltered,
    count(*) FILTER (WHERE i < 5) AS filtered
FROM generate_series(1,10) AS s(i);
 unfiltered | filtered
------------+----------
         10 |        4
(1 row)
```

##########9.21######### #################

#####SELECT#############HAVING########### WHERE############## #############
##################

###########4.2.11#9.24####################################### ###########(######
#*filter_clause*)##################### ##
######
###HAVING##

4.2.8.

####################
########1##################################### ##############################
#PARTITION BY################################### #######################

```
function_name ([expression [, expression ... ]]) [ FILTER
 ( WHERE filter_clause ) ] OVER window_name
function_name ([expression [, expression ... ]]) [ FILTER
 ( WHERE filter_clause ) ] OVER ( window_definition )
function_name ( * ) [ FILTER ( WHERE filter_clause ) ] OVER window_name
function_name ( * ) [ FILTER ( WHERE filter_clause ) ] OVER
 ( window_definition )
```

####*window_definition*###########

```
[ existing_window_name ]
[ PARTITION BY expression [, ...] ]
[ ORDER BY expression [ ASC | DESC | USING operator ] [ NULLS { FIRST |
 LAST } ] [, ...] ]
```

[*frame_clause*]

######*frame_clause*#########

{ RANGE | ROWS | GROUPS } *frame_start* [*frame_exclusion*]
{ RANGE | ROWS | GROUPS } BETWEEN *frame_start* AND *frame_end*
 [*frame_exclusion*]

###*frame_start*###*frame_end*##########

UNBOUNDED PRECEDING
offset PRECEDING
CURRENT ROW
offset FOLLOWING
UNBOUNDED FOLLOWING

####*frame_exclusion*##########

EXCLUDE CURRENT ROW
EXCLUDE GROUP
EXCLUDE TIES
EXCLUDE NO OTHERS

####*expression*################################

window_name#######WINDOW###
#*window_definition*#WINDOW###
#SELECT###############OVER wname#OVER (wname ...)#######################################
##

PARTITION BY###PARTITION BY#######################################GROUP BY##########PARTITION BY################################ORDER BY###
#######ORDER BY##########################ORDER BY##################

frame_clause###
###
#####RANGE#######ROWS####GROUPS########*frame_start*##*frame_end*#####*frame_end*##########CURRENT ROW###

frame_start#UNBOUNDED PRECEDING#################################
##*frame_end*#UNBOUNDED FOLLOWING##############################

RANGE####GROUPS######*frame_start*#CURRENT ROW#######################
#ORDER BY##############################*frame_end*#CURRENT ROW#############
####ORDER BY###############ROWS######CURRENT ROW############

offset PRECEDING#*offset* FOLLOWING###########*offset*#####################
################*offset*################

- ROWS######*offset*######NULL###
  ################
- GROUPS######*offset*######NULL#################################(peer group)#####
  ### ##########ORDER BY###############
  ### ########GROUPS########ORDER BY###############
- RANGE######ORDER BY##################################*offset*#######
  ##############################*offset*################################ #
  ##interval###### ############date###

#timestamp###RANGE BETWEEN '1 day' PRECEDING AND '10 days' FOLLOWING# ############ ####*offset*##NULL############## ######################

#############

ROWS#GROUPS######0 PRECEDING#0 FOLLOWING#CURRENT ROW################## ## ########0##########RANGE#########

################################*frame_exclusion*############################ ## EXCLUDE CURRENT ROW################## EXCLUDE GROUP############## ########## EXCLUDE TIES#################################### EXCLUDE NO OTHERS##

################RANGE UNBOUNDED PRECEDING##RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW###### ORDER BY############################ORDER BY################ ORDER BY## ################

####*frame_start*#UNBOUNDED FOLLOWING###########*frame_end*#UNBOUNDED PRECEDING##################*frame_start*#*frame_end*###########*frame_end*### #*frame_start*######################### ####RANGE BETWEEN CURRENT ROW AND *offset* PRECEDING######## #########################ROWS BETWEEN 7 PRECEDING AND 8 PRECEDING######

FILTER#########*filter_clause*## ###### ######FILTER#########

############ 9.65################################ ############################## ################### (##################################)

*###########count(*) OVER (PARTITION BY x ORDER BY y)############### ######################### ######(*)######################## ################### #DISTINCT#ORDER BY##############

#################SELECT####ORDER BY############

##################3.5#9.22#7.2.5######

4.2.9.

################################ PostgreSQL#######2##############

CAST (*expression* AS *type*)
expression::*type*

CAST###SQL######### ::#########PostgreSQL################

#################################### ################################## 4.1.2.7### ################################## ## #### ##

##################################

################################

typename (*expression*)

################################## ####double precision ################
#float8####### ###interval#time#timestamp##############################
######### ##

##

########################### 2###############1###########################
################# ##
################## ################################ #######CREATE CAST#
########

4.2.10.

COLLATE################# ####################

expr COLLATE *collation*

###collation################ COLLATE################## ##################

###
########

COLLATE########2###### 1##ORDER BY############################

```
SELECT a, b, c FROM tbl WHERE ... ORDER BY a COLLATE "C";
```

##

```
SELECT * FROM tbl WHERE a > 'foo' COLLATE "C";
```

######COLLATE## ##################
#COLLATE###
#COLLATE############################## (##################COLLATE############
#####23.2##########) ###########################

```
SELECT * FROM tbl WHERE a COLLATE "C" > 'foo';
```

################

```
SELECT * FROM tbl WHERE (a > 'foo') COLLATE "C";
```

>#########################>################boolean#######

4.2.11.

#############1#1###########SELECT###### ################7########### #
#SELECT#################################### 1#######1#########################
################ #################################### ########NULL######### #
########################### ########################### ##################9.24###
#######

#######################

```
SELECT name, (SELECT max(pop) FROM cities WHERE cities.state =
 states.name)
    FROM states;
```

4.2.12.

################################### ################ARRAY#########[########
#####################]### #########

```
SELECT ARRAY[1,2,3+4];
  array
---------
 {1,2,7}
(1 row)
```

#######################UNION#CASE##################10.5########### ###########
##

```
SELECT ARRAY[1,2,22.7]::integer[];
  array
----------
 {1,2,23}
(1 row)
```

################################### #############4.2.9##########

################################## ############ARRAY############# #############
###

```
SELECT ARRAY[ARRAY[1,2], ARRAY[3,4]];
     array
---------------
 {{1,2},{3,4}}
(1 row)

SELECT ARRAY[[1,2],[3,4]];
     array
---------------
 {{1,2},{3,4}}
(1 row)
```

###ARRAY##########
################################

##################ARRAY################################ ##########

```
CREATE TABLE arr(f1 int[], f2 int[]);

INSERT INTO arr VALUES (ARRAY[[1,2],[3,4]], ARRAY[[5,6],[7,8]]);

SELECT ARRAY[f1, f2, '{{9,10},{11,12}}'::int[]] FROM arr;
                     array
-----------------------------------------------
 {{{1,2},{3,4}},{{5,6},{7,8}},{{9,10},{11,12}}}
(1 row)
```

###

```
SELECT ARRAY[]::integer[];
```

```
 array
-------
 {}
(1 row)
```

########################### ##################ARRAY########################### ############ #########

```
SELECT ARRAY(SELECT oid FROM pg_proc WHERE proname LIKE 'bytea%');
                              array
-----------------------------------------------------------------
 {2011,1954,1948,1952,1951,1244,1950,2005,1949,1953,2006,31,2412}
(1 row)

SELECT ARRAY(SELECT ARRAY[i, i*2] FROM generate_series(1,5) AS a(i));
              array
----------------------------------
 {{1,2},{2,4},{3,6},{4,8},{5,10}}
(1 row)
```

######################## ### ######################### ############################1######################## ##

ARRAY################1######## ##########8.15##########

4.2.13.

##########ROW############ ########0##########################

```
SELECT ROW(1,2.5,'this is a test');
```

ROW#######2######################

#########*rowvalue*.*############# ####SELECT##########.*################## #############(8.16.5##)# ########t#f1##f2##############

```
SELECT ROW(t.*, 42) FROM t;
SELECT ROW(t.f1, t.f2, 42) FROM t;
```

> ##
>
> PostgreSQL 8.2######.*######################ROW(t.*, 42)#######1###### #############2################### ########################### ######## ##################.*##########ROW(t, 42)#########

########ROW######################## ############################CREATE TYPE AS################## ################################ #########

```
CREATE TABLE mytable(f1 int, f2 float, f3 text);

CREATE FUNCTION getf1(mytable) RETURNS int AS 'SELECT $1.f1' LANGUAGE SQL;

-- getf1()#1################
SELECT getf1(ROW(1,2.5,'this is a test'));
```

```
 getf1
-------
     1
(1 row)

CREATE TYPE myrowtype AS (f1 int, f2 text, f3 numeric);

CREATE FUNCTION getf1(myrowtype) RETURNS int AS 'SELECT $1.f1' LANGUAGE
 SQL;


-- ##############################
SELECT getf1(ROW(1,2.5,'this is a test'));
ERROR:  function getf1(record) is not unique

SELECT getf1(ROW(1,2.5,'this is a test')::mytable);
 getf1
-------
     1
(1 row)

SELECT getf1(CAST(ROW(11,'this is a test',2.5) AS myrowtype));
 getf1
-------
    11
(1 row)
```

#
##9.2#################################9.25#########################9.24##########
####################################

4.2.14.

```
################### ##############################################

########################################################## ####


SELECT true OR somefunc();

#########somefunc()############ ###########


SELECT somefunc() OR true;

#########################################################

######################################### ###WHERE####HAVING####################
## ################################ ###########AND/OR/NOT######################
####################

####################CASE###9.18########### ########WHERE##0####################
###


SELECT ... WHERE x > 0 AND y/x > 1.5;

#################


SELECT ... WHERE CASE WHEN x > 0 THEN y/x > 1.5 ELSE false END;

#############CASE################################ ##########y > 1.5*x########
######################
```

#######CASE####################### #############1#########################36.7#######IMMUTABLE## ######## #

```
SELECT CASE WHEN x > 0 THEN x ELSE 1/0 END FROM tab;
```

###############x > 0########ELSE## ###############

###

```
SELECT concat_lower_or_upper('Hello', 'World', true);
 concat_lower_or_upper
-----------------------
 HELLO WORLD
(1 row)
```

##################uppercase#true#################### #########

```
SELECT concat_lower_or_upper('Hello', 'World');
 concat_lower_or_upper
-----------------------
 hello world
(1 row)
```

####uppercase##########################false###################### ######## ##############################

4.3.2.

################=>################### #######

```
SELECT concat_lower_or_upper(a => 'Hello', b => 'World');
 concat_lower_or_upper
-----------------------
 hello world
(1 row)
```

######uppercase#################false######## ############1################## ######## #########

```
SELECT concat_lower_or_upper(a => 'Hello', b => 'World', uppercase =>
 true);
 concat_lower_or_upper
-----------------------
 HELLO WORLD
(1 row)

SELECT concat_lower_or_upper(a => 'Hello', uppercase => true, b =>
 'World');
 concat_lower_or_upper
-----------------------
 HELLO WORLD
(1 row)
```

":="##########################

```
SELECT concat_lower_or_upper(a := 'Hello', uppercase := true, b :=
 'World');
 concat_lower_or_upper
-----------------------
 HELLO WORLD
(1 row)
```

4.3.3.

############################ ##
#######

```
SELECT concat_lower_or_upper('Hello', 'World', uppercase => true);
 concat_lower_or_upper
------------------------
 HELLO WORLD
(1 row)
```

##########a#b########uppercase############ ######################### ######
##
########

##
##

#5#

################################### ############################## ########
####################
######################### ##
#############

5.1.

################################ ################ ##########################
SQL##################
#######7########### ####SQL#####
####SQL######################
############ ######################

################ ######################### ################################
########### ################################ ############################### #
##################################### ######################################

PostgreSQL## ######################## #
##8####### ####################
#integer###########numeric######text######date######time#############
#timestamp######

#################CREATE TABLE########### ##################################
####### #######

```
CREATE TABLE my_first_table (
    first_column text,
    second_column integer
);
```

###2#####my_first_table################# #######first_column#######
#text### 2#######second_column########integer### ###########4.1.1#########
####### #################### ############################

########################### ## #####
############

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric
);
```

#numeric################################

###
##################################

######################### ###########250#1600######## ########################
########################

################DROP TABLE###################### #######

```
DROP TABLE my_first_table;
DROP TABLE products;
```

############################ ##
################################SQL##################### ##################
#######DROP TABLE IF EXISTS############ #########SQL########

##############################5.7##########

#################################### ##
############## ##################################6############

5.2.

####################### ##
############
#######6############

 ############################NULL####### NULL################################
#########

################################## #######

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric DEFAULT 9.99
);
```

###
##timestamp##########################CURRENT_TIMESTAMP############1#####
################### PostgreSQL############################

```
CREATE TABLE products (
    product_no integer DEFAULT nextval('products_product_no_seq'),
    ...
);
```

####nextval()############################9.17########### ######################
###################

```
CREATE TABLE products (
    product_no SERIAL,
    ...
);
```

######SERIAL#8.1.4############

5.3. Identity Columns

##################

#####[81.884058]##########CREATE TABLE#GENERATED ALWAYS AS################
#######################CREATE##GENERATED.AS IDENTITY########### ####:

```
CREATE TABLE people (
    id bigint GENERATED ALWAYS AS IDENTITY,
    ...,
);
```

#########

```
CREATE TABLE people (
    id bigint GENERATED BY DEFAULT AS IDENTITY,
    ...,
);
```

########CREATE TABLE#########

######INSERT###
######################## ####################################

```
INSERT INTO people (name, address) VALUE ('A', 'foo');
INSERT INTO people (name, address) VALUE ('B', 'bar');
```

########1#####id##########################

```
 id | name | address
----+------+---------
  1 | A    | foo
  2 | B    | bar
```

###

```
INSERT INTO people (id, name, address) VALUE (DEFAULT, 'C', 'baz');
```

##################################

##

#############ALWAYS###BY######INSERT#################################
INSERT#######ALWAYS##########INSERT####OVERRIDING#######VALUE#########
#################### BY########################### ######BY################
################################## ###ALWAYS##################################

#######################################1############CREATE SEQUENCE##### ####
################################CREATE TABLE#############ALTER TABLE#####

#######################null############# ########################### ###########
####################
############

#######################ID################ID#################### ##############
ID################## INSERT###########################ID###########ID########
##############ID############

#######
##

5.4.

#####[91.415663]######################## ################################ ########
###2######## ### ###########

######################## ##
######## ###PostgreSQL################

#########CREATE TABLE#GENERATED ALWAYS AS################

```
CREATE TABLE people (
    ...,
    height_cm numeric,
    height_in numeric GENERATED ALWAYS AS (height_cm / 2.54) STORED
);
```

######################STORED############ ######CREATE TABLE########

################# INSERT####UPDATE#########################DEFAULT########

########################## ## ######
######################## ## ######
###############random()##################################

################################

- ##
- ##################
- ####tableoid################
- #####################
- ########################
- #################### ###CREATE FOREIGN TABLE########
- ###############:
  - ############################## ######################## ####################
    ############################## #################### ######################
    ########################
  - ##############################
  - ############CREATE TABLE ... INHERITS#GENERATED####################
    #GENERATED################ ALTER TABLE ... INHERIT##################
    ##############################
  - ######################## CREATE TABLE ... PARTITION OF#GENERATED##
    ###################GENERATED################ ALTER TABLE ... ATTACH
    PARTITION##
  - ## ##################################
    #########

########################

- ########################### ##
  ##
- #########BEFORE################ #####BEFORE########################### ###
  ######BEFORE####################
- ########################CREATE PUBLICATION###############

5.5.

################################## ################################### ########
################################## ############################## #############
################# ####################1#############2#############

###############SQL####################### ######################### ###########
#################### ####################################

5.5.1.

################### ###################################### ####################
#######

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric CHECK (price > 0)
);
```

#################################### ###################### #####CHECK########
################# ######################## ########################

################### ## ############

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric CONSTRAINT positive_price CHECK (price > 0)
);
```

################CONSTRAINT######################## ########################
############

###################### ##

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric CHECK (price > 0),
    discounted_price numeric CHECK (discounted_price > 0),
    CHECK (price > discounted_price)
);
```

###2############### 3################### ####################################
######### ############################

###2####################3################################ ######################
#PostgreSQL###########
####################

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric,
    CHECK (price > 0),
    discounted_price numeric,
```

```
    CHECK (discounted_price > 0),
    CHECK (price > discounted_price)
);
```

################

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric CHECK (price > 0),
    discounted_price numeric,
    CHECK (discounted_price > 0 AND price > discounted_price)
);
```

#################

##############################

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric,
    CHECK (price > 0),
    discounted_price numeric,
    CHECK (discounted_price > 0),
    CONSTRAINT valid_discount CHECK (price > discounted_price)
);
```

###############NULL############################## #################NULL### #NULL#####################NULL########### ##NULL#######################NULL#########

> ##
>
> PostgreSQL##################################CHECK############## ######### #CHECK## ################ ################################### ################### ### #### #UNIQUE#EXCLUDE#FOREIGN KEY##############################
>
> ## #pg_dump# ###########################/#########################/###############
>
> ##
>
> PostgreSQL#CHECK################# ############################ ###### #CHECK### ######################### ###################
>
> ###############CHECK################################ PostgreSQL######### ####CHECK########################### ################################## ## #####################ALTER TABLE#################################### ###########################

5.5.2. #NULL##

#NULL########NULL############## ##############

```
CREATE TABLE products (
    product_no integer NOT NULL,
    name text NOT NULL,
    price numeric
);
```

#NULL################ #NULL###CHECK (*column_name* IS NOT NULL)####### ##########PostgreSQL#######NULL################ ##########NULL########### ##########

#####1#################### ######################

```
CREATE TABLE products (
    product_no integer NOT NULL,
    name text NOT NULL,
    price numeric NOT NULL CHECK (price > 0)
);
```

########## ############################

NOT NULL##############NULL####### ######NULL#################### ######## ######## ########NULL############################## NULL#####SQL########### ######################## #####PostgreSQL#################################### ## #######

```
CREATE TABLE products (
    product_no integer NULL,
    name text NULL,
    price numeric NULL
);
```

############NOT############

> ###
>
> ######################NOT NULL##############

5.5.3.

##################

```
CREATE TABLE products (
    product_no integer UNIQUE,
    name text,
    price numeric
);
```

###############

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric,
```

```
    UNIQUE (product_no)
);
```

######

```
CREATE TABLE example (
    a integer,
    b integer,
    c integer,
    UNIQUE (a, c)
);
```

```
CREATE TABLE products (
    product_no integer CONSTRAINT must_be_different UNIQUE,
    name text,
    price numeric
);
```

```
CREATE TABLE products (
    product_no integer UNIQUE NULLS NOT DISTINCT,
    name text,
    price numeric
);
```

###

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric,
    UNIQUE NULLS NOT DISTINCT (product_no)
);
```

5.5.4.

```
CREATE TABLE products (
    product_no integer UNIQUE NOT NULL,
    name text,
    price numeric
```

```
);

CREATE TABLE products (
    product_no integer PRIMARY KEY,
    name text,
    price numeric
);
```

##################################

```
CREATE TABLE example (
    a integer,
    b integer,
    c integer,
    PRIMARY KEY (a, c)
);
```

####################################B#################### ##########NOT NULL# ########

1########1############### ##########NULL############### ####################### ########1####### ############################1########### #####PostgreSQL###### ###########################

####################################### #############GUI######################### ###################### ## ######## ################################

5.5.5.

##########2##############

#############products##############

```
CREATE TABLE products (
    product_no integer PRIMARY KEY,
    name text,
    price numeric
);
```

##################################### #####orders################################ ## ####products#########orders#################

```
CREATE TABLE orders (
    order_id integer PRIMARY KEY,
    product_no integer REFERENCES products (product_no),
    quantity integer
);
```

####products###########NULL#product_no#########################

#########orders###############product#################### ##################

##########################

```
CREATE TABLE orders (
    order_id integer PRIMARY KEY,
    product_no integer REFERENCES products,
```

```
    quantity integer
);
```

##################################

##############################

################################## ########################### ################
##

```
CREATE TABLE t1 (
  a integer PRIMARY KEY,
  b integer,
  c integer,
  FOREIGN KEY (b, c) REFERENCES other_table (c1, c2)
);
```

##

################################## ################# ########################
##############################

```
CREATE TABLE tree (
    node_id integer PRIMARY KEY,
    parent_id integer REFERENCES tree,
    name text,
    ...
);
```

##########parent_id#NULL######NULL#parent_id##########################
###

######################## ############################## ######################
############################## ############### ####################

```
CREATE TABLE products (
    product_no integer PRIMARY KEY,
    name text,
    price numeric
);

CREATE TABLE orders (
    order_id integer PRIMARY KEY,
    shipping_address text,
    ...
);

CREATE TABLE order_items (
    product_no integer REFERENCES products,
    order_id integer REFERENCES orders,
    quantity integer,
    PRIMARY KEY (product_no, order_id)
);
```

##################################

##################################
SQL###################

- ###############

- #########
- ########

################################ #order_items############################## ############ ####################


```
CREATE TABLE products (
    product_no integer PRIMARY KEY,
    name text,
    price numeric
);

CREATE TABLE orders (
    order_id integer PRIMARY KEY,
    shipping_address text,
    ...
);

CREATE TABLE order_items (
    product_no integer REFERENCES products ON DELETE RESTRICT,
    order_id integer REFERENCES orders ON DELETE CASCADE,
    quantity integer,
    PRIMARY KEY (product_no, order_id)
);
```

###############2################ RESTRICT################## NO ACTION#### ## ########################## ######### ###NO ACTION##########################RESTRICT################## CASCADE# ##################################### ###2#######SET NULL#SET DEFAULT#### ## ##############################NULL################## ######################## ############# #######SET DEFAULT######################################

ON DELETE## ######################## ######################################CASCADE###### 2######################## ##RESTRICT###NO ACTION###### ######################################2######## ############ ### ############ ## ##########################SET NULL###SET DEFAULT###### ### #########NULL##################

#####SET NULL###SET DEFAULT######################## ###################### ####################################


```
CREATE TABLE tenants (
    tenant_id integer PRIMARY KEY
);

CREATE TABLE users (
    tenant_id integer REFERENCES tenants ON DELETE CASCADE,
    user_id integer NOT NULL,
    PRIMARY KEY (tenant_id, user_id)
);

CREATE TABLE posts (
    tenant_id integer REFERENCES tenants ON DELETE CASCADE,
    post_id integer NOT NULL,
    author_id integer,
    PRIMARY KEY (tenant_id, post_id),
    FOREIGN KEY (tenant_id, author_id) REFERENCES users ON DELETE SET
 NULL (author_id)
);
```

##################tenant_id##null#########################

ON DELETE#############################ON UPDATE###### #####SET NULL#SET DEFAULT######################### #####CASCADE#############################

#################null###################### ##MATCH FULL#################################null#################################null##null#######MATCH FULL############# ##### ##################################NOT NULL###########

###########DELETE######UPDATE## ##

####################6########### ###CREATE TABLE############################ ####

5.5.6.

#########2###1#######null####### ###### ############

```
CREATE TABLE circles (
    c circle,
    EXCLUDE USING gist (c WITH &&)
);
```

```
###CREATE TABLE ... CONSTRAINT ... EXCLUDE#########
```

```
########################################
```

5.6.

#####################

```
tableoid
```

```
    ###########OID### ############################5.12#########################
    ########5.11###########                    ##################################
    tableoid#pg_class#oid#######################
```

```
xmin
```

```
    ##################################ID#### ################### ################
    #######################
```

```
cmin
```

```
    #############0###############
```

```
xmax
```

```
    ########################ID#### #################### ########################
    ## ############################################################
```

```
cmax
```

#############################

ctid

############################# ctid#################################ctid####
UPDATE########VACUUM FULL############## ######ctid################## ###
#####################

###########32####### ########################ID################ ###############
################ ####24########### #######10#########################ID#########
############

###########32####### ########################2^{32}#40########SQL################ #
############### ###SQL## ############
###########################

5.7.

###########
###
##PostgreSQL ##
################################

##########

- ####
- ####
- #####
- #####
- #########
- #########
- #####
- ########

#########ALTER TABLE##

5.7.1.

#########################

```
ALTER TABLE products ADD COLUMN description text;
```

#########################DEFAULT##########NULL########

> ###
>
> PostgreSQL 11#############################ALTER TABLE##################
> ## ##ALTER TABLE##
> ##################
>
> ########################clock_timestamp()##########ALTER TABLE#######
> ############### ##
> #UPDATE############################ ##############################

##########################

```
ALTER TABLE products ADD COLUMN description text CHECK (description <>
 '');
```

####CREATE TABLE################################ ############################
##########################ADD##

5.7.2.

#######################

```
ALTER TABLE products DROP COLUMN description;
```

#################### ######################## ################################
##PostgreSQL#################### CASCADE################################

```
ALTER TABLE products DROP COLUMN description CASCADE;
```

####################5.15#########

5.7.3.

##########################

```
ALTER TABLE products ADD CHECK (name <> '');
ALTER TABLE products ADD CONSTRAINT some_name UNIQUE (product_no);
ALTER TABLE products ADD FOREIGN KEY (product_group_id) REFERENCES
 product_groups;
```

#NULL####################################

```
ALTER TABLE products ALTER COLUMN product_no SET NOT NULL;
```

##

5.7.4.

######################### ################ ##################################
################### ####psql#\d *tablename*############## ####################
############ #########################

```
ALTER TABLE products DROP CONSTRAINT some_name;
```

########$2##

####################################CASCADE########### ######################
################

#####[91.666667]######NULL################## #NULL##################

```
ALTER TABLE products ALTER COLUMN product_no DROP NOT NULL;
```

##NULL#####################

5.7.5.

##################################

```
ALTER TABLE products ALTER COLUMN price SET DEFAULT 7.77;
```

##INSERT#######################

#####################

```
ALTER TABLE products ALTER COLUMN price DROP DEFAULT;
```

##########NULL############ ################################## #### NULL#####
################

5.7.6.

##################################

```
ALTER TABLE products ALTER COLUMN price TYPE numeric(10,2);
```

################################
####USING############

PostgreSQL## ############
################################## ##
#########

5.7.7.

##################

```
ALTER TABLE products RENAME COLUMN product_no TO product_number;
```

5.7.8.

#####################

```
ALTER TABLE products RENAME TO items;
```

5.8.

##(#
########)################## ##

#####[90.784044]################# ####SELECT#
INSERT#UPDATE#DELETE#TRUNCATE#REFERENCES#TRIGGER#CREATE#CONNECT#TEMPORARY#
EXECUTE#USAGE#SET#ALTER SYSTEM### ##
################################

##############################
###########21.3########

#########################ALTER####################################

```
ALTER TABLE table_name OWNER TO new_owner;
```

###################### ##
#SET ROLE##################

###########GRANT########### ####joe##########accounts######################
#####################

GRANT UPDATE ON accounts TO joe;

##############ALL##################################

##################################PUBLIC####### ############################
######################### ####21###########

#############(revoke)############REVOKE##########

REVOKE ALL ON accounts FROM PUBLIC;

####with grant option#########
##################################
####GRANT#REVOKE##########

########
##

###########

SELECT

    ##SELECT################
    ######### ###COPY TO######### #####UPDATE#DELETE###MERGE###############
    ######### #############currval########### ##############################
    ####

INSERT

    ################INSERT########## ######INSERT##########################
    #### ######################## COPY FROM############

UPDATE

    #############UPDATE########## #############UPDATE######SELECT###########
    ## SELECT ... FOR
    UPDATE#SELECT ... FOR SHARE#SELECT#################### ##########
    #nextval#setval########### ##

DELETE

    #############DELETE########## #############DELETE######SELECT###########
    ##

TRUNCATE

    #####TRUNCATE#######

REFERENCES

    ##

TRIGGER

    ##########################

CREATE

##

################################### ######################################
##############################

###
#################

###

CONNECT

############################ #pg_hba.conf##############################

TEMPORARY

############################

EXECUTE

###################################### ############################

USAGE

##################################### #########################

#####
################################
########################## ##
######################################

##########currval#nextval###########

##############
########################
##

##

#################################### ####################################
#######

SET

###
##################

ALTER SYSTEM

ALTER SYSTEM#####################################

MAINTAIN

##################ANALYZE#######################################
###########

###

PostgreSQL###PUBLIC###### ###########
##PUBLIC########
################PUBLIC#################### CONNECT#TEMPORARY

##############################EXECUTE######################USAGE### ###### ##############################REVOKE##### ###################REVOKE######## ## ##############ALTER DEFAULT PRIVILEGES###########

5.1##*ACL*#Access Control List###############1########### psql################ACL

#5.1 ACL###

##	###	##############
SELECT	r (#read#)	LARGE OBJECT, SEQUENCE, TABLE####### ##################
INSERT	a (#append#)	TABLE#######
UPDATE	w (#write#)	LARGE OBJECT, SEQUENCE, TABLE, ######
DELETE	d	TABLE
TRUNCATE	D	TABLE
REFERENCES	x	TABLE#######
TRIGGER	t	TABLE
CREATE	C	DATABASE, SCHEMA, TABLESPACE
CONNECT	c	DATABASE
TEMPORARY	T	DATABASE
EXECUTE	X	FUNCTION, PROCEDURE
USAGE	U	DOMAIN, FOREIGN DATA WRAPPER, FOREIGN SERVER, LANGUAGE, SCHEMA, SEQUENCE, TYPE
SET	s	PARAMETER
ALTER SYSTEM	A	PARAMETER
MAINTAIN	m	TABLE

5.2####################SQL###################### ########################### #####psql##########

#5.2

#########	######	#####PUBLIC##	psql####
DATABASE	CTc	Tc	\l
DOMAIN	U	U	\dD+
FUNCTION or PROCEDURE	X	X	\df+
FOREIGN DATA WRAPPER	U	none	\dew+
FOREIGN SERVER	U	none	\des+
LANGUAGE	U	U	\dL+
LARGE OBJECT	rw	none	\dl+
PARAMETER	sA	none	\dconfig+
SCHEMA	UC	none	\dn+
SEQUENCE	rwU	none	\dp
TABLE (and table-like objects)	arwdDxtm	none	\dp
######	arwx	none	\dp

#########	######	#####PUBLIC##	psql####
TABLESPACE	C	none	\db+
TYPE	U	U	\dT+

##################aclitem################# ################

grantee=privilege-abbreviation[*]*.../grantor*

#aclitem################1################### ####### 5.1#1############ ################*######## #####calvin=r*w/hobbes#####calvin#########*### #SELECT#r###########UPDATE (w)##########hobbes############### ############ ####################calvin##############aclitem############# aclitem###### ##############PUBLIC######

###miriam#####mytable################

```
GRANT SELECT ON mytable TO PUBLIC;
GRANT SELECT, UPDATE, INSERT ON mytable TO admin;
GRANT SELECT (col1), UPDATE (col1) ON mytable TO miriam_rw;
```

####psql#\dp##################

```
=> \dp mytable
                                 Access privileges
 Schema |  Name   | Type  |   Access privileges    |   Column privileges
 | Policies
--------+---------+-------+------------------------
+------------------------+----------
 public | mytable | table | miriam=arwdDxtm/miriam+| col1:
 +|
        |         |       | =r/miriam             +|   miriam_rw=rw/miriam
 |
        |         |       | admin=arw/miriam       |
 |
(1 row)
```

############Access privileges################################## ################ #########NULL######### ##PUBLIC# ############### ############GRANT####REVOKE################miriam=arwdDxt/ miriam######################## #####Column privileges######################## ########## ## ALTER DEFAULT PRIVILEGES##########################ALTER##########################

*############################

################################null###################### ##############—##### ## ###########################

5.9.

GRANT########SQL## ################################ ######################### ################### ####SQL###

##################ALTER TABLE ... ENABLE ROW LEVEL SECURITY####### ### ########################

################# ##
TRUNCATE#REFERENCES################################

#####ALL####################
#SELECT#INSERT#UPDATE#DELETE############## 1##############################
####################

##############################
########## ##########################leakproof####### ######################
####################### ##true################ ##############################
################ ### #################
##################################

###########BYPASSRLS## ##############
############ALTER TABLE ... FORCE ROW LEVEL SECURITY#####################
###########

##

#####CREATE POLICY#########ALTER POLICY#########DROP POLICY###########
##########################ALTER TABLE#########

############1##################### ###################1#####################
################################

################################(permissive)##########OR######(restrictive)########
AND########### ###
###########

########managers##account########
#############

```
CREATE TABLE accounts (manager text, company text, contact_email text);

ALTER TABLE accounts ENABLE ROW LEVEL SECURITY;

CREATE POLICY account_managers ON accounts TO managers
    USING (manager = current_user);
```

############USING####WITH CHECK##
####################SELECT#UPDATE#DELETE#################################
################INSERT####UPDATE############

##########################PUBLIC################################   #######
#users##

```
CREATE POLICY user_policy ON users
    USING (user_name = current_user);
```

################

##################
####users######################################

```
CREATE POLICY user_sel_policy ON users
    FOR SELECT
    USING (true);
CREATE POLICY user_mod_policy ON users
    USING (user_name = current_user);
```

SELECT#######OR####2################################ ######################
###############

########ALTER TABLE############## ##
######## #####SQL################################

#################################### passwd#####Unix##################

```
-- passwd############
CREATE TABLE passwd (
  user_name             text UNIQUE NOT NULL,
  pwhash                text,
  uid                   int  PRIMARY KEY,
  gid                   int  NOT NULL,
  real_name             text NOT NULL,
  home_phone            text,
  extra_info            text,
  home_dir              text NOT NULL,
  shell                 text NOT NULL
);


CREATE ROLE admin;  -- ###
CREATE ROLE bob;    -- #####
CREATE ROLE alice;  -- #####


-- ##########
INSERT INTO passwd VALUES
  ('admin','xxx',0,0,'Admin','111-222-3333',null,'/root','/bin/dash');
INSERT INTO passwd VALUES
  ('bob','xxx',1,1,'Bob','123-456-7890',null,'/home/bob','/bin/zsh');
INSERT INTO passwd VALUES
  ('alice','xxx',2,1,'Alice','098-765-4321',null,'/home/alice','/bin/
zsh');


-- ####################
ALTER TABLE passwd ENABLE ROW LEVEL SECURITY;


-- #########
-- ##############################
CREATE POLICY admin_all ON passwd TO admin USING (true) WITH CHECK (true);

-- ####################
CREATE POLICY all_view ON passwd FOR SELECT USING (true);

-- #####################
-- #####################
CREATE POLICY user_mod ON passwd FOR UPDATE
  USING (current_user = user_name)
  WITH CHECK (
    current_user = user_name AND
    shell IN ('/bin/bash','/bin/sh','/bin/dash','/bin/zsh','/bin/tcsh')
  );


-- admin################
GRANT SELECT, INSERT, UPDATE, DELETE ON passwd TO admin;

-- ##########SELECT#############
```

```
GRANT SELECT
  (user_name, uid, gid, real_name, home_phone, extra_info, home_dir,
 shell)
  ON passwd TO public;

-- #####################
GRANT UPDATE
  (pwhash, real_name, home_phone, extra_info, shell)
  ON passwd TO public;

##################################################### ########################
#################



-- admin#################
postgres=> set role admin;
SET
postgres=> table passwd;
 user_name | pwhash | uid | gid | real_name |  home_phone  | extra_info |
 home_dir    |   shell
-----------+--------+-----+-----+-----------+--------------+------------
+-------------+-----------
 admin     | xxx    |   0 |   0 | Admin     | 111-222-3333 |
 | /root       | /bin/dash
 bob       | xxx    |   1 |   1 | Bob       | 123-456-7890 |
 | /home/bob   | /bin/zsh
 alice     | xxx    |   2 |   1 | Alice     | 098-765-4321 |
 | /home/alice | /bin/zsh
(3 rows)


-- Alice############
postgres=> set role alice;
SET
postgres=> table passwd;
ERROR:  permission denied for table passwd
postgres=> select user_name,real_name,home_phone,extra_info,home_dir,shell
 from passwd;
 user_name | real_name |  home_phone  | extra_info | home_dir    |   shell
-----------+-----------+--------------+------------+-------------
+-----------
 admin     | Admin     | 111-222-3333 |            | /root       | /bin/
dash
 bob       | Bob       | 123-456-7890 |            | /home/bob   | /bin/
zsh
 alice     | Alice     | 098-765-4321 |            | /home/alice | /bin/
zsh
(3 rows)

postgres=> update passwd set user_name = 'joe';
ERROR:  permission denied for table passwd

-- Alice####real_name################
postgres=> update passwd set real_name = 'Alice Doe';
UPDATE 1
postgres=> update passwd set real_name = 'John Doe' where user_name =
 'admin';
UPDATE 0
postgres=> update passwd set shell = '/bin/xx';
ERROR:  new row violates WITH CHECK OPTION for "passwd"
postgres=> delete from passwd;
ERROR:  permission denied for table passwd
postgres=> insert into passwd (user_name) values ('xxx');
```

```
ERROR:  permission denied for table passwd

-- Alice################
-- RLS#######################
postgres=> update passwd set pwhash = 'abc';
UPDATE 1

#######################################################OR############### #######
#########################################################################
##############################AND############ ##################Unix#########
###passwd####################################


CREATE POLICY admin_local_only ON passwd AS RESTRICTIVE TO admin
    USING (pg_catalog.inet_client_addr() IS NULL);

##########################################################


=> SELECT current_user;
 current_user
--------------
 admin
(1 row)

=> select inet_client_addr();
 inet_client_addr
------------------
 127.0.0.1
(1 row)

=> TABLE passwd;
 user_name | pwhash | uid | gid | real_name | home_phone | extra_info |
 home_dir | shell
-----------+--------+-----+-----+-----------+------------+------------
+----------+-------
(0 rows)

=> UPDATE passwd set pwhash = NULL;
UPDATE 0

############################################################ ##################
##########################(covert channel)##########################

######################################## ########################################
######################## ################row_security#off########### ##########
################################################## ##########################

#############################################  #####################  #########
##############################  ##############################################
#SELECT#######SELECT############### ########################################
###################### ##################



-- #########
CREATE TABLE groups (group_id int PRIMARY KEY,
                     group_name text NOT NULL);

INSERT INTO groups VALUES
  (1, 'low'),
  (2, 'medium'),
  (5, 'high');
```

```
GRANT ALL ON groups TO alice;  -- alice####
GRANT SELECT ON groups TO public;


-- ############
CREATE TABLE users (user_name text PRIMARY KEY,
                    group_id int NOT NULL REFERENCES groups);

INSERT INTO users VALUES
  ('alice', 5),
  ('bob', 2),
  ('mallory', 2);

GRANT ALL ON users TO alice;
GRANT SELECT ON users TO public;


-- ################
CREATE TABLE information (info text,
                          group_id int NOT NULL REFERENCES groups);

INSERT INTO information VALUES
  ('barely secret', 1),
  ('slightly secret', 2),
  ('very secret', 5);

ALTER TABLE information ENABLE ROW LEVEL SECURITY;


-- #######group_id###group_id#############
-- ##################
CREATE POLICY fp_s ON information FOR SELECT
  USING (group_id <= (SELECT group_id FROM users WHERE user_name =
 current_user));
CREATE POLICY fp_u ON information FOR UPDATE
  USING (group_id <= (SELECT group_id FROM users WHERE user_name =
 current_user));


-- information###########RLS#######
GRANT ALL ON information TO public;

###alice##slightly secret#########################mallory################ #######
#########


BEGIN;
UPDATE users SET group_id = 1 WHERE user_name = 'mallory';
UPDATE information SET info = 'secret from mallory' WHERE group_id = 2;
COMMIT;

############# mallory##secret from mallory#################### #################
####mallory###################


SELECT * FROM information WHERE group_id = 2 FOR UPDATE;

##############READ COMMITTED#########secret from mallory########### #########
######alice############information################# ###alice##############
###############FOR UPDATE######################## ####users######SELECT####
############ #######SELECT##FOR UPDATE#####users#########################
######### ############mallory#####################################
```

#################### ####################SELECT#SELECT ... FOR SHARE### ##### ################################users##UPDATE######################### ### ####SELECT############# ########### ## ################ ######### ##ACCESS EXCLUSIVE####### ##### ################################### ################################## ##

#######CREATE POLICY#ALTER TABLE#########

5.10.

PostgreSQL############1#################### ################################## ### ##

> ##
>
> ## ##################joe#### ######################2########################## ####joe############# ########################

#####[63.731170]##########1################################ ################### ##################### ################################ ####schema1#myschema# #######mytable################### ## ## ####################1######### ## ## 1################2############### ######### ################################(Materialized View)##################### ## ############################ ###### #########1#myschema#######mytable############### ################### ######### ##

#######################

- 1#####################################
- ####################################
- ##

##

5.10.1.

###########CREATE SCHEMA########### ################ #######

```
CREATE SCHEMA myschema;
```

##

schema.table

##

################

database.schema.table

#######################SQL#################### ############################
###########

################################

```
CREATE TABLE myschema.mytable (
 ...
);
```

##

```
DROP SCHEMA myschema;
```

####################################

```
DROP SCHEMA myschema CASCADE;
```

####################5.15##########

###1##### #############
##

```
CREATE SCHEMA schema_name AUTHORIZATION user_name;
```

#################################### #############5.10.6##########

pg_##

5.10.2. public####

################################ ##################################public######
############ ############################ #######2##########

```
CREATE TABLE products ( ... );
```

###

```
CREATE TABLE public.products ( ... );
```

5.10.3.

################
######################### ## ###
############################## ##
##########

################
################################   PostgreSQL#######################
##search_path#############CREATE################################### #####
###SQL##
########

```
CREATE TABLE
```

##################################   ####################################
#CREATE TABLE######################################

#######################

```
SHOW search_path;
```

####################

```
 search_path
--------------
 "$user", public
```

#################################### ################################ 2#######
######public############

#################public
############## ##
####################### #######################public##############

#########################

```
SET search_path TO myschema,public;
```

#$user##################### ################################

```
DROP TABLE mytable;
```

###myschema#####################################

###############

```
SET search_path TO myschema;
```

#################public####################### public##############################
#####################

#####################9.27#########

################################
############

```
OPERATOR(schema.operator)
```

######################### ##########

```
SELECT 3 OPERATOR(pg_catalog.+) 4;
```

##

5.10.4.

########################
#USAGE############### ##########public############# ######################
#####################################

##################CREATE###########
PostgreSQL 14#########################public###################
##################:

```
REVOKE CREATE ON SCHEMA public FROM PUBLIC;
```

#####public########2####public################ ###public######2###public##########
##################4.1.1##################

5.10.5.

#########public###############pg_catalog############ ####################
#################### pg_catalog############### ######################
################## ############################ #########################
##pg_catalog###################

###########pg_############################### ##############################
################## ##
###########pg_###############pg_#######################################
########

5.10.6.

####################### ##
##
#####search_path################################search_path########
################# ####################################

- ######################## ########################public CREATE#############
  #### ## #####CREATE
  SCHEMA alice AUTHORIZATION alice######## ###########$user############
  ################# ## ########
  ####################################ADMIN OPTION#########################
  ########################

  PostgreSQL 15################################## ############################
  ###############public#####public CREATE##############REVOKE CREATE ON
  SCHEMA public FROM PUBLIC####### ####public########pg_catalog######
  ######################################

- postgresql.conf#######ALTER ROLE ALL SET search_path = ""#######
  #############public########## #####public################ #########public#####
  ############ ########################public################################
  public#################################### ################################
  ###############ADMIN OPTION####################

- #############public############### #######public################ ##########
  ## ########################
  ## PostgreSQL 14########
  ########################

###
####### ## ####################
##

5.10.7.

##SQL###1####################################### ###########################
##################### ##
################## #########*user_name.table_name*#######################
PostgreSQL###########1#####################

#####SQL###public############### ############public################

##
#SQL########### ##

5.11.

PostgreSQL################################## #SQL:1999##########################
#################

########### ##cities############################ ########################capitals#
#1###### ######################### ####2######################### 1############1###
############# ## ##############
cities#######capitals############

```
CREATE TABLE cities (
    name            text,
    population      float,
    elevation       int     -- in feet
);

CREATE TABLE capitals (
    state           char(2)
) INHERITS (cities);
```

#####capitals###############cities############### ###1######state#########
##

PostgreSQL###1########0#################### ##################################
######################### ################ ############500####################
#################

```
SELECT name, elevation
    FROM cities
    WHERE elevation > 500;
```

PostgreSQL#################2.1##################################

```
   name    | elevation
-----------+-----------
 Las Vegas |      2174
 Mariposa  |      1953
 Madison   |       845
```

##################500############################

```
SELECT name, elevation
    FROM ONLY cities
    WHERE elevation > 500;

   name    | elevation
-----------+-----------
 Las Vegas |      2174
 Mariposa  |      1953
```

###ONLY############cities##########cities############################ ###
###########—SELECT#UPDATE###DELETE —#ONLY###############

#####################################*##########:

```
SELECT name, elevation
    FROM cities*
    WHERE elevation > 500;
```

*################################ ##
###

################################### ###########tableoid####################
#####

```
SELECT c.tableoid, c.name, c.elevation
FROM cities c
WHERE c.elevation > 500;
```

############

```
 tableoid |   name    | elevation
----------+-----------+-----------
   139793 | Las Vegas |      2174
   139793 | Mariposa  |      1953
   139798 | Madison   |       845
```

#######################OID########### pg_class#########################

```
SELECT p.relname, c.name, c.elevation
FROM cities c, pg_class p
WHERE c.elevation > 500 AND c.tableoid = p.oid;
```

###########

```
 relname  |   name    | elevation
----------+-----------+-----------
 cities   | Las Vegas |      2174
 cities   | Mariposa  |      1953
 capitals | Madison   |       845
```

################regclass################OID###########

```
SELECT c.tableoid::regclass, c.name, c.elevation
FROM cities c
WHERE c.elevation > 500;
```

###INSERT###COPY################################ ########INSERT########

```
INSERT INTO cities (name, population, elevation, state)
VALUES ('Albany', NULL, NULL, 'NY');
```

###########capitals################################## INSERT################
############## ########39############################## ######################
######## #####cities#####state######################################

#############NULL####NO INHERIT#################################### #########
########################

#####1################## ############################## ########################
######## ##1###
###########NULL#######
############ ##################NULL######################NULL########## ######
######################################

##########CREATE TABLE##INHERITS########################## ################## ######################## ####ALTER TABLE#INHERIT######## ################## ################################### ################################### ALTER TABLE#NO INHERIT################################## ########################## #############5.12##################

##################################1###CREATE TABLE#LIKE########## ###### ########################## ## #CHECK##########LIKE#INCLUDING CONSTRAINTS#############

############################## ## ### ########################CASCADE############################5.15#####

ALTER TABLE################################## ####################CASCADE## ################### ALTER TABLE###############CREATE TABLE##############

############################### ########UPDATE###cities############cities## ###############capitals######################## ########################## ####### ####capitals########################## ####################5.9###### ################################## ## ##### ##################################

#######5.13## ##################### ####################################

5.11.1.

####SQL################################## ################################ ####SELECT#UPDATE#DELETE#ALTER TABLE###############INSERT#ALTER TABLE ... RENAME#######################################ONLY######## ## ####################REINDEX#VACUUM## ########## ##########################SQL##############

###########

- ###cities.name#UNIQUE###PRIMARY KEY######cities############capitals## ##################### ###################cities################### #### ##capitals## capitals######## ############cities#########################
- ######cities.name REFERENCES##########################capitals######### ####### #####capitals#######REFERENCES####################
- #########REFERENCES cities(name)## ##### ################

#########################

5.12.

PostgreSQL################################ ################################## ##################

5.12.1.

############

- ################################## ### ########### ## #####
- ## ##
- ### ### #######DROP TABLE#########ALTER TABLE DETACH PARTITION############## ############## ##############DELETE#######VACUUM##################
- ####################################

##

PostgreSQL##############################

###########

5.12.2.1.

```
################################### ##################################### #####
##############


CREATE TABLE measurement (
    city_id         int not null,
    logdate         date not null,
    peaktemp        int,
    unitsales       int
);

##############################################################################
############ ######################3################### #######################
##################################################

#################################
```

1. PARTITION BY######measurement#################### PARTITION BY### ################RANGE##########################

   ```
   CREATE TABLE measurement (
       city_id         int not null,
       logdate         date not null,
       peaktemp        int,
       unitsales       int
   ) PARTITION BY RANGE (logdate);
   ```

2. ############# ## ## ## ################ ################################ ################################

   ##########################PostgreSQL################################ ####### ################################

   ## ################### ######

   ```
   CREATE TABLE measurement_y2006m02 PARTITION OF measurement
       FOR VALUES FROM ('2006-02-01') TO ('2006-03-01');

   CREATE TABLE measurement_y2006m03 PARTITION OF measurement
       FOR VALUES FROM ('2006-03-01') TO ('2006-04-01');

   ...
   CREATE TABLE measurement_y2007m11 PARTITION OF measurement
       FOR VALUES FROM ('2007-11-01') TO ('2007-12-01');

   CREATE TABLE measurement_y2007m12 PARTITION OF measurement
       FOR VALUES FROM ('2007-12-01') TO ('2008-01-01')
       TABLESPACE fasttablespace;

   CREATE TABLE measurement_y2008m01 PARTITION OF measurement
       FOR VALUES FROM ('2008-01-01') TO ('2008-02-01')
       WITH (parallel_workers = 4)
       TABLESPACE fasttablespace;
   ```

   ##

   ##PARTITION BY###########

```
CREATE TABLE measurement_y2006m02 PARTITION OF measurement
    FOR VALUES FROM ('2006-02-01') TO ('2006-03-01')
    PARTITION BY RANGE (peaktemp);
```

measurement_y2006m02############measurement########
#measurement_y2006m02##############measurement_y2006m02############
#########################peaktemp################################ ########
###
############################ ################################

#########################

#################
3. ## ######################
   ########################## ##
   ################################### ######################################
   ############### ################################

```
CREATE INDEX ON measurement (logdate);
```

4. postgresql.conf#######enable_partition_pruning################## ##########
   ######################

##################################DDL###########################

5.12.2.2.

####################
#################################### ######################################
###
#####

##

DROP TABLE measurement_y2006m02;

####################ACCESS
EXCLUSIVE########################

###
####2##########

```
ALTER TABLE measurement DETACH PARTITION measurement_y2006m02;
ALTER TABLE measurement DETACH PARTITION measurement_y2006m02
 CONCURRENTLY;
```

#################################### ####COPY#pg_dump######################
############### ##
##########ACCESS EXCLUSIVE################ #########CONCURRENTLY###
###############SHARE UPDATE EXCLUSIVE############################ALTER
TABLE ... DETACH PARTITION#########

################################ ##
##############

```
CREATE TABLE measurement_y2008m02 PARTITION OF measurement
    FOR VALUES FROM ('2008-02-01') TO ('2008-03-01')
    TABLESPACE fasttablespace;
```

#####[66.033755]## #
####ATTACH PARTITION##
##CREATE TABLE ... PARTITION OF######ACCESS EXCLUSIVE###############
#########SHARE UPDATE EXCLUSIVE##
CREATE TABLE ... LIKE################################
###############
################################### ##################################SHARE
##EXCLUSIVE##########CREATE##################EXCLUSIVE########### ##
#####OF########################### ALTER TABLE ... ATTACH PARTITION####
CREATE TABLE ... LIKE### ####:#

```
CREATE TABLE measurement_y2008m02
  (LIKE measurement INCLUDING DEFAULTS INCLUDING CONSTRAINTS)
  TABLESPACE fasttablespace;

ALTER TABLE measurement_y2008m02 ADD CONSTRAINT y2008m02
   CHECK ( logdate >= DATE '2008-02-01' AND logdate < DATE '2008-03-01' );

\copy measurement_y2008m02 from 'measurement_y2008m02'

-- #####################

ALTER TABLE measurement ATTACH PARTITION measurement_y2008m02
    FOR VALUES FROM ('2008-02-01') TO ('2008-03-01' );
```

###
EXCLUSIVE######### ## ####
############################## ##
###
#########################

#####[91.126761]###############DEFAULT#################################
#CHECK################ ##DEFAULT##
############ #######ACCESS EXCLUSIVE####DEFAULT############## DEFAULT####
###

#####[64.509804]## #########
##############################
#CONCURRENTLY############ ##########################CREATE INDEX ON
ONLY########## ## ##############
##CONCURRENTLY####################ALTER INDEX .. ATTACH PARTITION##
###########*attached*##### ##
####### ## #############
####################
###########1######CONCURRENTLY############## ######################## #####
#####CREATE######ON ONLY################### ############################
####################### #####CONCURRENTLY##########################ALTER
INDEX ... ATTACH PARTITION############################## ###############
#:#

```
CREATE INDEX measurement_usls_idx ON ONLY measurement (unitsales);

CREATE INDEX CONCURRENTLY measurement_usls_200602_idx
    ON measurement_y2006m02 (unitsales);
ALTER INDEX measurement_usls_idx
    ATTACH PARTITION measurement_usls_200602_idx;
...
```

######UNIQUE#PRIMARY KEY########## ########################### #######

```
ALTER TABLE ONLY measurement ADD UNIQUE (city_id, logdate);

ALTER TABLE measurement_y2006m02 ADD UNIQUE (city_id, logdate);
ALTER INDEX measurement_city_id_logdate_key
    ATTACH PARTITION measurement_y2006m02_city_id_logdate_key;
...
```

#########ALTER###
##
######### ##########################-###
#####3##############1##############
########

```
ALTER TABLE measurement
    MERGE PARTITIONS (measurement_y2006m01,
                      measurement_y2006m02,
                      measurement_y2006m03) INTO measurement_y2006q1;
```

#############
######1##
#####-## ############################
############ ##

```
ALTER TABLE measurement SPLIT PARTITION measurement_y2006q1 INTO
   (PARTITION measurement_y2006m01 FOR VALUES FROM ('2006-01-01') TO
 ('2006-02-01'),
    PARTITION measurement_y2006m02 FOR VALUES FROM ('2006-02-01') TO
 ('2006-03-01'),
    PARTITION measurement_y2006m03 FOR VALUES FROM ('2006-03-01') TO
 ('2006-04-01'));
```

5.12.2.3.

#########################

- ##
  #################### ##
  ##

- ##############################include########## ################################
  #&&########### ## #################
  #######include##############################

- INSERT#BEFORE ROW##

- ## ################################
  ############### ## ############
  ##

##
######################### ##
##########################
#####################################

###tableoid################5.11#
########################### ###############

- ########################### #######CREATE TABLE#########################
  #ALTER TABLE###################### #####ALTER TABLE ... ATTACH
  PARTITION##

- ###########CHECK###NOT NULL########################## ###########NO INHERIT######CHECK############# ############################NOT NULL### ######################
- ##ONLY################## ###### ###########ONLY######UNIQUE###PRIMARY KEY################## ########## ##
- ############################TRUNCATE ONLY################################ #####

5.12.3.

##

- ## ################### ########################
- ####################
- ### ################### ## ###########

5.12.3.1.

##########

1. #root########### ################### ################# ################### ######################### ################################### #######root### #############measurement######

   ```
   CREATE TABLE measurement (
       city_id         int not null,
       logdate         date not null,
       peaktemp        int,
       unitsales       int
   );
   ```
2. ####################root############### ###########root################## ### ##################################PostgreSQL#####################

   ```
   CREATE TABLE measurement_y2006m02 () INHERITS (measurement);
   CREATE TABLE measurement_y2006m03 () INHERITS (measurement);
   ...
   CREATE TABLE measurement_y2007m11 () INHERITS (measurement);
   CREATE TABLE measurement_y2007m12 () INHERITS (measurement);
   CREATE TABLE measurement_y2008m01 () INHERITS (measurement);
   ```
3. ##

   ###############

   ```
   CHECK ( x = 1 )
   CHECK ( county IN ( 'Oxfordshire', 'Buckinghamshire', 'Warwickshire' ))
   CHECK ( outletID >= 100 AND outletID < 200 )
   ```

   ### ##########################

```
CHECK ( outletID BETWEEN 100 AND 200 )
CHECK ( outletID BETWEEN 200 AND 300 )
```

###200################################### ###################

```
CREATE TABLE measurement_y2006m02 (
    CHECK ( logdate >= DATE '2006-02-01' AND logdate < DATE
 '2006-03-01' )
) INHERITS (measurement);

CREATE TABLE measurement_y2006m03 (
    CHECK ( logdate >= DATE '2006-03-01' AND logdate < DATE
 '2006-04-01' )
) INHERITS (measurement);

...
CREATE TABLE measurement_y2007m11 (
    CHECK ( logdate >= DATE '2007-11-01' AND logdate < DATE
 '2007-12-01' )
) INHERITS (measurement);

CREATE TABLE measurement_y2007m12 (
    CHECK ( logdate >= DATE '2007-12-01' AND logdate < DATE
 '2008-01-01' )
) INHERITS (measurement);

CREATE TABLE measurement_y2008m01 (
    CHECK ( logdate >= DATE '2008-01-01' AND logdate < DATE
 '2008-02-01' )
) INHERITS (measurement);
```

4. ##

```
CREATE INDEX measurement_y2006m02_logdate ON measurement_y2006m02
 (logdate);
CREATE INDEX measurement_y2006m03_logdate ON measurement_y2006m03
 (logdate);
CREATE INDEX measurement_y2007m11_logdate ON measurement_y2007m11
 (logdate);
CREATE INDEX measurement_y2007m12_logdate ON measurement_y2007m12
 (logdate);
CREATE INDEX measurement_y2008m01_logdate ON measurement_y2008m01
 (logdate);
```

5. ########INSERT INTO measurement ...################################## ########### root################################## ########################## #########################

```
CREATE OR REPLACE FUNCTION measurement_insert_trigger()
RETURNS TRIGGER AS $$
BEGIN
    INSERT INTO measurement_y2008m01 VALUES (NEW.*);
    RETURN NULL;
END;
$$
LANGUAGE plpgsql;
```

##############################

```
CREATE TRIGGER insert_measurement_trigger
    BEFORE INSERT ON measurement
```

```
    FOR EACH ROW EXECUTE FUNCTION measurement_insert_trigger();
```

#######################

############################
#######

```
CREATE OR REPLACE FUNCTION measurement_insert_trigger()
RETURNS TRIGGER AS $$
BEGIN
    IF ( NEW.logdate >= DATE '2006-02-01' AND
         NEW.logdate < DATE '2006-03-01' ) THEN
        INSERT INTO measurement_y2006m02 VALUES (NEW.*);
    ELSIF ( NEW.logdate >= DATE '2006-03-01' AND
            NEW.logdate < DATE '2006-04-01' ) THEN
        INSERT INTO measurement_y2006m03 VALUES (NEW.*);
    ...
    ELSIF ( NEW.logdate >= DATE '2008-01-01' AND
            NEW.logdate < DATE '2008-02-01' ) THEN
        INSERT INTO measurement_y2008m01 VALUES (NEW.*);
    ELSE
        RAISE EXCEPTION 'Date out of range.  Fix the
 measurement_insert_trigger() function!';
    END IF;
    RETURN NULL;
END;
$$
LANGUAGE plpgsql;
```

############ #####IF#########CHECK############################

##

##
################

##########################root########################### ############

```
CREATE RULE measurement_insert_y2006m02 AS
ON INSERT TO measurement WHERE
    ( logdate >= DATE '2006-02-01' AND logdate < DATE '2006-03-01' )
DO INSTEAD
    INSERT INTO measurement_y2006m02 VALUES (NEW.*);
...
CREATE RULE measurement_insert_y2008m01 AS
ON INSERT TO measurement WHERE
    ( logdate >= DATE '2008-01-01' AND logdate < DATE '2008-02-01' )
DO INSTEAD
    INSERT INTO measurement_y2008m01 VALUES (NEW.*);
```

##
############### ######################################

COPY#################### #######COPY#########root########################### #
#######COPY##################################

   ## ###########
   ########root#########
6. ######constraint_exclusion#postgresql.conf################# ###############
   #########

######################DDL######## ##################################DDL####
###################

5.12.3.2.

##################################

```
DROP TABLE measurement_y2006m02;
```

###

```
ALTER TABLE measurement_y2006m02 NO INHERIT measurement;
```

##

```
CREATE TABLE measurement_y2008m02 (
    CHECK ( logdate >= DATE '2008-02-01' AND logdate < DATE '2008-03-01' )
) INHERITS (measurement);
```

##############################
##############

```
CREATE TABLE measurement_y2008m02
  (LIKE measurement INCLUDING DEFAULTS INCLUDING CONSTRAINTS);
ALTER TABLE measurement_y2008m02 ADD CONSTRAINT y2008m02
   CHECK ( logdate >= DATE '2008-02-01' AND logdate < DATE '2008-03-01' );
\copy measurement_y2008m02 from 'measurement_y2008m02'

-- #####################
ALTER TABLE measurement_y2008m02 INHERIT measurement;
```

5.12.3.3.

##################################

- ####CHECK############################## ################################
  ########################
- ##
- ## #
  ############UPDATE#CHECK############## ############################
  UPDATE################################
- ####VACUUM####ANALYZE## ###
  ########

  ```
  ANALYZE measurement;
  ```

  root############
- ON CONFLICT####INSERT################### ON CONFLICT###################
  ##

- ##
  ## ##

5.12.4.

####

```
SET enable_partition_pruning = on;                 -- the default
SELECT count(*) FROM measurement WHERE logdate >= DATE '2008-01-01';
```

#######################measurement####################### #################
##WHERE##############################
##

EXPLAIN###########enable_partition_pruning ####################################
############### #####################################

```
SET enable_partition_pruning = off;
EXPLAIN SELECT count(*) FROM measurement WHERE logdate >= DATE
 '2008-01-01';
                                     QUERY PLAN
-----------------------------------------------------------------
----------------
 Aggregate  (cost=188.76..188.77 rows=1 width=8)
   ->  Append  (cost=0.00..181.05 rows=3085 width=0)
         ->  Seq Scan on measurement_y2006m02  (cost=0.00..33.12 rows=617
 width=0)
               Filter: (logdate >= '2008-01-01'::date)
         ->  Seq Scan on measurement_y2006m03  (cost=0.00..33.12 rows=617
 width=0)
               Filter: (logdate >= '2008-01-01'::date)
...
         ->  Seq Scan on measurement_y2007m11  (cost=0.00..33.12 rows=617
 width=0)
               Filter: (logdate >= '2008-01-01'::date)
         ->  Seq Scan on measurement_y2007m12  (cost=0.00..33.12 rows=617
 width=0)
               Filter: (logdate >= '2008-01-01'::date)
         ->  Seq Scan on measurement_y2008m01  (cost=0.00..33.12 rows=617
 width=0)
               Filter: (logdate >= '2008-01-01'::date)
```

#
#################
##############################

```
SET enable_partition_pruning = on;
EXPLAIN SELECT count(*) FROM measurement WHERE logdate >= DATE
 '2008-01-01';
                                     QUERY PLAN
-----------------------------------------------------------------
----------------
 Aggregate  (cost=37.75..37.76 rows=1 width=8)
   ->  Seq Scan on measurement_y2008m01  (cost=0.00..33.12 rows=617
 width=0)
         Filter: (logdate >= '2008-01-01'::date)
```

###########
################### ##
################################## ##############################

################################
################ ####PREPARE##
############# ############################

- ############ ################################## #######################
  #EXPLAIN#EXPLAIN ANALYZE################ EXPLAIN#####Subplans removed####
  ##

- ########### ## ##
  ### ######################
  ## ################
  ######EXPLAIN ANALYZE####loops################ ########################
  ############################### ####################(never executed)#####
  ######

##########enable_partition_pruning#################

5.12.5.

################################ ## ###
##########################

###########CHECK##
####################################
#########

########CHECK## ################
###

######constraint_exclusion################on##off#####partition########### #####
on#######################
##################CHECK#########

#######################

- ##

- #############WHERE##
  ##CURRENT_TIMESTAMP#####immutable##
  ##################

- ## ############
  ## ##########
  ##########################B-tree################################### ###########
  ####B-tree#######################

- ##################################### ################################## ##
  ###########################100############### ########################

5.12.6.

##################################### ##############################

##################################### ##############################WHERE#####
########################### ####################WHERE#######################
######## #######PRIMARY KEY####UNIQUE######################### ############
########################### ##
##

#

##
#################### ##################################

###################PostgreSQL################################### ####5.5.5#####
products###############orders##

```
DROP TABLE products;

ERROR:  cannot drop table products because other objects depend on it
DETAIL:  constraint orders_product_no_fkey on table orders depends on
 table products
HINT:  Use DROP ... CASCADE to drop the dependent objects too.
```

######################### #######################1##############

```
DROP TABLE products CASCADE;
```

#####orders#################### ##### ################################ #DROP ... CASCADE############ ##CASCADE######DROP#####DETAIL###########

PostgreSQL#######DROP#####CASCADE####### ################################## ##### ###CASCADE####RESTRICT############ ################################## ##############

##
##SQL###DROP#####RESTRICT###CASCADE################# ############### ##################RESTRICT#####CASCADE###################

DROP#####################CASCADE################################ ####DROP TABLE tab1, tab2#######tab2##tab1############CASCADE################## #########

##PostgreSQL########################## ## ###############

```
CREATE TYPE rainbow AS ENUM ('red', 'orange', 'yellow',
                             'green', 'blue', 'purple');

CREATE TABLE my_colors (color rainbow, note text);

CREATE FUNCTION get_color_note (rainbow) RETURNS text AS
  'SELECT note FROM my_colors WHERE color = $1'
  LANGUAGE SQL;
```

#SQL##############36.5########## PostgreSQL###get_color_note##rainbow## ########### ## ####PostgreSQL #get_color_note#####my_colors############ ########################## ##### ###################### ##

#####[94.400000]####SQL##########################SQL######################## ############################## #####################

```
CREATE FUNCTION get_color_note (rainbow) RETURNS text
```

```
BEGIN ATOMIC
  SELECT note FROM my_colors WHERE color = $1;
END;
```

###my_colors##########DROP###############

#6#

############
#################### ##

6.1.

###################### ##################################### #######1########
####1## ##################1####
############

###########INSERT################################## #####5##products#########
######

```
CREATE TABLE products (
    product_no integer,
    name text,
    price numeric
);
```

#########################

```
INSERT INTO products VALUES (1, 'Cheese', 9.99);
```

##############################

#######2##
############################

```
INSERT INTO products (product_no, name, price) VALUES (1, 'Cheese', 9.99);
INSERT INTO products (name, price, product_no) VALUES ('Cheese', 9.99, 1);
```

###############################

######################### ##################### #########

```
INSERT INTO products (product_no, name) VALUES (1, 'Cheese');
INSERT INTO products VALUES (1, 'Cheese');
```

###PostgreSQL######## ##

##

```
INSERT INTO products (product_no, name, price) VALUES (1, 'Cheese',
 DEFAULT);
INSERT INTO products DEFAULT VALUES;
```

######################

```
INSERT INTO products (product_no, name, price) VALUES
    (1, 'Cheese', 9.99),
    (2, 'Bread', 1.99),
    (3, 'Milk', 2.99);
```

###########0###1#########################

```
INSERT INTO products (product_no, name, price)
  SELECT product_no, name, price FROM new_products
    WHERE release_date = 'today';
```

##################SQL#########7##############

> **###**
>
> ################COPY################ INSERT######################### ##########################14.4##########

6.2.

##########################(update)######## ################################## ## ##########################

###########UPDATE############## #####3############

1. ############
2. #######
3. #####

#5#########SQL##########ID################## ############################## ### ######################### ## ##

#######5###########10#####################

```
UPDATE products SET price = 10 WHERE price = 5;
```

##########################1################## ############################## ####

##################### ###UPDATE################### ########################## ### ################### ##SET################################ ################ ############### ############10%##############

```
UPDATE products SET price = price * 1.10;
```

################################## #####WHERE######## WHERE################## ######## #########WHERE################## SET################WHERE############## ######### ######################### ########WHERE######################### ## #######################9###### ########################

UPDATE#####SET############################## #######

```
UPDATE mytable SET a = 5, b = 3, c = 1 WHERE a > 0;
```

6.3.

####SQL############################ ## ## ############################### ##

#########DELETE########### ###UPDATE############# ####products#########10###### #################

```
DELETE FROM products WHERE price = 10;
```

####################################

```
DELETE FROM products;
```

#############

6.4.

#####[89.655172]#################################### INSERT#UPDATE#DELETE# ################RETURNING################ RETURNING#################### ## ############# ################################### INSERT################################## #RETURNING###### RETURNING## #######################

RETURNING#########SELECT##########7.3######## ########################### ################### ##########RETURNING *########################

INSERT###RETURNING###################### ################################## ############## ### ###serial########## ###############RETURNING################IID##########

```
CREATE TABLE users (firstname text, lastname text, id serial primary key);
```

```
INSERT INTO users (firstname, lastname) VALUES ('Joe', 'Cool') RETURNING
 id;
```

###RETURNING##INSERT ... SELECT###########

UPDATE###RETURNING######################## #######

```
UPDATE products SET price = price * 1.10
  WHERE price <= 99.99
  RETURNING name, price AS new_price;
```

DELETE###RETURNING###################### #######

```
DELETE FROM products
  WHERE obsoletion_date = 'today'
  RETURNING *;
```

############RETURNING## ## ################################RETURNING*########################### ####### ######################## ############ ###:# ##

```
MERGE INTO products p USING new_products n ON p.product_no = n.product_no
  WHEN NOT MATCHED THEN INSERT VALUES (n.product_no, n.name, n.price)
  WHEN MATCHED THEN UPDATE SET name = n.name, price = n.price
  RETURNING p.*;
```

#############37########RETURNING######################## ################### #####RETURNING##############

#7#

########################
######

7.1.

SQL###SELECT##################
SELECT################

```
[WITH with_queries] SELECT select_list FROM table_expression
 [sort_specification]
```

#################################### WITH######################

#####################

```
SELECT * FROM table1;
```

table1##################table1##################### ##################
####### ##psql#######################
###########*###########################
################## ####table1#a#b#c##
##

```
SELECT a, b + c FROM table1;
```

#####b###c################### #######7.3#########

FROM table1#####################1###### ##################################
######## ######################SELECT####################

```
SELECT 3 * 4;
```

############################## #######################

```
SELECT random();
```

7.2.

#################### ######FROM####################WHERE##GROUP
BY##HAVING############ ### ########
#####################################

###########WHERE##GROUP BY#####HAVING###FROM##############################
################1########

7.2.1. FROM#

FROM#########################1##########1############

FROM *table_reference* [, *table_reference* [, ...]]

##JOIN##############################
####### FROM### FROM######
#WHERE##GROUP BY#####HAVING##

#######################################ONLY################################
#################### ##

########ONLY#################*################################ #############
#################################### ############################

7.2.1.1.

########2## ########################
####################

T1 join_type T2 [*join_condition*]

#################################### *T1#T2*############################## ##
#JOIN######################## ########JOIN##############

#####

#####

T1 CROSS JOIN *T2*

T1###*T2*###*T1*#########*T2*##############
########N##M########################### N * M

FROM *T1* CROSS JOIN *T2* # FROM *T1* INNER JOIN *T2* ON TRUE #############
FROM *T1*, *T2*

> ##
>
> ##################################### #####JOIN################## ###
> FROM *T1* CROSS JOIN *T2* INNER JOIN *T3* ON *condition* # FROM *T1*, *T2*
> INNER JOIN *T3* ON *condition* ########## ###########*condition*#*T1*#
> ########2#############

######

T1 { [INNER] | { LEFT | RIGHT | FULL } [OUTER] } JOIN *T2*
 ON *boolean_expression*
T1 { [INNER] | { LEFT | RIGHT | FULL } [OUTER] } JOIN *T2* USING (*join*
 column list)
T1 NATURAL { [INNER] | { LEFT | RIGHT | FULL } [OUTER] } JOIN *T2*

INNER#OUTER######## INNER########### LEFT#RIGHT#FULL###########

######ON##USING###########NATURAL############ ####################2######
#####################

##################

INNER JOIN######

T1###R1#####T2#####R1################################

LEFT OUTER JOIN#######

############# ####T2###############T1########T2###NULL##############
#################T1###########1########

RIGHT OUTER JOIN#######

############# ####T1##############T2########T1###NULL##############
########### ########T2#########

FULL OUTER JOIN########

############# ####T2##############T1########T2###NULL##############
####T1###############T2########T1###NULL##############

ON###############WHERE#################### ON###########*T1*###*T2*#######
######

USING## ###################
#################################### ###, *T1*#*T2*#USING (a, b)############ON
T1.a = *T2*.a AND *T1*.b = *T2*.b#############

####JOIN USING## JOIN ON # *T1*
############## *T2* ############## JOIN USING##############################
########*T1*#########*T2*###########

#####[73.264781]#####NATURAL#USING################################
###USING######### USING######################## #############NATURAL
JOIN#JOIN ... ON TRUE################### #########NATURAL#USING###
####### ################################USING######### USING############
#####1######### ################################

##
USING#################################### NATURAL##USING######## ### ##

####### #######t1

```
 num | name
-----+------
   1 | a
   2 | b
   3 | c
```

########t2

```
 num | value
-----+-------
   1 | xxx
   3 | yyy
```

```
   5 | zzz
```

##############################

```
=> SELECT * FROM t1 CROSS JOIN t2;
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
   1 | a    |   3 | yyy
   1 | a    |   5 | zzz
   2 | b    |   1 | xxx
   2 | b    |   3 | yyy
   2 | b    |   5 | zzz
   3 | c    |   1 | xxx
   3 | c    |   3 | yyy
   3 | c    |   5 | zzz
(9 rows)

=> SELECT * FROM t1 INNER JOIN t2 ON t1.num = t2.num;
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
   3 | c    |   3 | yyy
(2 rows)

=> SELECT * FROM t1 INNER JOIN t2 USING (num);
 num | name | value
-----+------+-------
   1 | a    | xxx
   3 | c    | yyy
(2 rows)

=> SELECT * FROM t1 NATURAL INNER JOIN t2;
 num | name | value
-----+------+-------
   1 | a    | xxx
   3 | c    | yyy
(2 rows)

=> SELECT * FROM t1 LEFT JOIN t2 ON t1.num = t2.num;
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
   2 | b    |     |
   3 | c    |   3 | yyy
(3 rows)

=> SELECT * FROM t1 LEFT JOIN t2 USING (num);
 num | name | value
-----+------+-------
   1 | a    | xxx
   2 | b    |
   3 | c    | yyy
(3 rows)

=> SELECT * FROM t1 RIGHT JOIN t2 ON t1.num = t2.num;
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
   3 | c    |   3 | yyy
     |      |   5 | zzz
(3 rows)

=> SELECT * FROM t1 FULL JOIN t2 ON t1.num = t2.num;
```

```
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
   2 | b    |     |
   3 | c    |   3 | yyy
     |      |   5 | zzz
(4 rows)
```

ON################################## ################################## ####
###

```
=> SELECT * FROM t1 LEFT JOIN t2 ON t1.num = t2.num AND t2.value = 'xxx';
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
   2 | b    |     |
   3 | c    |     |
(3 rows)
```

WHERE##############################

```
=> SELECT * FROM t1 LEFT JOIN t2 ON t1.num = t2.num WHERE t2.value =
 'xxx';
 num | name | num | value
-----+------+-----+-------
   1 | a    |   1 | xxx
(1 row)
```

#####ON##################WHERE######################## ####################
#############

7.2.1.2.

#####
##########

#######################

```
FROM table_reference AS alias
```

####

```
FROM table_reference alias
```

AS################### *alias*############

#######

```
SELECT * FROM some_very_long_table_name s JOIN another_fairly_long_name a
 ON s.id = a.num;
```

###################################### ###################################### ##
###############

```
SELECT * FROM my_table AS m WHERE my_table.a > 5;    -- ###
```

######################### #########1##########################

```
SELECT * FROM people AS mother JOIN people AS child ON mother.id =
 child.mother_id;
```

################## ##########2###my_table#######b############2############
###########

```
SELECT * FROM my_table AS a CROSS JOIN my_table AS b ...
SELECT * FROM (my_table AS a CROSS JOIN my_table) AS b ...
```

##

```
FROM table_reference [AS] alias ( column1 [, column2 [, ...]] )
```

###########################

###JOIN###############JOIN################ #########

```
SELECT a.* FROM my_table AS a JOIN your_table AS b ON ...
```

####SQL####

```
SELECT a.* FROM (my_table AS a JOIN your_table AS b ON ...) AS c
```

########## #######a####c################

7.2.1.3.

############################## #######################7.2.1.2###################
######

```
FROM (SELECT * FROM table1) AS alias_name
```

####FROM table1 AS alias_name###### ######################################
##############

#########VALUES#############

```
FROM (VALUES ('anne', 'smith'), ('bob', 'jones'), ('joe', 'blow'))
     AS names(first, last)
```

################### VALUES### ###7.7######
####

SQL################################ PostgreSQL###AS#########################
##SQL###################

7.2.1.4.

##########FROM#######
################### ####################################
##SELECT#JOIN#WHERE############

#######ROWS FROM################################## ########################
######################null#########

```
function_call [WITH ORDINALITY] [[AS] table_alias [(column_alias
 [, ... ])]]
```

```
ROWS FROM( function_call [, ... ] ) [WITH ORDINALITY] [[AS] table_alias
 [(column_alias [, ... ])]]
```

WITH ORDINALITY###################bigint########## ###########1####### ## ####SQL###UNNEST ... WITH ORDINALITY####### ###########(ordinal)## #ordinality#############AS###############

#########UNNEST######################## ####################UNNEST(9.19)#### #####ROWS FROM######################

```
UNNEST( array_expression [, ... ] ) [WITH ORDINALITY] [[AS] table_alias
 [(column_alias [, ... ])]]
```

table_alias###########################ROWS FROM()################

##

##########

```
CREATE TABLE foo (fooid int, foosubid int, fooname text);

CREATE FUNCTION getfoo(int) RETURNS SETOF foo AS $$
    SELECT * FROM foo WHERE fooid = $1;
$$ LANGUAGE SQL;

SELECT * FROM getfoo(1) AS t1;

SELECT * FROM foo
    WHERE foosubid IN (
                        SELECT foosubid
                        FROM getfoo(foo.fooid) z
                        WHERE z.fooid = foo.fooid
                      );

CREATE VIEW vw_getfoo AS SELECT * FROM getfoo(1);

SELECT * FROM vw_getfoo;
```

###################OUT######### #record################### ## ###

```
function_call [AS] alias (column_definition [, ... ])
function_call AS [alias] (column_definition [, ... ])
ROWS FROM( ... function_call AS (column_definition [, ... ]) [, ... ] )
```

ROWS FROM()###########*column_definition*#####FROM#################### ##### ################### ROWS FROM()##########*column_definition*# ######################## ########1#########WITH ORDINALITY###### ##*column_definition*#####ROWS FROM()######################

##########

```
SELECT *
    FROM dblink('dbname=mydb', 'SELECT proname, prosrc FROM pg_proc')
      AS t1(proname name, prosrc text)
    WHERE proname LIKE 'bytea%';
```

dblink###dblink####################### ###################record############ ## ##############*##

ROWS FROM######:

```
SELECT *
FROM ROWS FROM
    (
        json_to_recordset('[{"a":40,"b":"foo"},{"a":"100","b":"bar"}]')
            AS (a INTEGER, b TEXT),
        generate_series(1, 3)
    ) AS x (p, q, s)
ORDER BY p;

  p  |  q  | s
-----+-----+---
  40 | foo | 1
 100 | bar | 2
     |     | 3
```

2########1##FROM######### json_to_recordset()##2###(###integer#2###text)# ########### generate_series()############ ORDER BY####################

7.2.1.5. LATERAL

FROM#################LATERAL########## ################FROM############### #### #LATERAL################################FROM#############

FROM##################LATERAL################################ ############### ###### FROM########################

LATERAL###FROM###########JOIN################# ###########JOIN############## #########

FROM###LATERAL###################### #################FROM################ ######FROM##########LATERAL######################### ###################### ##### ##

LATERAL##################

```
SELECT * FROM foo, LATERAL (SELECT * FROM bar WHERE bar.id = foo.bar_id)
 ss;
```

##

```
SELECT * FROM foo, bar WHERE bar.id = foo.bar_id;
```

LATERAL## ############################ ###### ####vertices(polygon)## ###########

```
SELECT p1.id, p2.id, v1, v2
FROM polygons p1, polygons p2,
     LATERAL vertices(p1.poly) v1,
     LATERAL vertices(p2.poly) v2
WHERE (v1 <-> v2) < 10 AND p1.id != p2.id;
```

########################

```
SELECT p1.id, p2.id, v1, v2
FROM polygons p1 CROSS JOIN LATERAL vertices(p1.poly) v1,
     polygons p2 CROSS JOIN LATERAL vertices(p2.poly) v2
WHERE (v1 <-> v2) < 10 AND p1.id != p2.id;
```

#################### ##########LATERAL#################################
###

LATERAL######LEFT JOIN################## ###LATERAL#####################
################## #####get_product_names()##############################
##

```
SELECT m.name
FROM manufacturers m LEFT JOIN LATERAL get_product_names(m.id) pname ON
 true
WHERE pname IS NULL;
```

7.2.2. WHERE#

WHERE############

WHERE *search_condition*

####*search_condition*##boolean############4.2###########

FROM################################## ##################### ################
#NULL################ #########FROM##################### ################
##WHERE###############

> ##
>
> ##########WHERE###JOIN############## #################
>
> FROM a, b WHERE a.id = b.id AND b.val > 5
>
> ###
>
> FROM a INNER JOIN b ON (a.id = b.id) WHERE b.val > 5
>
> ###########
>
> FROM a NATURAL JOIN b WHERE b.val > 5
>
> ################## FROM##JOIN###SQL################SQL############
> ############ #########FROM############### #####ON####USING###WHERE###
> ######### ##

WHERE###########

```
SELECT ... FROM fdt WHERE c1 > 5

SELECT ... FROM fdt WHERE c1 IN (1, 2, 3)

SELECT ... FROM fdt WHERE c1 IN (SELECT c1 FROM t2)

SELECT ... FROM fdt WHERE c1 IN (SELECT c3 FROM t2 WHERE c2 = fdt.c1 + 10)

SELECT ... FROM fdt WHERE c1 BETWEEN (SELECT c3 FROM t2 WHERE c2 = fdt.c1
 + 10) AND 100

SELECT ... FROM fdt WHERE EXISTS (SELECT c1 FROM t2 WHERE c2 > fdt.c1)
```

fdt#FROM############# WHERE################fdt######## ################## ######### ################################### #############fdt############# #### c1#fdt.c1###########c1################################## ############## ################# ##

7.2.3. GROUP BY##HAVING#

WHERE####################GROUP BY###########HAVING#######################
##

```
SELECT select_list
    FROM ...
    [WHERE ...]
    GROUP BY grouping_column_reference [, grouping_column_reference]...
```

GROUP BY## ############## ########### ## ################################## ########### #########

```
=> SELECT * FROM test1;
 x | y
---+---
 a | 3
 c | 2
 b | 5
 a | 1
(4 rows)

=> SELECT x FROM test1 GROUP BY x;
 x
---
 a
 b
 c
(3 rows)
```

2###########SELECT * FROM test1 GROUP BY x########### #############y##### ######## GROUP BY##

#######################GROUP BY################################ ##############

```
=> SELECT x, sum(y) FROM test1 GROUP BY x;
 x | sum
---+-----
 a |   4
 b |   5
 c |   2
(3 rows)
```

###sum() ########################### ##################9.21#########

> ###
>
> ##################################### ###DISTINCT##7.3.3################ #########

######### ################ ################

```
SELECT product_id, p.name, (sum(s.units) * p.price) AS sales
    FROM products p LEFT JOIN sales s USING (product_id)
    GROUP BY product_id, p.name, p.price;
```

######product_id##p.name##p.price####GROUP BY############# ############## ########################### s.units##GROUP BY############# ################ #sum(...)############## ################################

products#########product_id##################name#price####ID#product_id##### ###########ID##############name#price###########################product_id## #############

###SQL###GROUP BY###########################PostgreSQL###GROUP BY######## ###################### ############################

GROUP BY##WHERE#####HAVING#### ######## ###########

```
SELECT select_list FROM ... [WHERE ...] GROUP BY ...
 HAVING boolean_expression
```

HAVING###

#######

```
=> SELECT x, sum(y) FROM test1 GROUP BY x HAVING sum(y) > 3;
 x | sum
---+-----
 a |   4
 b |   5
(2 rows)

=> SELECT x, sum(y) FROM test1 GROUP BY x HAVING x < 'c';
 x | sum
---+-----
 a |   4
 b |   5
(2 rows)
```

################

```
SELECT product_id, p.name, (sum(s.units) * (p.price - p.cost)) AS profit
    FROM products p LEFT JOIN sales s USING (product_id)
    WHERE s.date > CURRENT_DATE - INTERVAL '4 weeks'
    GROUP BY product_id, p.name, p.price, p.cost
    HAVING sum(p.price * s.units) > 5000;
```

#####WHERE#################################4###################HAVING##### ####5000################ ##################################

##################GROUP BY######################### ##############HAVING#### ############################ HAVING#################GROUP BY############## #######

7.2.4. GROUPING SETS#CUBE#ROLLUP

FROM##WHERE#################### ######################GROUP BY#########################

```
=> SELECT * FROM items_sold;
 brand | size | sales
-------+------+-------
 Foo   | L    |  10
 Foo   | M    |  20
 Bar   | M    |  15
 Bar   | L    |  5
(4 rows)

=> SELECT brand, size, sum(sales) FROM items_sold GROUP BY GROUPING SETS
 ((brand), (size), ());
 brand | size | sum
-------+------+-----
 Foo   |      |  30
 Bar   |      |  20
       | L    |  15
       | M    |  35
       |      |  50
(5 rows)
```

GROUPING SETS################################ ########GROUP BY#########
######### ## #######
#GROUP BY################

##NULL########### ####################
######## 9.64#########

##

```
ROLLUP ( e1, e2, e3, ... )
```

##############

```
GROUPING SETS (
    ( e1, e2, e3, ... ),
    ...
    ( e1, e2 ),
    ( e1 ),
    ( )
)
```

###

```
CUBE ( e1, e2, ... )
```

#####

```
CUBE ( a, b, c )
```

#########

```
GROUPING SETS (
    ( a, b, c ),
    ( a, b    ),
    ( a,    c ),
    ( a       ),
    (    b, c ),
    (    b    ),
    (       c ),
```

```
    (         )
)
```

CUBE##ROLLUP## ########################
######################### ###

```
CUBE ( (a, b), (c, d) )
```

#########

```
GROUPING SETS (
    ( a, b, c, d ),
    ( a, b       ),
    (       c, d ),
    (            )
)
```

###

```
ROLLUP ( a, (b, c), d )
```

#########

```
GROUPING SETS (
    ( a, b, c, d ),
    ( a, b, c    ),
    ( a          ),
    (            )
)
```

CUBE#ROLLUP####GROUP BY###########GROUPING SETS############ GROUPING SETS###

#####[89.361702]#########GROUP BY## ### ##########################BY## # ###:#

```
GROUP BY a, CUBE (b, c), GROUPING SETS ((d), (e))
```

#########

```
GROUP BY GROUPING SETS (
    (a, b, c, d), (a, b, c, e),
    (a, b, d),    (a, b, e),
    (a, c, d),    (a, c, e),
    (a, d),       (a, e)
)
```

 ### ###

```
GROUP BY ROLLUP (a, b), ROLLUP (a, c)
```

#########

```
GROUP BY GROUPING SETS (
    (a, b, c),
    (a, b),
    (a, b),
```

```
    (a, c),
    (a),
    (a),
    (a, c),
    (a),
    ()
)
```

################GROUP BY###DISTINCT############ ######

```
GROUP BY DISTINCT ROLLUP (a, b), ROLLUP (a, c)
```

#########

```
GROUP BY GROUPING SETS (
    (a, b, c),
    (a, b),
    (a, c),
    (a),
    ()
)
```

####SELECT DISTINCT################ ################### ################
NULL##########################NULL########

##
(a, b)########################## GROUP BY##########################(a, b)################### ################################ROW(a, b)### ######

7.2.5.

#############3.5#9.22#4.2.8############################HAVING###############
################GROUP BY###HAVING#####################FROM/WHERE#####
###################

#######################################PARTITION BY###ORDER BY#############
#################################### ######ORDER BY#######################
############ #######PARTITION BY###ORDER BY#################### #########
##############################ORDER BY##################################

###PARTITION BY/ORDER BY##
################# ##################### ####################################
##ORDER BY######

7.3.

#########SELECT### ########
################### ##############################

7.3.1.

###################*############## ########################4.2################
#########################

```
SELECT a, b, c FROM ...
```

a#b#c#######FROM#######################7.2.1.2################## ##############
##############WHERE####### #############HAVING#########

```
###2#################################################
```

```
SELECT tbl1.a, tbl2.a, tbl1.b FROM ...
```

```
###########################################
```

```
SELECT tbl1.*, tbl2.a FROM ...
```

table_name.*###############8.16.5#########

```
#######################################################  ######################
################## ############FROM##########################################
#########
```

7.3.2.

```
###########ORDER BY#########################################################
######
```

```
SELECT a AS value, b + c AS sum FROM ...
```

AS#################################### ##################### ################# ##
#########################

AS######################PostgreSQL##########################AS#################
######## ###C#########AS######################## ####FROM############1#########
####

```
SELECT a from, b + c AS sum FROM ...
```

```
###############
```

```
SELECT a AS from, b + c AS sum FROM ...
SELECT a "from", b + c AS sum FROM ...
```

#################################AS###########################

> ##
>
> ##############FROM########7.2.1.2########### #######2##################
> ####################

7.3.3. DISTINCT

```
#################################### ############SELECT####DISTINCT#######
#####
```

```
SELECT DISTINCT select_list ...
```

#DISTINCT#####ALL###

#####1##################2############ NULL#######################

##################################

```
SELECT DISTINCT ON (expression [, expression ...]) select_list ...
```

###*expression*#################### ### ####### DISTINCT## ################### #DISTINCT ON####ORDER BY################

DISTINCT ON####SQL####### ### GROUP BY#FROM################################DISTINCT ON##################

7.4. #########(UNION, INTERSECT, EXCEPT)

2#################################### ###########

```
query1 UNION [ALL] query2
query1 INTERSECT [ALL] query2
query1 EXCEPT [ALL] query2
```

query1#*query2*##################################

UNION##*query2*####*query1*################################### ####UNION ALL# #######DISTINCT######################

INTERSECT##*query1*####*query2*####################### INTERSECT ALL######## #############

EXCEPT##*query1*###############*query2*####################### ####2########### ############# ######EXCEPT ALL ######################

2#########################2########union############## ######################### ############10.5###############

############################

```
query1 UNION query2 EXCEPT query3
```

###########

```
(query1 UNION query2) EXCEPT query3
```

############################ ########UNION#EXCEPT###############INTERSECT## ###2############### ####

```
query1 UNION query2 INTERSECT query3
```

##########

```
query1 UNION (query2 INTERSECT query3)
```

###*query*############# ####*query*#LIMIT#################################### ## ####################################1########################## ########### ####

```
SELECT a FROM b UNION SELECT x FROM y LIMIT 10
```

####################

```
(SELECT a FROM b UNION SELECT x FROM y) LIMIT 10
```

##########

```
SELECT a FROM b UNION (SELECT x FROM y LIMIT 10)
```

7.5. ######(ORDER BY)

########1## ########################### ### ## ################ ######### ###########################

ORDER BY###############

```
SELECT select_list
    FROM table_expression
    ORDER BY sort_expression1 [ASC | DESC] [NULLS { FIRST | LAST }]
             [, sort_expression2 [ASC | DESC] [NULLS { FIRST |
 LAST }] ...]
```

#####(###)################################## ##########

```
SELECT a, b FROM table1 ORDER BY a + b, c;
```

############ASC####DESC########### ########################## ASC######### ################## ############<###### ######## #######>########### [1]

NULLS FIRST###NULLS LAST####################NULL###NULL########## ############## ########NULL##########NULL################## #######NULLS FIRST#DESC##################NULLS LAST###

####ORDER BY x, y DESC##ORDER BY x DESC, y DESC########ORDER BY x ASC, y DESC#######

sort_expression################################

```
SELECT a + b AS sum, c FROM table1 ORDER BY sum;
SELECT a, max(b) FROM table1 GROUP BY a ORDER BY 1;
```

################### ## ############

```
SELECT a + b AS sum, c FROM table1 ORDER BY sum + c;          -- ###
```

################# ORDER BY## # ################ AS##

ORDER BY##UNION#INTERSECT#EXCEPT#################### #################### ###########################

7.6. LIMIT#OFFSET

[1] ###PostgreSQL##ASC#DESC##########################*B-tree*############ ##########<#>###################### ## ################################

LIMIT###OFFSET#####################################

```
SELECT select_list
    FROM table_expression
    [ ORDER BY ... ]
    [ LIMIT { count | ALL } ]
    [ OFFSET start ]
```

##(limit)###
LIMIT ALL##LIMIT#############LIMIT####NULL########

OFFSET#################### OFFSET 0##OFFSET#############OFFSET####NULL#
########

OFFSET###LIMIT###########OFFSET##############LIMIT########

LIMIT####################ORDER BY############# ORDER BY#################
#################### 10####20##############################10####20#########
ORDER BY##################

##########################LIMIT###### #####LIMIT#OFFSET##################
######################### ####1############################LIMIT/OFFSET####
#####ORDER BY############################## ############# ORDER BY#########
#####SQL#####################################

OFFSET############################## #########OFFSET###############

7.7. VALUES###

VALUES####################### ### ####
#######

```
VALUES ( expression [, ...] ) [, ...]
```

############################## ################################## ###########
################### ####################UNION################ #10.5#######
####

#########

```
VALUES (1, 'one'), (2, 'two'), (3, 'three');
```

####2#3########### #############

```
SELECT 1 AS column1, 'one' AS column2
UNION ALL
SELECT 2, 'two'
UNION ALL
SELECT 3, 'three';
```

########PostgreSQL#VALUES#######column1#column2########### ##SQL#########
####################################
#############

```
=> SELECT * FROM (VALUES (1, 'one'), (2, 'two'), (3, 'three')) AS t
 (num,letter);
 num | letter
-----+--------
```

```
   1 | one
   2 | two
   3 | three
(3 rows)
```

######VALUES##########################

```
SELECT select_list FROM table_expression
```

####SELECT############################ ####UNION################ ##*sort_specification* (ORDER BY#LIMIT#OFFSET)########### VALUES#INSERT## ###################### ######################

####VALUES#########

7.8. WITH###############

#####[91.376912]#WITH################################## ############(Common Table Expressions)###CTE## WITH# #########SELECT#INSERT#UPDATE###DELETE########## ###WITH###### #SELECT#INSERT#UPDATE#DELETE####MERGE##################

7.8.1. WITH##SELECT

WITH##SELECT################################## #########

```
WITH regional_sales AS (
    SELECT region, SUM(amount) AS total_sales
    FROM orders
    GROUP BY region
), top_regions AS (
    SELECT region
    FROM regional_sales
    WHERE total_sales > (SELECT SUM(total_sales)/10 FROM regional_sales)
)
SELECT region,
       product,
       SUM(quantity) AS product_units,
       SUM(amount) AS product_sales
FROM orders
WHERE region IN (SELECT region FROM top_regions)
GROUP BY region, product;
```

############region################### WITH###regional_sales#top_regions#### ############## ####regional_sales####top_regions##### ##top_regions#SELECT############# #### WITH##################sub-SELECT### ########################

7.8.2.

######RECURSIVE#####WITH#######################SQL############### RECURSIVE#######WITH##########################1##100####################### ########

```
WITH RECURSIVE t(n) AS (
    VALUES (1)
  UNION ALL
    SELECT n+1 FROM t WHERE n < 100
)
```

```
SELECT sum(n) FROM t;
```

###WITH#####################*non-recursive term*#####UNION####UNION ALL#########
#recursive term#### ################################## #######################

###########

1. ############ UNION####UNION ALL################## ######################
   ##########################

2. ############################

   a. ################################## UNION####UNION ALL#############
      ################## ##

   b. ####################################

##
RECURSIVE###########(recursively)###########################(iteratively)#### ####

#####################################1##100########### 100#######WHERE##########
##############

####################################
#########################

```
WITH RECURSIVE included_parts(sub_part, part, quantity) AS (
    SELECT sub_part, part, quantity FROM parts WHERE part = 'our_product'
  UNION ALL
    SELECT p.sub_part, p.part, p.quantity * pr.quantity
    FROM included_parts pr, parts p
    WHERE p.part = pr.sub_part
)
SELECT sub_part, SUM(quantity) as total_quantity
FROM included_parts
GROUP BY sub_part
```

7.8.2.1.

############
##################################SQL#
################## ######################################

####link#############tree###########
######

```
WITH RECURSIVE search_tree(id, link, data) AS (
    SELECT t.id, t.link, t.data
    FROM tree t
  UNION ALL
    SELECT t.id, t.link, t.data
    FROM tree t, search_tree st
    WHERE t.id = st.link
)
SELECT * FROM search_tree;
```

#############################

```
WITH RECURSIVE search_tree(id, link, data, path) AS (
    SELECT t.id, t.link, t.data, ARRAY[t.id]
    FROM tree t
  UNION ALL
    SELECT t.id, t.link, t.data, path || t.id
    FROM tree t, search_tree st
    WHERE t.id = st.link
)
SELECT * FROM search_tree ORDER BY path;
```

##########f1#f2################
#####

```
WITH RECURSIVE search_tree(id, link, data, path) AS (
    SELECT t.id, t.link, t.data, ARRAY[ROW(t.f1, t.f2)]
    FROM tree t
  UNION ALL
    SELECT t.id, t.link, t.data, path || ROW(t.f1, t.f2)
    FROM tree t, search_tree st
    WHERE t.id = st.link
)
SELECT * FROM search_tree ORDER BY path;
```

###
##############################ROW()######### ########################### ####

######################################

```
WITH RECURSIVE search_tree(id, link, data, depth) AS (
    SELECT t.id, t.link, t.data, 0
    FROM tree t
  UNION ALL
    SELECT t.id, t.link, t.data, depth + 1
    FROM tree t, search_tree st
    WHERE t.id = st.link
)
SELECT * FROM search_tree ORDER BY depth;
```

##################2##############

###
################################## ################################ ##### ##

################################### ####

```
WITH RECURSIVE search_tree(id, link, data) AS (
    SELECT t.id, t.link, t.data
    FROM tree t
  UNION ALL
    SELECT t.id, t.link, t.data
    FROM tree t, search_tree st
```

```
    WHERE t.id = st.link
) SEARCH DEPTH FIRST BY id SET ordercol
SELECT * FROM search_tree ORDER BY ordercol;

WITH RECURSIVE search_tree(id, link, data) AS (
    SELECT t.id, t.link, t.data
    FROM tree t
  UNION ALL
    SELECT t.id, t.link, t.data
    FROM tree t, search_tree st
    WHERE t.id = st.link
) SEARCH BREADTH FIRST BY id SET ordercol
SELECT * FROM search_tree ORDER BY ordercol;
```

################################## SEARCH#################################### ############################## #####CTE###############

7.8.2.2.

```
################################################  ###########################
## UNION ALL#####UNION########################################### ############
##########################1################################################ ##
#################################### ####link############graph##############
######


WITH RECURSIVE search_graph(id, link, data, depth) AS (
    SELECT g.id, g.link, g.data, 0
    FROM graph g
  UNION ALL
    SELECT g.id, g.link, g.data, sg.depth + 1
    FROM graph g, search_graph sg
    WHERE g.id = sg.link
)
SELECT * FROM search_graph;
```

#######link################## #depth############UNION ALL#UNION########### ############## ######link## ############## ##path#cycle########

```
WITH RECURSIVE search_graph(id, link, data, depth, is_cycle, path) AS (
    SELECT g.id, g.link, g.data, 0,
      false,
      ARRAY[g.id]
    FROM graph g
  UNION ALL
    SELECT g.id, g.link, g.data, sg.depth + 1,
      g.id = ANY(path),
      path || g.id
    FROM graph g, search_graph sg
    WHERE g.id = sg.link AND NOT is_cycle
)
SELECT * FROM search_graph;
```

#######################path# ############################

#########f1#f2#############

```
WITH RECURSIVE search_graph(id, link, data, depth, is_cycle, path) AS (
    SELECT g.id, g.link, g.data, 0,
      false,
```

```
      ARRAY[ROW(g.f1, g.f2)]
    FROM graph g
  UNION ALL
    SELECT g.id, g.link, g.data, sg.depth + 1,
      ROW(g.f1, g.f2) = ANY(path),
      path || ROW(g.f1, g.f2)
    FROM graph g, search_graph sg
    WHERE g.id = sg.link AND NOT is_cycle
)
SELECT * FROM search_graph;
```

> ###
>
> ##ROW()######### ################
> ##############

########################## ##########################

```
WITH RECURSIVE search_graph(id, link, data, depth) AS (
    SELECT g.id, g.link, g.data, 1
    FROM graph g
  UNION ALL
    SELECT g.id, g.link, g.data, sg.depth + 1
    FROM graph g, search_graph sg
    WHERE g.id = sg.link
) CYCLE id SET is_cycle USING path
SELECT * FROM search_graph;
```

######################## CYCLE###
######################## ###########CTE###############

> ###
>
> ## #######SEARCH##CYCLE#########
> ##CYCLE########################## ##
> ###########SEARCH#CYCLE#############

##LIMIT####### #############LIMIT#####
#########

```
WITH RECURSIVE t(n) AS (
    SELECT 1
  UNION ALL
    SELECT n+1 FROM t
)
SELECT n FROM t LIMIT 100;
```

##########PostgreSQL##########################WITH################## #########
################ ########################### ################################
################################### ##################WITH##################
#########

7.8.3.

###WITH######################WITH##
################################WITH##################

######################## ################################WITH###############
################ ####WITH########1################################### ######
#WITH## ##########################
#########################

####WITH#########################volatile########SELECT#####################
##2##################### ##############WITH######1###################WITH#
#####2################# ############MATERIALIZED#####WITH################
##NOT MATERIALIZED##################### ########WITH####################
##WITH############WITH################################

###################

```
WITH w AS (
    SELECT * FROM big_table
)
SELECT * FROM w WHERE key = 123;
```

##WITH#############################

```
SELECT * FROM big_table WHERE key = 123;
```

###key###########key = 123############################## ####

```
WITH w AS (
    SELECT * FROM big_table
)
SELECT * FROM w AS w1 JOIN w AS w2 ON w1.key = w2.ref
WHERE w2.key = 123;
```

##WITH############big_table##
##############################

```
WITH w AS NOT MATERIALIZED (
    SELECT * FROM big_table
)
SELECT * FROM w AS w1 JOIN w AS w2 ON w1.key = w2.ref
WHERE w2.key = 123;
```

###########big_table###################

NOT MATERIALIZED###############

```
WITH w AS (
    SELECT key, very_expensive_function(val) as f FROM some_table
)
SELECT * FROM w AS w1 JOIN w AS w2 ON w1.f = w2.f;
```

####WITH###########very_expensive_function##############################
#####

#####SELECT#####WITH######### ########INSERT#UPDATE#DELETE###MERGE######
######### ##

7.8.4. WITH########

#####[86.842105]############INSERT#UPDATE#DELETE########MERGE########
##WITH######## ################################ ######WITH######-########
#INSERT#################### ########################### ###:###

```
WITH moved_rows AS (
    DELETE FROM products
    WHERE
        "date" >= '2010-10-01' AND
        "date" < '2010-11-01'
    RETURNING *
)
INSERT INTO products_log
SELECT * FROM moved_rows;
```

##########products##products_log######## WITH##DELETE#products######## #####RETURNING############### ######################products_log#######

############WITH##INSERT##sub-SELECT#####INSERT############ ############### ########WITH################## #######WITH################sub-SELECT##WITH### #############

##########WITH##########RETURNING##6.4######### ########################### #######RETURNING################################ WITH########RETURNING### ## ##################### ######### ########

```
WITH t AS (
    DELETE FROM foo
)
DELETE FROM bar;
```

####foo#####bar################# ######################bar################

######################## ##############WITH############################ ######## ##

```
WITH RECURSIVE included_parts(sub_part, part) AS (
    SELECT sub_part, part FROM parts WHERE part = 'our_product'
  UNION ALL
    SELECT p.sub_part, p.part
    FROM included_parts pr, parts p
    WHERE p.part = pr.sub_part
)
DELETE FROM parts
  WHERE part IN (SELECT part FROM included_parts);
```

##################################

WITH## ##### #########################SELECT##########WITH##SELECT################### ##

WITH########################## #####WITH#################################### ####### #################13############## ################################### ##############################RETURNING######WITH############################ ####### ############

```
WITH t AS (
    UPDATE products SET price = price * 1.05
    RETURNING *
)
SELECT * FROM products;
```

###SELECT#UPDATE###############

```
WITH t AS (
    UPDATE products SET price = price * 1.05
    RETURNING *
)
SELECT * FROM t;
```

##########SELECT###############

################################## ### ############### ## ############### ############################## ##################################WITH######### ######## #####################

###WITH##ALSO####INSTEAD######### #####

#8#

PostgreSQL############################## CREATE TYPE#####PostgreSQL########### #######

8.1#################### ##############################PostgreSQL############### ##

#8.1

##	##	##
bigint	int8	8#########
bigserial	serial8	####8#####
bit [(*n*)]		#######
bit varying [(*n*)]	varbit [(*n*)]	#######
boolean	bool	#####/##
box		######
bytea		################byte array###
character [(*n*)]	char [(*n*)]	######
character varying [(*n*)]	varchar [(*n*)]	######
cidr		IPv4####IPv6##########
circle		#####
date		#########
double precision	float8	#########8####
inet		IPv4####IPv6#######
integer	int, int4	4#########
interval [*fields*] [(*p*)]		####
json		#####JSON###
jsonb		#### JSON ###, ###
line		########
lseg		######
macaddr		MAC#Media Access Control#####
macaddr8		MAC (Media Access Control) #### (EUI-64 ##)
money		####
numeric [(*p*, *s*)]	decimal [(*p*, *s*)]	#############
path		##########
pg_lsn		PostgreSQL#########
pg_snapshot		################ID########
point		#########
polygon		#############
real	float4	#########4####

##	##	##
smallint	int2	2########
smallserial	serial2	####2#####
serial	serial4	####4#####
text		######
time [(*p*)] [without time zone]		########
time [(*p*)] with time zone	timetz	#######
timestamp [(*p*)] [without time zone]		###########
timestamp [(*p*)] with time zone	timestamptz	#########
tsquery		##########
tsvector		########
txid_snapshot		##############ID########(### ##pg_snapshot###)
uuid		#######
xml		XML###

###

####################SQL######### bigint#bit#bit varying#boolean#char# character varying#character#varchar# date#double precision#integer# interval#numeric#decimal#real# smallint#time############# timestamp#############xml#

##PostgreSQL## ################## ##

8.1.

########2#4#8#######4#8######################### # 8.2##############

#8.2

##	#####	##	##
smallint	2###	######	-32768##+32767
integer	4###	#########	-2147483648## +2147483647
bigint	8###	#####	-9223372036854775808## +9223372036854775807
decimal	###	#########	#######131072######### ##16383###
numeric	###	#########	#######131072######### ##16383###
real	4###	########	6###

##	#####	##	##
double precision	8###	########	15###
smallserial	2###	#######	1##32767
serial	4###	######	1##2147483647
bigserial	8###	#########	1##9223372036854775807

################4.1.2######### ############################## ####9########### ################

8.1.1.

smallint#integer#bigint############################## ####################### #######

integer## smallint############# ################ bigint##integer##############################

SQL########integer####int##smallint#bigint########## int2#int4###int8## ###########SQL###################

8.1.2.

numeric##################### ##################################### numeric# ######################################10######################### ##### ##numeric##

################### numeric###*(precision)*############## ######################## ## numeric####*(scale)*###################### #####23.5141#########6#####4###### ### #####################

numeric############################## numeric###################

NUMERIC(*precision*, *scale*)

################################(#####)# #############

NUMERIC(*precision*)

#####0########### ###########

NUMERIC

####################numeric################# ###SQL###########0############ ########### ########################

> ##
>
> ####numeric################1000### #####numeric### 8.2############

####

NUMERIC(3, 1)

################1#####-99.9##99.9############

PostgreSQL 15#############numeric################ ################### ######## ################ ######

NUMERIC(2, -3)

######################-99000##99000############ ##############################
#########################0##

NUMERIC(3, 5)

################5#######-0.00999##0.00999############

> ##
>
> PostgreSQL###numeric#######-1000#1000################## ####SQL#######0################ ##

############0################ ## #####
##numeric#char(*n*)###varchar(*n*)####### #############10##4###############2###
##3##8##############

#########numeric####################

Infinity
-Infinity
NaN

####IEEE 754## SQL################
###UPDATE table SET x = '-Infinity'################### ################
############## ########inf#-inf##########

######################## ####Infinity############Infinity#Infinity######
#Infinity#########Infinity##Infinity##################NaN(######)###### ##
######numeric##

NaN(######)############################## ###NaN#########NaN###### ######
#NaN## ####NaN################
(##########NaN#0##1######)#

> ##
>
> ###################NaN##NaN######################### numeric#########
> ########################PostgreSQL#NaN###########NaN##################
> #####

decimal#numeric###### 2###########SQL######

#######numeric##0############################real#double precision#######
########### #########:

```
SELECT x,
  round(x::numeric) AS num_round,
```

```
  round(x::double precision) AS dbl_round
FROM generate_series(-3.5, 3.5, 1) as x;
  x   | num_round | dbl_round
------+-----------+-----------
 -3.5 |        -4 |        -4
 -2.5 |        -3 |        -2
 -1.5 |        -2 |        -2
 -0.5 |        -1 |        -0
  0.5 |         1 |         0
  1.5 |         2 |         2
  2.5 |         3 |         2
  3.5 |         4 |         4
(8 rows)
```

8.1.3.

real#double precision########10################################ ############ ## ##################IEEE##754######

##

- ##########################numeric#########
- ## ############################
- 2#####################################

#########################real####6########1E-37##1E+37######## double precision####15########1E-307##1E+308######## ######################## ####################### ##

######################10################## #####10################################## ################## (########2##) ####float8#######17##10###float4#######9##10##### ####

> ##
>
> ##

PostgreSQL##extra_float_digits############ 10################## ##0#####################6(float4###)#15(float8###)######### ###### ###########################-2##############4####13#########

0#####extra_float_digits#######################

> ##
>
> ########################extra_float_digits#3#################### ##

##############################

```
Infinity
-Infinity
NaN
```

#########IEEE 754############################## ######SQL##################
#UPDATE table SET x = '-Infinity'################## ####################
########## #########inf#-inf###########

> ##
>
> IEEE 754###NaN##NaN### ############
> ############################PostgreSQL#NaN###########NaN###############
> ########

###PostgreSQL###############SQL######float#float(*p*)########## ####*p*#2######
################ PostgreSQL#float(1)##float(24)#real############
##float(25)##float(53)#double precision################ ######*p*##########
#######float#double precision#########

8.1.4.

> ##
>
> #####[83.743842]##########PostgreSQL######################## SQL############
> ###CREATE TABLE########## ###########################PostgreSQL########
> ####### ##1######5.3########SQL##########################

smallserial#serial###bigserial##
################AUTO_INCREMENT############ ########

```
CREATE TABLE tablename (
    colname SERIAL
);
```

###############

```
CREATE SEQUENCE tablename_colname_seq AS integer;
CREATE TABLE tablename (
    colname integer NOT NULL DEFAULT nextval('tablename_colname_seq')
);
ALTER SEQUENCE tablename_colname_seq OWNED BY tablename.colname;
```

###NOT NULL###########
##NULL############## ############################UNIQUE#####PRIMARY KEY###
############################## ############################## ################
################

> ##
>
> smallserial# serial###bigserial##################################
> #"#"################## ################################"#######"########
> ### ###9.17#nextval()##########

serial#################serial######################### ####INSERT#######
############DEFAULT#####################

serial#serial4############ ###integer####### bigserial#serial8######
#bigint#################### ##############2^{31}###################bigserial##
smallserial#serial2##########smallint####################

serial## ##############################
###################

8.2.

money#################### # 8.3######### ############lc_monetary###############
#########2######### '$1,000.00'### #
############################

#8.3 ###

##	#####	##	##
money	8###	####	-92233720368547758.08 ## +92233720368547758.07

#######################lc_monetary############money#####################
################################lc_monetary##########################
########

numeric#int###bigint##money###############real##double precision####
#numeric ############################

```
SELECT '12.34'::float8::numeric::money;
```

###

money#############numeric############## ##################################
##

```
SELECT '52093.89'::money::numeric::float8;
```

money###################0############### ################################
######money####numeric##########money###### ############################
money#####money############double precision################## #########
#######

8.3.

#8.4 ###

##	##
character varying(*n*), varchar(*n*)	#######
character(*n*), char(*n*), bpchar(*n*)	###########
bpchar	variable unlimited length, blank-trimmed
text	#######

8.4#PostgreSQL##################

SQL#2################## character varying(*n*)#character(*n*)### ###*n*#####
########2###*n*#########################
##############SQL######
####character varying(*n*)#character(*n*)##############################
#########*n*########### ########SQL########## ##################character##
########character varying##################

####PostgreSQL###############text####### text####SQL###############SQL##
################ text#PostgreSQL##
#character varying####text######### #########character varying#text###
##################

##varchar#character varying##############bpchar#char#character#####
varchar#char#####SQL#########bpchar#PostgreSQL#####

#########*n*#################10,485,760############ #######character varying
####varchar############################## bpchar##########################
######################## character####char##############character(1)######

character########*n*################################ ##########################
##2##character################ ####################################
##SELECT 'a '::CHAR(2) collate "C" < E'a\n'::CHAR(2)#C################
############ character################################ character varying#
#text######################LIKE##################################

##############
######0####NUL#################### #######23.3########

######126################################## character################## ######
############################## ##
####### ### ################
#######1######## ###########*n*#################### ###########################
########################### ####################################text########
###character varying##########

###
##CPU######## ########3#################### ################character(*n*)######### ######PostgreSQL############## ####################character(*n*)#3#### ######## #########text#character varying##########

###############4.1.2.1#################9##########

#8.1

```
CREATE TABLE test1 (a character(4));
INSERT INTO test1 VALUES ('ok');
SELECT a, char_length(a) FROM test1; -- 1

  a   | char_length
------+-------------
 ok   |           2


CREATE TABLE test2 (b varchar(5));
INSERT INTO test2 VALUES ('ok');
INSERT INTO test2 VALUES ('good      ');
```

```
INSERT INTO test2 VALUES ('too long');
ERROR:  value too long for type character varying(5)

INSERT INTO test2 VALUES ('too long'::varchar(5)); -- ########
SELECT b, char_length(b) FROM test2;

   b   | char_length
-------+-------------
 ok    |           2
 good  |           5
 too l |           5
```

1 `char_length`###9.4#########

PostgreSQL#### 8.5#########2############## ################################### ############### `name`################## #####64####63######################### ##C########NAMEDATALEN################## ################################## ##### ############################### `"char"`###########`char(1)`######1######## ##################ASCII################ ################################

#8.5 #########

##	#####	##
`"char"`	1###	###########
`name`	64###	##############

8.4.

`bytea`################### # 8.6#########

#8.6 #########

##	#####	##
`bytea`	1###4##############	########

######################## ###################### 1############################ #########10#####32##126######################## ######################## ###### ## 2############################ ########################## ## ############

`bytea`######2############# #hex####PostgreSQL################ ################## ##### #####bytea_output##################hex### #hex###PostgreSQL 9.0############### ######## ###########################

##SQL##`BLOB`###`BINARY LARGE OBJECT`################### #####`bytea`########## ##################

8.4.1. `bytea`#hex##

#hex###################4######4######2##16######### ######################\x## ############ ##(##### 4.1.2.1)# ######## ## ####16################### ########## ############# ################\x################ hex########################## ################################### ############

#

```
SET bytea_output = 'hex';

SELECT '\xDEADBEEF'::bytea;
   bytea
------------
 \xdeadbeef
```

8.4.2. bytea########

#########bytea######PostgreSQL##### #########ASCII##########ASCII######### ################################## ### ### ########################## ####### ################################

#######bytea################################ ############################ ########################3##8##################### #############(10######### #92)######################### # 8.7### ##

#8.7 #############bytea####

10######	##	###########	#	####
0	######	'\000'	'\000'::bytea;	\x00
39	#####	''''### #'\047'	''''::bytea;	\x27
92	#######	'\\'####'\ \134'	'\\'::bytea;	\x5c
0##31#####127 ##255##	############	'*xxx*'(8##)	'\001'::bytea;	\x01

################################

8.7#########################SQL####################### ################### ################################## bytea################################## ######### ####bytea########################## 8.7###########################

###
4.1.2.1#######

Bytea############hex######### bytea_output#escape########################### ######3################ ##:

```
SET bytea_output = 'escape';

SELECT 'abc \153\154\155 \052\251\124'::bytea;
     bytea
----------------
 abc klm *\251T
```

10###92(########)################### #### 8.8##########

#8.8 bytea################

10######	##	###########	#	####
92	#######	\\	'\134'::bytea	\\
0##31###127## 255	############	*xxx*#8###	'\001'::bytea;	\001

10#######	##	###########	#	####
32##126	###########	###############	'\176'::bytea;	~

####PostgreSQL##############bytea#### ##### ##

8.5. ##/######

PostgreSQL#### 8.9#######SQL######################### ####################9.9### #### ######################B.6#########################

#8.9 ##/######

##	#####	##	#####	#####	##
timestamp [(*p*)] [without time zone]	8 ###	##############	4713 BC	294276 AD	1#####
timestamp [(*p*)] with time zone	8###	########### ##	4713 BC	294276 AD	1#####
date	4###	########	4713 BC	5874897 AD	1#
time [(*p*)] [without time zone]	8###	########	00:00:00	24:00:00	1#####
time [(*p*)] with time zone	12 ###	##############	00:00:00+1559	24:00:00-1559	1 #####
interval [*fields*] [(*p*)]	16###	####	-178000000#	178000000#	1#####

> **##**
>
> ##SQL#####timestamp#####timestamp without time zone############ ## PostgreSQL######## timestamp with time zone#timestamptz######## ########PostgreSQL######

time#timestamp###interval#############################*p*############## #### ######################### *p*######0##6###

interval###

YEAR
MONTH
DAY

```
HOUR
MINUTE
SECOND
YEAR TO MONTH
DAY TO HOUR
DAY TO MINUTE
DAY TO SECOND
HOUR TO MINUTE
HOUR TO SECOND
MINUTE TO SECOND
```

fields###*p*#######################*fields*#SECOND#####################

time with time zone###SQL### ### #####date#time#timestamp without time zone#timestamp with time zone# ##########################/##################

8.5.1. ##/#####

##########ISO 8601#SQL######POSTGRES############################ ########## #######-#-### DateStyle######MDY######## ###########DMY##############YMD####################

PostgreSQL###/#############SQL########## ##/################################ ####################B##########

###4.1.2.7########## SQL####

type [(*p*)] '*value*'

####*p*################################ ###time#timestamp###interval#####0## 6########### ################ ######################################(######### #########)#

8.5.1.1.

8.10#date#################

#8.10 ####

#	##
1999-01-08	ISO 8601########1#8############
January 8, 1999	####datestyle##################
1/8/1999	MDY#####1#8##DMY#####8#1##
1/18/1999	MDY#####1#18################
01/02/03	MDY#####2003#1#2##DMY#####2003#2#1##YMD#####2001#2#3##
1999-Jan-08	#######1#8#######
Jan-08-1999	#######1#8#######
08-Jan-1999	#######1#8#######
99-Jan-08	YMD####1#8############
08-Jan-99	1#8#####YMD########
Jan-08-99	1#8#####YMD########
19990108	ISO 8601########1999#1#8#######

#	##
990108	ISO 8601########1999#1#8#######
1999.008	#######
J2451187	#####
January 8, 99 BC	#####99#

8.5.1.2.

`###########time [(p) ] without time zone#time [ (p) ] with time zone#` ## `time####time without time zone#####`

#################################### ## 8.11## 8.12########## `time without time zone`############################## ##################America/New_York##### ################################## ### ######## ###########`time with time zone`################################## #############

#8.11

#	##
`04:05:06.789`	ISO 8601
`04:05:06`	ISO 8601
`04:05`	ISO 8601
`040506`	ISO 8601
`04:05 AM`	04:05####AM##########
`04:05 PM`	16:05########12#############
`04:05:06.789-8`	ISO 8601, UTC ##############
`04:05:06-08:00`	ISO 8601, UTC ##############
`04:05-08:00`	ISO 8601, UTC ##############
`040506-08`	ISO 8601, UTC ##############
`040506+0730`	ISO 8601, UTC ################ #####
`040506+07:30:00`	#######UTC#####(ISO 8601#### ###)
`04:05:06 PST`	############
`2003-04-12 04:05:06 America/New_York`	###########

#8.12

#	##
`PST`	##############
`America/New_York`	#######
`PST8PDT`	POSIX#######
`-8:00:00`	PST#UTC#####
`-8:00`	PST#UTC#####(ISO 8601#########)
`-800`	PST#UTC#####(ISO 8601#########)
`-8`	PST#UTC#####(ISO 8601#########)
`zulu`	UTC######

#	##
z	zulu###(###ISO 8601)

###############8.5.3#########

8.5.1.3.

##AD####BC####### ###AD/BC## ################################## ######

```
1999-01-08 04:05:06
```

#

```
1999-01-08 04:05:06 -8:00
```

#######ISO 8601######## ############

```
January 8 04:05:06 1999 PST
```

##############

##SQL###timestamp without time zone######timestamp with time zone#### #########+######-################### ############

```
TIMESTAMP '2004-10-19 10:23:54'
```

#timestamp without time zone##

```
TIMESTAMP '2004-10-19 10:23:54+02'
```

#timestamp with time zone###### PostgreSQL############################ #### #######timestamp without time zone####### #######timestamp with time zone##################

```
TIMESTAMP WITH TIME ZONE '2004-10-19 10:23:54+02'
```

#################### timestamp without time zone############PostgreSQL### ################ ##################/################################

timestamp with time zone################UTC####################GMT#### ####### ##################################UTC####### ################### #####TimeZone#####################timezone##############UTC#######

timestamp with time zone#############UTC#####timezone################## ######## ################timezone##########AT TIME ZONE###9.9.4 ##########

timestamp without time zone#timestamp with time zone#########timestamp without time zone###timezone################################ AT TIME ZONE######################

8.5.1.4.

PostgreSQL########## 8.13##############/############### infinity#-infinity# ############################## ################################/########## ## #now################################ #########SQL########################## #######

#8.13 #####/####

#####	####	##
epoch	date, timestamp	1970-01-01 00:00:00+00#Unix #############
infinity	date, timestamp, interval	################
-infinity	date, timestamp, interval	################
now	date, time, timestamp	##############
today	date, timestamp	#######
tomorrow	date, timestamp	#######
yesterday	date, timestamp	#######
allballs	time	00:00:00.00 UTC

SQL###### ##CURRENT_DATE#CURRENT_TIME#CURRENT_TIMESTAMP#LOCALTIME#LOCALTIMESTAMP## ################################## #9.9.5 ########## ####SQL################## ################

> **##**
>
> #########now#today#tomorrow##yesterday#########SQL############### ## ######### ##################################### ##############SQL############ ## ##'tomorrow'::date###CURRENT_DATE + 1#######

8.5.2. ##/#####

##/##########ISO 8601#SQL#Ingres######POSTGRES#Unix date######German########### ##### ######ISO##### ###SQL##ISO 8601############## #SQL################### # 8.14############## date#time##############################POSTGRES##ISO### #############(YMD#MDY###12-17-1997####DMY###17-12-1997######)

#8.14 ##/#######

####	##	#
ISO	ISO 8601, ##SQL	1997-12-17 07:37:16-08
SQL	######	12/17/1997 07:37:16.00 PST
Postgres	#####	Wed Dec 17 07:37:16 1997 PST
German	######	17.12.1997 07:37:16.00 PST

> **##**
>
> ISO 8601####################T####### PostgreSQL########################## #T############ ##############RFC3339[1]#########################

[1] https://datatracker.ietf.org/doc/html/rfc3339

SQL#POSTGRES###DMY######################## ################# ############ ##############8.5.1########### # 8.15########

#8.15

datestyle###	#####	###
SQL, DMY	*day*###/*month*###/*year*###	17/12/1997 15:37:16.00 CET
SQL, MDY	*month*###/*day*###/*year*###	12/17/1997 07:37:16.00 PST
Postgres, DMY	*day*###/*month*###/*year*###	Wed 17 Dec 07:37:16 1997 PST

ISO###########UTC### ################*hh*(#####)#########*hh*:*mm*##########*hh*:*mm*:*ss*## ## ################################ ############## ########ISO 8601########(*hh*#*hhmm*)#####################

####SET DATESTYLE#####postgresql.conf#######DateStyle################## #PGDATESTYLE###########/################

##/#########################to_char#9.8################

8.5.3.

###########1900#################### ################### PostgreSQL##########################IANA############### #

PostgreSQL##############SQL#################### ######SQL################### ########### 2##################

- date####################time######## ##################################### ######################################

- ##########UTC######################## #####DST#################/############ ##############

###/################### time with time zone########### #####PostgreSQL#################SQL#########time with time zone############### PostgreSQL##

################UTC############ #################TimeZone################### #############

PostgreSQL###3####################

- America/New_York########### ###########pg_timezone_names########### #52.32########### PostgreSQL#################IANA############ ############### #############

- PST######### ############UTC############## ############################ ########pg_timezone_abbrevs############52.31########### TimeZone######## #log_timezone#####################################AT TIME ZONE############ #######

- #############PostgreSQL##B.5########POSIX############### ################### ########################IANA################################

######################### ####UTC###
##2##UTC############### ###2014-06-04 12:00 America/New_York#########
#######################(UTC-4)### ###2014-06-04 12:00 EDT############ ##
##2014-06-04 12:00 EST################################(UTC-5)#########

#####################################UTC############### ########MSK######
UTC+3############UTC+4####### PostgreSQL#####################################
#################### ####EST#####################################

#################################### ####PostgreSQL#8.2################ #######
################################

################################## ############.../share/timezone/###.../
share/timezonesets/######################B.4#########

TimeZone#postgresql.conf######19########################## ################
####

- SQL####SET TIME ZONE################ ###SET TIMEZONE TO#######SQL#####
  ##############

- PGTZ######libpq##############SET TIME ZONE##################

8.5.4.

interval####################

[@] *quantity unit* [*quantity unit...*] [*direction*]

####quantity##########unit####
#microsecond#millisecond#second####minute####hour####day####week####month####year###
#####century#100#####millennium#1000######################### *direction*##
###ago######## #######@######################### ##############################
ago################### ########IntervalStyle#postgres_verbose##############
###########

################################### ####'1 12:59:10'#'1 day 12 hours
59 min 10 sec'#1##12##59#10######### ############################## ##
##'200-10'#'200 years 10 months'#200##10########## #########SQL########
######### ###IntervalStyle#sql_standard##########################

###4.4.3.2##############4.4.3.3#########################ISO 8601################ ##
#################

P *quantity unit* [*quantity unit* ...] [T [*quantity unit* ...]]

####P######################T########### ############# 8.16###### ###############
#############1########T############## ##M####T#################

#8.16 ISO 8601##############

###	##
Y	#
M	##########
W	#
D	#

###	##
H	##
M	##########
S	#

#########

P [*years-months-days*] [T *hours*:*minutes*:*seconds*]

############P############## ####T##################### ##ISO 8601########### ####

fields#######################*fields*#####################################
#*fields*###### ###INTERVAL '1' YEAR#1#########INTERVAL '1'#1########
####*fields*## ####INTERVAL '1 day 2:03:04' HOUR TO MINUTE#######################################

##SQL################################### ############################## ###### #####'-1 2:03:04'########################## PostgreSQL################## ###### ## ########################## ## IntervalStyle#sql_standard## ############ #########PostgreSQL########## ################################## ##########

######interval##3################## ########### ########## #############1##23 #####25############################## ## ###########################

```
SELECT '2 years 15 months 100 weeks 99 hours 123456789
 milliseconds'::interval;
               interval
--------------------------------------
 3 years 3 mons 700 days 133:17:36.789
```

########7###

#############################'1.5 weeks'#'01:02:03.45'##### ## ##interval## ################### ########'1.5 years'#'1 year 6 mons'###### #########1###30##1##24######### #####################'1.75 months'#1 mon 22 days 12:00:00###### ####### ###########

8.17####interval################

#8.17

#	##
1-2	##SQL###1#2##
3 4:05:06	##SQL###3#4##5#6#
1 year 2 months 3 days 4 hours 5 minutes 6 seconds	###Postgres###1#2#3#4##5#6#
P1Y2M3DT4H5M6S	ISO 8601 #################
P0001-02-03T04:05:06	ISO 8601 #############

8.5.5.

#######PostgreSQL#interval################### ######months######12############### days################ microseconds ############################ ############### #0~11#0~59#0~59## ################################justify_days###justify_hours##########

###########SET intervalstyle######## ##sql_standard#postgres#postgres_verbose###iso_8601################ ### ###postgres##### # 8.18#################

sql_standard## #######SQL############## ##################-###############-################################### ##################################

postgres#######DateStyle######ISO########PostgreSQL 8.4###############

postgres_verbose#######DateStyle######ISO###########PostgreSQL 8.4########## #####

iso_8601######ISO 8601 ###4.4.3.2######format with designators################

#8.18 ##########

####	#-#####	#-######	########
sql_standard	1-2	3 4:05:06	-1-2 +3 -4:05:06
postgres	1 year 2 mons	3 days 04:05:06	-1 year -2 mons +3 days -04:05:06
postgres_verbose	@ 1 year 2 mons	@ 3 days 4 hours 5 mins 6 secs	@ 1 year 2 mons -3 days 4 hours 5 mins 6 secs ago
iso_8601	P1Y2M	P3DT4H5M6S	P-1Y-2M3D T-4H-5M-6S

8.6.

PostgreSQL#####SQL#boolean########## # 8.19######### boolean################ ### ###############3####SQL##NULL################

#8.19 #######

##	#####	##
boolean	1###	########

#####SQL#########SQL######TRUE#FALSE###NULL##########

boolean##

true
yes
on
1

####################

false
no
off
0

t # n#################### ##################################

boolean############# 8.2########t#f#######

#8.2 boolean####

```
CREATE TABLE test1 (a boolean, b text);
INSERT INTO test1 VALUES (TRUE, 'sic est');
INSERT INTO test1 VALUES (FALSE, 'non est');
SELECT * FROM test1;
 a |    b
---+---------
 t | sic est
 f | non est

SELECT * FROM test1 WHERE a;
 a |    b
---+---------
 t | sic est
```

#######TRUE #FALSE#SQL##################(SQL##)##### #### 4.1.2.7############ ##########################'yes'::boolean##################

#########TRUE#FALSE#boolean########NULL###############boolean########### ####### ###############NULL::boolean######NULL#boolean################# # ##########boolean##

8.7.

###enum######################## ########################enum####### ####### ############################

8.7.1.

####CREATE TYPE############### #########

```
CREATE TYPE mood AS ENUM ('sad', 'ok', 'happy');
```

##

```
CREATE TYPE mood AS ENUM ('sad', 'ok', 'happy');
CREATE TABLE person (
    name text,
    current_mood mood
);
INSERT INTO person VALUES ('Moe', 'happy');
SELECT * FROM person WHERE current_mood = 'happy';
 name | current_mood
------+--------------
 Moe  | happy
(1 row)
```

8.7.2.

################################## ################################## ######## ##

```
INSERT INTO person VALUES ('Larry', 'sad');
INSERT INTO person VALUES ('Curly', 'ok');
SELECT * FROM person WHERE current_mood > 'sad';
 name  | current_mood
-------+--------------
 Moe   | happy
 Curly | ok
(2 rows)

SELECT * FROM person WHERE current_mood > 'sad' ORDER BY current_mood;
 name  | current_mood
-------+--------------
 Curly | ok
 Moe   | happy
(2 rows)

SELECT name
FROM person
WHERE current_mood = (SELECT MIN(current_mood) FROM person);
 name
-------
 Larry
(1 row)
```

8.7.3.

```
#################################### ##############


CREATE TYPE happiness AS ENUM ('happy', 'very happy', 'ecstatic');
CREATE TABLE holidays (
    num_weeks integer,
    happiness happiness
);
INSERT INTO holidays(num_weeks,happiness) VALUES (4, 'happy');
INSERT INTO holidays(num_weeks,happiness) VALUES (6, 'very happy');
INSERT INTO holidays(num_weeks,happiness) VALUES (8, 'ecstatic');
INSERT INTO holidays(num_weeks,happiness) VALUES (2, 'sad');
ERROR:  invalid input value for enum happiness: "sad"
SELECT person.name, holidays.num_weeks FROM person, holidays
  WHERE person.current_mood = holidays.happiness;
ERROR:  operator does not exist: mood = happiness

################################################################


SELECT person.name, holidays.num_weeks FROM person, holidays
  WHERE person.current_mood::text = holidays.happiness::text;
 name | num_weeks
------+-----------
 Moe  |         4
(1 row)
```

8.7.4.

```
######################## #####'happy'#'HAPPY'########## ################

################################################################(ALTER TYPE###)# #
#############################################################

#############4######### ##################PostgreSQL#######NAMEDATALEN#########
### ##############63##########
```

#######################pg_enum################ ###########################
##

8.8.

#######2################ # 8.20##PostgreSQL#####################

#8.20 ######

##	#####	##	##
point	16###	########	(x,y)
line	24###	#####	{A,B,C}
lseg	32 bytes	Finite line segment	[(x1,y1),(x2,y2)]
box	32 bytes	Rectangular box	(x1,y1),(x2,y2)
path	16+16n###	###########	((x1,y1),...)
path	16+16n###	###	[(x1,y1),...]
polygon	40+16n###	###########	((x1,y1),...)
circle	24###	#	<(x,y),r>#######

#############double precision(float8)#############

#########9.11#########

8.8.1.

################2######### point####################

```
( x , y )
  x , y
```

###x#y##########################

####1############

8.8.2.

########Ax + By + C = 0###########A#B####0############ line#################

```
{ A, B, C }
```

##############################

```
[ ( x1 , y1 ) , ( x2 , y2 ) ]
( ( x1 , y1 ) , ( x2 , y2 ) )
  ( x1 , y1 ) , ( x2 , y2 )
    x1 , y1   ,   x2 , y2
```

###(x1,y1)#(x2,y2)#######2########

8.8.3.

########2################ lseg####################

```
[ ( x1 , y1 ) , ( x2 , y2 ) ]
( ( x1 , y1 ) , ( x2 , y2 ) )
  ( x1 , y1 ) , ( x2 , y2 )
    x1 , y1   ,   x2 , y2
```

####(x1,y1)#(x2,y2)#########

###1############

8.8.4.

############################box####################

```
( ( x1 , y1 ) , ( x2 , y2 ) )
  ( x1 , y1 ) , ( x2 , y2 )
    x1 , y1   ,   x2 , y2
```

###(x1,y1)#(x2,y2)############

###2############

######################### ##

8.8.5.

######################## ##

path####################

```
[ ( x1 , y1 ) , ... , ( xn , yn ) ]
( ( x1 , y1 ) , ... , ( xn , yn ) )
  ( x1 , y1 ) , ... , ( xn , yn )
  ( x1 , y1   , ... ,   xn , yn )
    x1 , y1   , ... ,   xn , yn
```

########################## ####[]##########()########### 3####4###

#########################

8.8.6. ###(####)

#####[90.638298]############################ ##

##

polygon####################

```
( ( x1 , y1 ) , ... , ( xn , yn ) )
  ( x1 , y1 ) , ... , ( xn , yn )
  ( x1 , y1   , ... ,   xn , yn )
    x1 , y1   , ... ,   xn , yn
```

#########################

################

8.8.7.

################# circle###################

```
< ( x , y ) , r >
( ( x , y ) , r )
  ( x , y ) , r
    x , y   , r
```

###(x,y)#######r#######

##############

8.9.

PostgreSQL### 8.21#######IPv4#####IPv6#####MAC################### ############## ##################################### ##################################### ### #9.12###########

#8.21

##	#####	##
cidr	7####19###	IPv4####IPv6######
inet	7####19###	IPv4####IPv6#############
macaddr	6###	MAC####
macaddr8	8 ###	MAC #### (EUI-64 ##)

inet####cidr#######IPv4#######IPv6############# ::10.2.3.4#::ffff:10.4.3.2##IPv6##### ###################IPv4##########

8.9.1. inet

inet##IPv4####IPv6#################################### ######################## ######################### #######32######IPv4############################ IPv6## ######128#######128#################### ################inet####cidr####### ###

##############address/y### ####address#IPv4###IPv6######y############## /y## ################IPv4##32#IPv6##128######################## ##################### ##/y###########

8.9.2. cidr

cidr#####IPv4#IPv6############## ######CIDR##################### ############ ###address/y##address#IPv4####IPv6#######################y############## y## ## ## ##

8.22############

#8.22 cidr#######

cidr##	cidr##	abbrev(cidr)
192.168.100.128/25	192.168.100.128/25	192.168.100.128/25

cidr##	cidr##	abbrev(cidr)
192.168/24	192.168.0.0/24	192.168.0/24
192.168/25	192.168.0.0/25	192.168.0.0/25
192.168.1	192.168.1.0/24	192.168.1/24
192.168	192.168.0.0/24	192.168.0/24
128.1	128.1.0.0/16	128.1/16
128	128.0.0.0/16	128.0/16
128.1.2	128.1.2.0/24	128.1.2/24
10.1.2	10.1.2.0/24	10.1.2/24
10.1	10.1.0.0/16	10.1/16
10	10.0.0.0/8	10/8
10.1.2.3/32	10.1.2.3/32	10.1.2.3/32
2001:4f8:3:ba::/64	2001:4f8:3:ba::/64	2001:4f8:3:ba/64
2001:4f8:3:ba: 2e0:81ff:fe22:d1f1/128	2001:4f8:3:ba: 2e0:81ff:fe22:d1f1/128	2001:4f8:3:ba: 2e0:81ff:fe22:d1f1/128
::ffff:1.2.3.0/120	::ffff:1.2.3.0/120	::ffff:1.2.3/120
::ffff:1.2.3.0/128	::ffff:1.2.3.0/128	::ffff:1.2.3.0/128

8.9.3. inet#cidr#######

inet#####cidr#############inet###########0###############cidr############ ####192.168.0.1/24 #inet#######cidr###########

> ###
>
> ##inet####cidr####################host#text###abbrev########

8.9.4. macaddr

macaddr##################################MAC##########MAC############### ### ###############

```
'08:00:2b:01:02:03'
'08-00-2b-01-02-03'
'08002b:010203'
'08002b-010203'
'0800.2b01.0203'
'0800-2b01-0203'
'08002b010203'
```

###################### a##f#################### ####################

IEEE##802-2001###2###############MAC################## ######################### ####MSB####################08-00-2b-01-02-03#10:00:D4:80:40:C0### ################ ############################## PostgreSQL#################### ################## ##LSB########

##5##################

8.9.5. macaddr8

macaddr8#############################EUI-64############MAC############## ###### ########################MAC############################## 6########## MAC#####8##########################FF#FE######### IPv6#EUI-48#############1# ############EUI-64################# macaddr8_set7bit############### #####(## ######)16############':'#'-' #### '.'########################### 16######16#(8 # ##)#12#(6###)############### ############ ##################

```
'08:00:2b:01:02:03:04:05'
'08-00-2b-01-02-03-04-05'
'08002b:0102030405'
'08002b-0102030405'
'0800.2b01.0203.0405'
'0800-2b01-0203-0405'
'08002b01:02030405'
'08002b0102030405'
```

################## #######A ##F#####a ##f######## #############

######6##############

###EUI-48###48####MAC######IPv6##############EUI-64################## #macaddr8_set7bit######

```
SELECT macaddr8_set7bit('08:00:2b:01:02:03');

    macaddr8_set7bit
-------------------------
 0a:00:2b:ff:fe:01:02:03
(1 row)
```

8.10.

######1#0############# ########################## SQL#######2###### bit(*n*)#bit varying(*n*)### ###*n*#######

bit######*n*######################## ################################ bit varying########*n*######## ################### #######bit#####bit(1)######### ######bit varying##############

> ##
>
> ##########bit(*n*)###########*n*## ## ############bit varying(*n*)##############*n*##################

####################4.1.2.5######### ##########################9.6#########

#8.3

```
CREATE TABLE test (a BIT(3), b BIT VARYING(5));
INSERT INTO test VALUES (B'101', B'00');
INSERT INTO test VALUES (B'10', B'101');

ERROR:  bit string length 2 does not match type bit(3)
```

```
INSERT INTO test VALUES (B'10'::bit(3), B'101');
SELECT * FROM test;

  a  |  b
-----+-----
 101 | 00
 100 | 101
```

#######8#####1################5###8################# ############8.3############# ####################

8.11.

PostgreSQL###2##### ####### tsvector######################### tsquery################ #12###### ########## ###9.13###################

8.11.1. tsvector

tsvector###################### ##################################12###### #### ############################

```
SELECT 'a fat cat sat on a mat and ate a fat rat'::tsvector;
                      tsvector
--------------------------------------------------
 'a' 'and' 'ate' 'cat' 'fat' 'mat' 'on' 'rat' 'sat'
```

####################################

```
SELECT $$the lexeme '    ' contains spaces$$::tsvector;
                 tsvector
-------------------------------------------
 '    ' 'contains' 'lexeme' 'spaces' 'the'
```

################################

```
SELECT $$the lexeme 'Joe''s' contains a quote$$::tsvector;
                    tsvector
-----------------------------------------------
 'Joe''s' 'a' 'contains' 'lexeme' 'quote' 'the'
```

############################

```
SELECT 'a:1 fat:2 cat:3 sat:4 on:5 a:6 mat:7 and:8 ate:9 a:10 fat:11
 rat:12'::tsvector;
                                  tsvector
-------------------------------------------------------------------
------------
 'a':1,6,10 'and':8 'ate':9 'cat':3 'fat':2,11 'mat':7 'on':5 'rat':12
 'sat':4
```

#################### ################### #####1##16383################ 16383####### ######################

################################ ####A#B#C#D########## D################################

```
SELECT 'a:1A fat:2B,4C cat:5D'::tsvector;
          tsvector
----------------------------
 'a':1A 'cat':5 'fat':2B,4C
```

####################
##################

tsvector############################ ######################################
#########

```
SELECT 'The Fat Rats'::tsvector;
      tsvector
--------------------
 'Fat' 'Rats' 'The'
```

##tsvector######## ##############
##############to_tsvector########

```
SELECT to_tsvector('english', 'The Fat Rats');
   to_tsvector
-----------------
 'fat':2 'rat':3
```

###########12##########

8.11.2. tsquery

tsquery################### #########& (###)#| (###)#!(##)#########<->(FOLLOWED BY)############## FOLLOWED BY#####<*N*>#########*N*############## <->#<1>######

############################ ########! (NOT)#########<-> (FOLLOWED BY)###### ##### ####& (AND)####### | (OR)############

#########

```
SELECT 'fat & rat'::tsquery;
    tsquery
---------------
 'fat' & 'rat'

SELECT 'fat & (rat | cat)'::tsquery;
          tsquery
---------------------------
 'fat' & ( 'rat' | 'cat' )

SELECT 'fat & rat & ! cat'::tsquery;
        tsquery
------------------------
 'fat' & 'rat' & !'cat'
```

############tsquery######1###################### ##############tsvector### ####################

```
SELECT 'fat:ab & cat'::tsquery;
    tsquery
------------------
```

```
 'fat':AB & 'cat'
```

####tsquery#################*###############

```
SELECT 'super:*'::tsquery;
  tsquery
-----------
 'super':*
```

#########super#####tsvector#############

###############tsvector############ ###tsvector############tsquery#######
########## ###########to_tsquery#######

```
SELECT to_tsquery('Fat:ab & Cats');
    to_tsquery
------------------
 'fat':AB & 'cat'
```

to_tsquery########################### ########true######

```
SELECT to_tsvector( 'postgraduate' ) @@ to_tsquery( 'postgres:*' );
 ?column?
----------
 t
```

###postgres##postgr#############

```
SELECT to_tsvector( 'postgraduate' ), to_tsquery( 'postgres:*' );
  to_tsvector  | to_tsquery
---------------+------------
 'postgradu':1 | 'postgr':*
```

###postgraduate###########

8.12. UUID#

uuid######RFC 4122[2]:#ISO/IEC 9834-8:2005####################UUID######## #######
######################GUID######### ##
################################128######### ##################################
####################################

UUID######16################################## ######8#######1##4#######3####
##12#######1#######32##128######## #######UUID###########

```
a0eebc99-9c0b-4ef8-bb6d-6bb9bd380a11
```

###PostgreSQL###
################################ ##########

```
A0EEBC99-9C0B-4EF8-BB6D-6BB9BD380A11
{a0eebc99-9c0b-4ef8-bb6d-6bb9bd380a11}
a0eebc999c0b4ef8bb6d6bb9bd380a11
a0ee-bc99-9c0b-4ef8-bb6d-6bb9-bd38-0a11
{a0eebc99-9c0b4ef8-bb6d6bb9-bd380a11}
```

[2] https://datatracker.ietf.org/doc/html/rfc4122

##############

PostgreSQL#UUID#######9.14#########

8.13. XML#

xml#########XML############### text#######XML########################### ########### ######################## 9.15######### ######################## #configure --with-libxml###############

xml###XML##################XQuery#XPath##################[3]################# ############ ########################2################################ *xmlvalue* IS DOCUMENT############xml################################

xml################### D.3########

8.13.1. XML####

#######xml############xmlparse###########

```
XMLPARSE ( { DOCUMENT | CONTENT } value)
```

##

```
XMLPARSE (DOCUMENT '<?xml version="1.0"?><book><title>Manual</
title><chapter>...</chapter></book>')
XMLPARSE (CONTENT 'abc<foo>bar</foo><bar>foo</bar>')
```

##SQL#######XML###########################PostgreSQL#################

```
xml '<foo>bar</foo>'
'<foo>bar</foo>'::xml
```

xml########DTD####################DTD################ ###########XML Schema#####XML##########################

xml##################xmlserialize###########

```
XMLSERIALIZE ( { DOCUMENT | CONTENT } value AS type [ [ NO ] INDENT ] )
```

####*type*##character#character varying#text################### ######## SQL####xml########################## PostgreSQL##################

INDENT######################NO INDENT(#######)################### ########### ###################

XMLPARSE#XMLSERIALIZE#########xml############DOCUMENT#CONTENT###### ##XML option######################## #############

```
SET XML OPTION { DOCUMENT | CONTENT };
```

######PostgreSQL#####

```
SET xmloption TO { DOCUMENT | CONTENT };
```

[3] https://www.w3.org/TR/2010/REC-xpath-datamodel-20101214/#DocumentNode

################ ######CONTENT###########XML#############

8.13.2.

##############################XML############################## ###########
##PostgreSQL####################
##################################### 23.3######### ##########XML#######
###########encoding##############/##################################
###XML#####encoding#################### ###########xml#####################
encoding## #################XML#######
################################ ##
###################### ####xml####encoding######## ###########################
###################

#########
#####XML#####encoding#####################UTF-8########## #XML#########
PostgreSQL#UTF-16##################### ########################encoding######
##################UTF-8####encoding#########

###########PostgreSQL#####XML#####XML#################################
################# XML#######UTF-8#######################UTF-8#############
####

##
#########UTF-8##########XML########ASCII##################### #### #xmltable()#xpath()################

8.13.3. XML#######

xml################################## ####XML##############################
######### #####xml######################## #######XML####ID##################
########### XML############################## ############XML#############
#######

xml## XML######################
##############################XPath############################ #############
################################

PostgreSQL##############XML#################### ####PostgreSQL##############
#########

8.14. JSON####

JSON#####JSON(JavaScript Object Notation)###############JSON####RFC 7159[4]#######
#########text################JSON################JSON##################
####### ############################JSON############### 9.16##########

PostgreSQL###JSON##########2####json#jsonb###### ###########################
#######PostgreSQL#8.14.7#######jsonpath##########

json##jsonb############## ################ ############### json################
############################## jsonb######################### ###############

[4] https://datatracker.ietf.org/doc/html/rfc7159

jsonb################################
#######

json###JSON################### ###JSON
##(############################) #####
####jsonb###
#######

###################JSON######jsonb#######################################
###########################

RFC 7159##JSON####UTF8################# #################UTF8#########JSON#
#JSON################ ##UTF8#########
##############################

RFC 7159 ###JSON####Unicode######### \u*XXXX* ################ json############
###############Unicode#########################(###\u####16###4#)############
####jsonb################ ##########################Unicode########### jsonb#
#\u0000#######(####PostgreSQL#text############) ###Unicode###########Unicode#
###################### ###Unicode##
###########

> ##
>
> 9.16#######JSON#########Unicode################## ##########jsonb##
> #json######################## json#######################################
> ##JSON Unicode##############(#####
> ######)########

###JSON#jsonb###########RFC 7159################ 8.23#########PostgreSQL#####
########## #####jsonb######json#########JSON#############################
############## ##jsonb###PostgreSQL#numeric##############json####### #######
#########RFC 7159######### ####### IEEE 754 ########JSON#number##############
##(RFC 7159################)# ######
####PostgreSQL###########JSON#######################################

##########JSON######################PostgreSQL########################

#8.23 JSON########PostgreSQL#####

JSON ######	PostgreSQL#	##
string	text	\u0000######## ##################### ##############################
number	numeric	NaN # infinity ########
boolean	boolean	####true # false ############
null	(none)	SQL#NULL#########

8.14.1. JSON######

JSON###########RFC 7159 #########

#########json#(### jsonb#)#####

```
-- ########/#######
-- ########################true# false####null###
SELECT '5'::json;


-- 0############################
SELECT '[1, 2, "foo", null]'::json;


-- ################
-- ############################################
SELECT '{"bar": "baz", "balance": 7.77, "active": false}'::json;


-- ###########################
SELECT '{"foo": [true, "bar"], "tags": {"a": 1, "b": null}}'::json;
```

########JSON################################json#################jsonb######
################### #######################

```
SELECT '{"bar": "baz", "balance": 7.77, "active":false}'::json;
                       json
-------------------------------------------------
 {"bar": "baz", "balance": 7.77, "active":false}
(1 row)

SELECT '{"bar": "baz", "balance": 7.77, "active":false}'::jsonb;
                       jsonb
-------------------------------------------------
 {"bar": "baz", "active": false, "balance": 7.77}
(1 row)
```

#############jsonb######numeric##################################E#########
########## #######

```
SELECT '{"reading": 1.230e-5}'::json, '{"reading": 1.230e-5}'::jsonb;
         json          |          jsonb
-----------------------+------------------------
 {"reading": 1.230e-5} | {"reading": 0.00001230}
(1 row)
```

###############jsonb###

JSON##################################9.16##########

8.14.2. JSON#########

JSON## #######
###################################### ############################JSON######
######################### ##
###

JSON###
################## ####################################JSON###################
######JSON##

8.14.3. jsonb##############

###########jsonb########## json################## jsonb######################
############ #################true######

```
-- ######/######################
SELECT '"foo"'::jsonb @> '"foo"'::jsonb;


-- ####################
SELECT '[1, 2, 3]'::jsonb @> '[1, 3]'::jsonb;


-- ################################
SELECT '[1, 2, 3]'::jsonb @> '[3, 1]'::jsonb;


-- ##########################
SELECT '[1, 2, 3]'::jsonb @> '[1, 2, 2]'::jsonb;


-- ####################################
SELECT '{"product": "PostgreSQL", "version": 9.4, "jsonb": true}'::jsonb
 @> '{"version": 9.4}'::jsonb;


-- ##################
-- ############################

SELECT '[1, 2, [1, 3]]'::jsonb @> '[1, 3]'::jsonb;  -- false###


-- ###########################
SELECT '[1, 2, [1, 3]]'::jsonb @> '[[1, 3]]'::jsonb;


-- ##############

SELECT '{"foo": {"bar": "baz"}}'::jsonb @> '{"bar": "baz"}'::jsonb;  --
 false###


-- ########################
SELECT '{"foo": {"bar": "baz"}}'::jsonb @> '{"foo": {}}'::jsonb;

##############################################################################
################ ##############################################################

#####################################################


-- #####################
SELECT '["foo", "bar"]'::jsonb @> '"bar"'::jsonb;


-- ################ -- ##########

SELECT '"bar"'::jsonb @> '["bar"]'::jsonb;  -- false###

jsonb############################(#####text#)##jsonb#######################
############# ################true######


-- ##############
SELECT '["foo", "bar", "baz"]'::jsonb ? 'bar';
```

```
-- ##################
SELECT '{"foo": "bar"}'::jsonb ? 'foo';


-- ################

SELECT '{"foo": "bar"}'::jsonb ? 'bar';  -- false###


-- ##################################

SELECT '{"foo": {"bar": "baz"}}'::jsonb ? 'bar'; -- false###


-- ####JSON#######################
SELECT '"foo"'::jsonb ? 'foo';
```

JSON#### ### ###

```
CREATE INDEX idxginp ON api USING GIN (jdoc jsonb_path_ops);
```

#######Web##########################JSON############################## #####
##############

```
{
    "guid": "9c36adc1-7fb5-4d5b-83b4-90356a46061a",
    "name": "Angela Barton",
    "is_active": true,
    "company": "Magnafone",
    "address": "178 Howard Place, Gulf, Washington, 702",
    "registered": "2009-11-07T08:53:22 +08:00",
    "latitude": 19.793713,
    "longitude": 86.513373,
    "tags": [
        "enim",
        "aliquip",
        "qui"
    ]
}
```

#####api#jsonb##jdoc############# ######GIN#################################
###########

```
-- "company"## # "Magnafone"###########
SELECT jdoc->'guid', jdoc->'name' FROM api WHERE jdoc @> '{"company":
 "Magnafone"}';
```

############################?###############jdoc######################
##

```
-- ## "tags" ###### "qui"############
SELECT jdoc->'guid', jdoc->'name' FROM api WHERE jdoc -> 'tags' ? 'qui';
```

###"tags"################
##########################

```
CREATE INDEX idxgintags ON api USING GIN ((jdoc -> 'tags'));
```

WHERE## jdoc -> 'tags' ? 'qui'########jdoc->'tags'###?#############
######## ################11.7###########

#################################

```
-- ## "tags"# ##"qui"#############
SELECT jdoc->'guid', jdoc->'name' FROM api WHERE jdoc @> '{"tags":
 ["qui"]}';
```

jdoc########GIN############################ #########tags################
###########################jdoc############################ ################
###
###########

GIN######jsonpath###########@?####@@########### #########

```
SELECT jdoc->'guid', jdoc->'name' FROM api WHERE jdoc @? '$.tags[*] ? (@
 == "qui")';
```

```
SELECT jdoc->'guid', jdoc->'name' FROM api WHERE jdoc @@ '$.tags[*] ==
 "qui"';
```

#####[93.108729]############GIN#######jsonpath#####*accessors_chain = constant*## #######(accessors chain) #.*key*#[*]#[*index*]######### jsonb_ops######.*#.**########### ##jsonb_path_ops###############

jsonb_path_ops#######@>#@?#@@##############################jsonb_ops##### ################# jsonb_path_ops###############jsonb_ops################ ## ##############################

jsonb_ops#jsonb_path_ops#GIN## ########################### [5] #####jsonb_path_ops######################## ##{"foo": {"bar": "baz"}}############foo#bar# baz############### ######## ## ###foo############################## # ##jsonb_ops##########foo#bar#baz###3################ ##################### ####3############# GIN#############################3######################## #####jsonb_path_ops###########

jsonb_path_ops#########{"a": {}}#############JSON################ ####### ## jsonb_path_ops###########################

jsonb###btree # hash ############## #######JSON############################## ## jsonb####btree##

Object > Array > Boolean > Number > String > Null

Object with n pairs > object with n - 1 pairs

Array with n elements > array with n - 1 elements

##########################

key-1, value-1, key-2 ...

################################### ## ############

```
{ "aa": 1, "c": 1} > {"b": 1, "d": 1}
```

#######################

element-1, element-2 ...

JSON###########PostgreSQL##

8.14.5. jsonb###

jsonb## #########jsonb_set####path## ############################ jsonb## ################## ##########jsonb#######

[5] ########## ################JSON################################

UPDATE####jsonb#####SET########## ################################### ##
####val['a']['b']['c']##val#val['a']#val['a']['b']###############c#####
######## val['a']###val['a']['b']################################### ####
###val#########1##########jsonb null#######################################
################

#########

```
-- ################
SELECT ('{"a": 1}'::jsonb)['a'];


-- #######################
SELECT ('{"a": {"b": {"c": 1}}}'::jsonb)['a']['b']['c'];


-- ################
SELECT ('[1, "2", null]'::jsonb)[1];


-- #################'1'#############
-- ######jsonb##########
UPDATE table_name SET jsonb_field['key'] = '1';


-- ###jsonb_field['a']['b']###############################
-- #####{"a": 1}###'a'#######
UPDATE table_name SET jsonb_field['a']['b']['c'] = '1';


-- WHERE#####################
-- ########jsonb#############jsonb##########
-- ########"value"####jsonb#######
SELECT * FROM table_name WHERE jsonb_field['key'] = '"value"';
```

#####jsonb#######jsonb_set############ ##jsonb##NULL####################
########JSON#(##########)##############

```
-- jsonb_field#NULL####{"a": 1}###
UPDATE table_name SET jsonb_field['a'] = '1';


-- jsonb_field#NULL####[1]###
UPDATE table_name SET jsonb_field[0] = '1';
```

###NULL##########

```
-- jsonb_field#[]###[null, null, 2]####
-- jsonb_field#[0]###[0, null, 2]###
UPDATE table_name SET jsonb_field[2] = '2';
```

##jsonb####################### (####
##############################) ##
##########null########

```
-- jsonb_field#{}#####{"a": [{"b": 1}]}###
```

```
UPDATE table_name SET jsonb_field['a'][0]['b'] = '1';


-- jsonb_field#[]#####[null, {"a": 1}]###
UPDATE table_name SET jsonb_field[1]['a'] = '1';
```

8.14.6.

###########jsonb#####################

PL/Perl######jsonb_plperl#jsonb_plperlu###### #######jsonb###Perl#######################

PL/Python######jsonb_plpython3u###### ########jsonb###Python#############################

########jsonb_plperl##trusted#########################CREATE################################ ##########################

8.14.7. jsonpath#

jsonpath###PostgreSQL#JSON##################SQL/JSON################## ######SQL/JSON##############SQL/JSON#########################JSON#############################

SQL/JSON####################SQL######## ####JSON######################SQL/JSON###########JavaScript#########

- ###(.)################
- ###([])##############
- 1#######SQL#######SQL/JSON###0######

SQL/JSON###########JavaScript#########JavaScript##############SQL#JSON######################### ####SQL/JSON####.1#1.######JSON###### ##1_000_000#0x1EEE_FFFF#0o273#0b100101###10#################################### SQL/JSON########JavaScript########SQL###

SQL/JSON######SQL######SQL###(4.1.2.1###)# ############################## ############### JavaScript/ECMAScript### ##########################\"##########################\\########### ################################JavaScript################## #####ASCII######\b#\f#\n#\r#\t#\v#2##16#################\x*NN*#4##16################Unicode####\u*NNNN*###1#6##16######## Unicode###########\u{*N...*}###

#############################

- JSON############### #############true#false###null###
- ##### 8.24#
- ######## 8.25#
- jsonpath########9.16.2.3#
- ####################################

jsonpath######SQL/JSON###########9.16.2##########

#8.24 `jsonpath`##

##	##
`$`	#######(*context item*)#JSON#######
`$varname`	####### ########JSON#########*vars*######## ## ## 9.49#########
`@`	###################

#8.25 `jsonpath` Accessors

#######	##
`.`*`key`* `."$`*`varname`*`"`	############################ ####$############### ########JavaScript################################## ######
`.*`	##
`.**`	#########JSON#################################### ################### ###SQL/JSON###PostgreSQL######

`.**{`*`level`*`}`
 	
`.**{`*`start_level`* `to` *`end_level`*`}`
	`.**`#######JSON################ ############### # ##0################ ###################`last`###### ###### ###SQL/JSON###PostgreSQL######

`[`*`subscript`*`, ...]`	######## *`subscript`*`##`*`index`*`###`*`start_index`*`#`
	`#`*`end_index`*`###2########### ######################## ###### ############`*`start_index`*`#`*`end_index`*`######## ##########################`
	
	`#####`*`index`*`#################################### ## ######0############## ###last################### ######### ######################`

`[*]`	##########################

8.15.

PostgreSQL########################## ## ########

8.15.1.

```
##############################


CREATE TABLE sal_emp (
    name            text,
    pay_by_quarter  integer[],
    schedule        text[][]
);

##################################[]########## #######text#####name##########
######integer#######pay_by_quarter####################text######
#schedule######sal_emp###############

CREATE TABLE###########################


CREATE TABLE tictactoe (
```

```
    squares   integer[3][3]
);
```

################################## ########################

#################### ## ####CREATE TABLE########################## ################

SQL#####ARRAY###########1################### pay_by_quarter############## ####

```
    pay_by_quarter  integer ARRAY[4],
```

###############################

```
    pay_by_quarter  integer ARRAY,
```

###########PostgreSQL####################

8.15.2.

#C#################### ################## ## ####### #######

```
'{ val1 delim val2 delim ... }'
```

###*delim*###pg_type################## PostgreSQL#########################;## ####box############,####### #####*val*################ ##############

```
'{{1,2,3},{4,5,6},{7,8,9}}'
```

#######3##############3×3######

#######NULL##############NULL######### #NULL#################### #NULL######################################

###########4.1.2.7######################## ################################## ## ################

###INSERT###########

```
INSERT INTO sal_emp
    VALUES ('Bill',
    '{10000, 10000, 10000, 10000}',
    '{{"meeting", "lunch"}, {"training", "presentation"}}');

INSERT INTO sal_emp
    VALUES ('Carol',
    '{20000, 25000, 25000, 25000}',
    '{{"breakfast", "consulting"}, {"meeting", "lunch"}}');
```

######2##################

```
SELECT * FROM sal_emp;
 name  |      pay_by_quarter       |                 schedule
-------+---------------------------
+-----------------------------------------
```

```
 Bill  | {10000,10000,10000,10000} | {{meeting,lunch},
{training,presentation}}
 Carol | {20000,25000,25000,25000} | {{breakfast,consulting},
{meeting,lunch}}
(2 rows)
```

########################### #####################

```
INSERT INTO sal_emp
    VALUES ('Bill',
    '{10000, 10000, 10000, 10000}',
    '{{"meeting", "lunch"}, {"meeting"}}');
ERROR:  malformed array literal: "{{"meeting", "lunch"}, {"meeting"}}"
DETAIL:  Multidimensional arrays must have sub-arrays with matching
 dimensions.
```

ARRAY##########

```
INSERT INTO sal_emp
    VALUES ('Bill',
    ARRAY[10000, 10000, 10000, 10000],
    ARRAY[['meeting', 'lunch'], ['training', 'presentation']]);

INSERT INTO sal_emp
    VALUES ('Carol',
    ARRAY[20000, 25000, 25000, 25000],
    ARRAY[['breakfast', 'consulting'], ['meeting', 'lunch']]);
```

########SQL####################### ### ARRAY######4.2.12##############

8.15.3.

############################## ######################### #########2############ ###########

```
SELECT name FROM sal_emp WHERE pay_by_quarter[1] <> pay_by_quarter[2];

 name
-------
 Carol
(1 row)
```

##################### ######PostgreSQL######1################## ######*n*##### #array[1]#####array[*n*]######

###############3############

```
SELECT pay_by_quarter[3] FROM sal_emp;

 pay_by_quarter
----------------
          10000
          25000
(2 rows)
```

################################ ########################*lower-bound*:*upper-bound*####### ##########Bill########2####################

```
SELECT schedule[1:2][1:1] FROM sal_emp WHERE name = 'Bill';

        schedule
------------------------
 {{meeting},{training}}
(1 row)
```

###1######
######## ####[2]######## [1:2]#######

```
SELECT schedule[1:2][2] FROM sal_emp WHERE name = 'Bill';

                 schedule
-------------------------------------------
 {{meeting,lunch},{training,presentation}}
(1 row)
```

####[2][1:1]#####[1:2][1:1]##
######

#######*lower-bound#upper-bound*##
####

```
SELECT schedule[:2][2:] FROM sal_emp WHERE name = 'Bill';

        schedule
------------------------
 {{lunch},{presentation}}
(1 row)

SELECT schedule[:][1:1] FROM sal_emp WHERE name = 'Bill';

        schedule
------------------------
 {{meeting},{training}}
(1 row)
```

#####NULL#########NULL###########NULL###### #################NULL######
################ ####schedule###[1:3][1:2]######schedule[3][3]####NULL#
######################################NULL#######

###############NULL#########NULL######NULL###### ##########################
###########NULL#######0########### ################################## ########
#############NULL######################

array_dims#####################

```
SELECT array_dims(schedule) FROM sal_emp WHERE name = 'Carol';

 array_dims
------------
 [1:2][1:2]
(1 row)
```

array_dims###text######### #################################### ##
#array_upper#array_lower##################################

```
SELECT array_upper(schedule, 1) FROM sal_emp WHERE name = 'Carol';

 array_upper
```

```
-------------
           2
(1 row)
```

array_length##################

```
SELECT array_length(schedule, 1) FROM sal_emp WHERE name = 'Carol';

 array_length
--------------
            2
(1 row)
```

cardinality################### ####unnest################

```
SELECT cardinality(schedule) FROM sal_emp WHERE name = 'Carol';

 cardinality
-------------
           4
(1 row)
```

8.15.4.

###################

```
UPDATE sal_emp SET pay_by_quarter = '{25000,25000,27000,27000}'
    WHERE name = 'Carol';
```

####ARRAY#################

```
UPDATE sal_emp SET pay_by_quarter = ARRAY[25000,25000,27000,27000]
    WHERE name = 'Carol';
```

###1################

```
UPDATE sal_emp SET pay_by_quarter[4] = 15000
    WHERE name = 'Bill';
```

###############

```
UPDATE sal_emp SET pay_by_quarter[1:2] = '{27000,27000}'
    WHERE name = 'Carol';
```

lower-bound#*upper-bound*###################NULL#0###################(### ########################)#

####################################### #######################NULL###### ## #######myarray#####4####myarray[6]#########6#########myarray[5]#NULL ###### ##############1############ #################

#######1################## ######-2##7#########array[-2:7]#######

############||###############

```
SELECT ARRAY[1,2] || ARRAY[3,4];
 ?column?
```

```
-----------
 {1,2,3,4}
(1 row)

SELECT ARRAY[5,6] || ARRAY[[1,2],[3,4]];
      ?column?
---------------------
 {{5,6},{1,2},{3,4}}
(1 row)
```

#######################1################ ####2##N-########N-#####N+1-#########
#####

1#####1### ##########

```
SELECT array_dims(1 || '[0:1]={2,3}'::int[]);
 array_dims
------------
 [0:2]
(1 row)

SELECT array_dims(ARRAY[1,2] || 3);
 array_dims
------------
 [1:3]
(1 row)
```

#########2### ############################
#######

```
SELECT array_dims(ARRAY[1,2] || ARRAY[3,4,5]);
 array_dims
------------
 [1:5]
(1 row)

SELECT array_dims(ARRAY[[1,2],[3,4]] || ARRAY[[5,6],[7,8],[9,0]]);
 array_dims
------------
 [1:5][1:2]
(1 row)
```

N-#####N+1-### #####N-##########N+1-#######
############ #######

```
SELECT array_dims(ARRAY[1,2] || ARRAY[[3,4],[5,6]]);
 array_dims
------------
 [1:3][1:2]
(1 row)
```

###array_prepend#array_append#####array_cat############### ###2##########
##########array_cat############ #######

```
SELECT array_prepend(1, ARRAY[2,3]);
 array_prepend
---------------
 {1,2,3}
(1 row)
```

```
SELECT array_append(ARRAY[1,2], 3);
 array_append
--------------
 {1,2,3}
(1 row)

SELECT array_cat(ARRAY[1,2], ARRAY[3,4]);
 array_cat
-----------
 {1,2,3,4}
(1 row)

SELECT array_cat(ARRAY[[1,2],[3,4]], ARRAY[5,6]);
      array_cat
---------------------
 {{1,2},{3,4},{5,6}}
(1 row)

SELECT array_cat(ARRAY[5,6], ARRAY[[1,2],[3,4]]);
      array_cat
---------------------
 {{5,6},{1,2},{3,4}}
```

#############3################### ##############1###########################

```
SELECT ARRAY[1, 2] || '{3, 4}';  -- ##################
 ?column?
-----------
 {1,2,3,4}


SELECT ARRAY[1, 2] || '7';                 -- #####
ERROR:  malformed array literal: "7"


SELECT ARRAY[1, 2] || NULL;                -- ########NULL###
 ?column?
----------
 {1,2}
(1 row)


SELECT array_append(ARRAY[1, 2], NULL);    -- ############
 array_append
--------------
 {1,2,NULL}
```

############(###########)########### ##########array_append#####array_cat## ###### ##array_append########

8.15.5.

################################## ############################## #######

```
SELECT * FROM sal_emp WHERE pay_by_quarter[1] = 10000 OR
                            pay_by_quarter[2] = 10000 OR
                            pay_by_quarter[3] = 10000 OR
                            pay_by_quarter[4] = 10000;
```

#########9.25########## ###

```
SELECT * FROM sal_emp WHERE 10000 = ANY (pay_by_quarter);
```

###########10000##################

```
SELECT * FROM sal_emp WHERE 10000 = ALL (pay_by_quarter);
```

#######generate_subscripts############ #########

```
SELECT * FROM
   (SELECT pay_by_quarter,
           generate_subscripts(pay_by_quarter, 1) AS s
      FROM sal_emp) AS foo
 WHERE pay_by_quarter[s] = 10000;
```

9.68##########

&&###################### ######################### ####

```
SELECT * FROM sal_emp WHERE pay_by_quarter && ARRAY[10000];
```

#################9.19############## 11.2##############################

##array_position#array_positions######################### ################################### ###########

```
SELECT array_position(ARRAY['sun','mon','tue','wed','thu','fri','sat'],
 'mon');
 array_position
----------------
              2
(1 row)

SELECT array_positions(ARRAY[1, 4, 3, 1, 3, 4, 2, 1], 1);
 array_positions
-----------------
 {1,4,8}
(1 row)
```

###
############ ## ################### ######################## #####################################

8.15.6.

####################I/O## ###########{#}# ############################## ###########,############### #######typdelim#### ########## PostgreSQL######################;####box################## ######### ##

##################################NULL##################################### ### ## ################################### ##

#######################1######### ##
######## ########################:############[]############### ######=######
########### #######

```
SELECT f1[1][-2][3] AS e1, f1[1][-1][5] AS e2
 FROM (SELECT '[1:1][-2:-1][3:5]={{{1,2,3},{4,5,6}}}'::int[] AS f1) AS ss;

 e1 | e2
----+----
  1 |  6
(1 row)
```

1#######################################

#########NULL###############NULL######## ##########################NULL##
#################### ###8.2###PostgreSQL##########array_nulls#######off##
##NULL#NULL####################

####################################
##
#####NULL###########################
################# ##
########

################################### ################################ ###########
########### ##

SQL#########################ARRAY######4.2.12#######################
ARRAY################################

8.16.

####################### ############################## PostgreSQL##############
####################### ################################

8.16.1.

###########2######

```
CREATE TYPE complex AS (
    r       double precision,
    i       double precision
);

CREATE TYPE inventory_item AS (
    name            text,
    supplier_id     integer,
    price           numeric
);
```

################################CREATE TABLE###### #######NOT NULL##########
AS###################### ###########CREATE TYPE#######################
######

#####################################

```
CREATE TABLE on_hand (
    item      inventory_item,
    count     integer
);

INSERT INTO on_hand VALUES (ROW('fuzzy dice', 42, 1.99), 1000);
```

#######################

```
CREATE FUNCTION price_extension(inventory_item, integer) RETURNS numeric
AS 'SELECT $1.price * $2' LANGUAGE SQL;

SELECT price_extension(item, 10) FROM on_hand;
```

##########

```
CREATE TABLE inventory_item (
    name            text,
    supplier_id     integer REFERENCES suppliers,
    price           numeric CHECK (price > 0)
);
```

##################inventory_item################################### #######
############################## ##
########## ################################CHECK#################

8.16.2.

################################
#################################### ################## ##############################

'(*val1* , *val2* , ...)'

##########

'("fuzzy dice",42,1.99)'

#######inventory_item############# ######NULL#################################### ########
##3########NULL########

'("fuzzy dice",42,)'

NULL#####################################

'("",42,)'

################NULL#######3#########NULL######

#############4.1.2.7########################### ################################## #
#####################################

###ROW########################### ##
################# ####################

```
ROW('fuzzy dice', 42, 1.99)
ROW('', 42, NULL)
```

####2#################ROW#################### #######################

```
('fuzzy dice', 42, 1.99)
('', 42, NULL)
```

ROW#######4.2.13###########

8.16.3.

####on_hand##################

```
SELECT item.name FROM on_hand WHERE item.price > 9.99;
```

####SQL#########item#on_hand############################## ##################

```
SELECT (item).name FROM on_hand WHERE (item).price > 9.99;
```

##

```
SELECT (on_hand.item).name FROM on_hand WHERE (on_hand.item).price > 9.99;
```

#####################item#####################################

#################################### ##################1#################### ############

```
SELECT (my_func(...)).field FROM ...
```

#########################

8.16.5############*##############################

8.16.4.

################################## #################

```
INSERT INTO mytab (complex_col) VALUES((1.1,2.2));
```

```
UPDATE mytab SET complex_col = ROW(1.1,2.2) WHERE ...;
```

######ROW#####2######ROW######### #################

#####################################

```
UPDATE mytab SET complex_col.r = (complex_col).r + 1 WHERE ...;
```

####SET###
####

###INSERT#########################

```
INSERT INTO mytab (complex_col.r, complex_col.i) VALUES(1.1, 2.2);
```

################################NULL#######

8.16.5.

################################ ##
###########

PostgreSQL## ########
#inventory_item############################

```
SELECT c FROM inventory_item c;
```

#####################################

```
           c
------------------------
 ("fuzzy dice",42,1.99)
(1 row)
```

##c##########################
#####

###table_name.column_name##
############################

```
SELECT c.* FROM inventory_item c;
```

###SQL#######SQL##

```
    name    | supplier_id | price
------------+-------------+-------
 fuzzy dice |          42 |  1.99
(1 row)
```

##################################

```
SELECT c.name, c.supplier_id, c.price FROM inventory_item c;
```

PostgreSQL############################ ##############.*#######################
############# ####myfunc()##a#b#c#####################################

```
SELECT (myfunc(x)).* FROM some_table;
SELECT (myfunc(x)).a, (myfunc(x)).b, (myfunc(x)).c FROM some_table;
```

> **###**
>
> PostgreSQL####################################### ##################
> #myfunc()################## ##
> ##
>
> ```
> SELECT m.* FROM some_table, LATERAL myfunc(x) AS m;
> ```
>
> LATERAL FROM##############1####2############m.*###m.a, m.b, m.c#####
> ########FROM#############(LATERAL###################some_table##x#
> ######################)

#####[93.203883]#*composite_value*.*#######SELECT#####
##INSERT/UPDATE/DELETE#RETURNING####VALUES#############################
####### ##.*############### ###########
################################ ####somefunc()############################

```
SELECT somefunc(c.*) FROM inventory_item c;
SELECT somefunc(c) FROM inventory_item c;
```

#######inventory_item######################### #######.*################
#######c.*#c######################
#######.*####c#########################c############################

```
########################################
```

```
SELECT * FROM inventory_item c ORDER BY c;
SELECT * FROM inventory_item c ORDER BY c.*;
SELECT * FROM inventory_item c ORDER BY ROW(c.*);
```

####ORDER BY###################9.25.6########################## ##
##inventory_item#c### ####
#####################################

```
SELECT * FROM inventory_item c ORDER BY ROW(c.name, c.supplier_id,
 c.price);
SELECT * FROM inventory_item c ORDER BY (c.name, c.supplier_id, c.price);
```

###########ROW######################

###########
##*field*(*table*)######*table.field*############### #################

```
SELECT c.name FROM inventory_item c WHERE c.price > 1000;
SELECT name(c) FROM inventory_item c WHERE price(c) > 1000;
```

################

```
SELECT somefunc(c) FROM inventory_item c;
SELECT somefunc(c.*) FROM inventory_item c;
SELECT c.somefunc FROM inventory_item c;
```

###############
########somefunc##############################

> ###
>
> ## ######
> ## #######11####
> PostgreSQL## ######
> ##############################*schema.func*(*compositevalue*)########

8.16.6.

#######################I/O##################################### ##############
#(###)##############,######## ### ######
####

```
'(  42)'
```

##

################################
##################
##################################
######################## ####SQL####################### ####################
##

################################NULL###### NULL################## "" #######
##

###
##################################

> ##
>
> SQL## ########
> ## ##########################
> #text############################
>
> ```
> INSERT ... VALUES ('("\"\\")');
> ```
>
> ##############1################################### ("\"\\") ######## ##
> ##text################"\##### #######bytea##########################
> ###########1################################8################### ######
> ##4.1.2.4################################

> ###
>
> SQL##################ROW########################## ROW################
> ##############################

8.17.

##########(##############)############### ####timestamp#####################
############### #########tsrange(#timestamp range####)##timestamp########## ##
###

##################
###

######################### ##################NULL############### ##############
########################

8.17.1.

PostgreSQL##################

- int4range—integer####int4multirange—#######
- int8range—bigint####int8multirange—#######

- numrange—numeric####nummultirange—#######
- tsrange—timestamp without time zone####tsmultirange—#######
- tstzrange—timestamp with time zone####tstzmultirange—#######
- daterange—date####datemultirange—#######

###########################CREATE TYPE#########

8.17.2.

```
CREATE TABLE reservation (room int, during tsrange);
INSERT INTO reservation VALUES
    (1108, '[2010-01-01 14:30, 2010-01-01 15:30)');


-- ##
SELECT int4range(10, 20) @> 3;


-- ###
SELECT numrange(11.1, 22.2) && numrange(20.0, 30.0);


-- #####
SELECT upper(int8range(15, 25));


-- #######
SELECT int4range(10, 20) * int4range(15, 25);


-- #####
SELECT isempty(numrange(1, 5));
```

########################## 9.56## 9.58#########

8.17.3.

#########2################### ######################## ################### ######################################

######################[###########(####### ###########]###########)####### (### # 8.17.5#########)

##lower_inc###upper_inc###############################

8.17.4. ###(#####)##

####(,3]## ############ ## ################################ ################ ################################[,]#(,)######## #######+/-##### ################################+/-##################

#################################### ####timestamp####[today,infinity)### #timestamp##infinity##########[today,infinity]#[today,)#[today,]##### #infinity######

##lower_inf###upper_inf############################

8.17.5.

###########################

(*lower-bound*,*upper-bound*)
(*lower-bound*,*upper-bound*]
[*lower-bound*,*upper-bound*)
[*lower-bound*,*upper-bound*]
empty

######empty#####################(#
#######1####)######

lower-bound## ####*upper-bound*##

####"(#####)########### ### ################################## ## ################ (###SQL##################################2#######1#########
#########) ## ##### ##############""#####################################

(##################
#################)

##
######8.16.6##########

##

```
-- 3####7###################
SELECT '[3,7)'::int4range;

-- 3#7###################
SELECT '(3,7)'::int4range;

-- 1####4#####
SELECT '[4,4]'::int4range;

-- ##########('empty'#######)
SELECT '[4,4)'::int4range;
```

#################0###############({###})### ################### ###############
##########################1######################(##########################
#)#

####

```
SELECT '{}'::int4multirange;
SELECT '{[3,7)}'::int4multirange;
SELECT '{[3,7), [8,9)}'::int4multirange;
```

8.17.6.

################################ ##
################## #########2#####3######### ###2######(######, ######)######
###############3#######3######################### 3############################
#()###(]###[)######[]## ####

```
-- ##################################
-- #############
SELECT numrange(1.0, 14.0, '(]');


-- 3#############'[)'############
SELECT numrange(1.0, 14.0);


-- ####'(]'#########int8range########(####)###
-- ###################
SELECT int8range(1, 14, '(]');


-- ####NULL###########################
SELECT numrange(NULL, 2.2);
```

################################## ########################0########### ####

```
SELECT nummultirange();
SELECT nummultirange(numrange(1.0, 14.0));
SELECT nummultirange(numrange(1.0, 14.0), numrange(20.0, 25.0));
```

8.17.7.

#########integer#date################################ ##########2###########
1########2################## ###########################2####################
#######(############)##### ####numeric##timestamp########## (timestamp####
###)

############1## ############
#########
#[4,8]#(3,9)##################numeric###################

##################################
################ ##
##################

##########int4range#int8range####daterange#####################[)######
################################

8.17.8.

################### ## ##
##float8############################

```
CREATE TYPE floatrange AS RANGE (
    subtype = float8,
    subtype_diff = float8mi
);
```

```
SELECT '[1.234, 5.678]'::floatrange;
```

float8####################################

################################

######################B-tree##

##############################CREATE TYPE#####canonical(###)############## ### ### ####################[1, 7]#[1, 8)################### ###################### ################################## ## ##################################### ###timestamp#############1############ ##############1###

###GiST###SP-GiST############################subtype_diff########## (#### #####subtype_diff##) ######### ##2#################(####*X*##*Y*)#float8########## ########float8############# #float8mi## ########################### ############ ##########subtype_diff## #######1######2################################

subtype_diff#######################

```
CREATE FUNCTION time_subtype_diff(x time, y time) RETURNS float8 AS
'SELECT EXTRACT(EPOCH FROM (x - y))' LANGUAGE sql STRICT IMMUTABLE;

CREATE TYPE timerange AS RANGE (
    subtype = time,
    subtype_diff = time_subtype_diff
);

SELECT '[11:10, 23:00]'::timerange;
```

##################CREATE TYPE#########

8.17.9.

##########GiST###SP-GiST################## GiST############################## # ###GiST#############

```
CREATE INDEX reservation_idx ON reservation USING GIST (during);
```

#######GiST####SP-GiST################################## =#&&#<@#@>#<<#>>#-|-#&<####&># ########GiST################################ ##### #######GiST###############GiST## ############### &&#<@#@>#<<#>>#-|-#&<####&># ######### 9.56##########

##############B-tree######################## ################################## ## ########B-tree#######<###>## ####B-tree###

8.17.10.

UNIQUE################################## ######(exclude)################### (CREATE TABLE ... CONSTRAINT ... EXCLUDE########)# ######################## ################## ####

```
CREATE TABLE reservation (
    during tsrange,
    EXCLUDE USING GIST (during WITH &&)
);
```

##################################

```
INSERT INTO reservation VALUES
    ('[2010-01-01 11:30, 2010-01-01 15:00)');
INSERT 0 1

INSERT INTO reservation VALUES
    ('[2010-01-01 14:45, 2010-01-01 15:45)');
ERROR:  conflicting key value violates exclusion constraint
 "reservation_during_excl"
DETAIL:  Key (during)=(["2010-01-01 14:45:00","2010-01-01 15:45:00"))
 conflicts
with existing key (during)=(["2010-01-01 11:30:00","2010-01-01
 15:00:00")).
```

btree_gist#### ##########################
##########
####btree_gist##

```
CREATE EXTENSION btree_gist;
CREATE TABLE room_reservation (
    room text,
    during tsrange,
    EXCLUDE USING GIST (room WITH =, during WITH &&)
);

INSERT INTO room_reservation VALUES
    ('123A', '[2010-01-01 14:00, 2010-01-01 15:00)');
INSERT 0 1

INSERT INTO room_reservation VALUES
    ('123A', '[2010-01-01 14:30, 2010-01-01 15:30)');
ERROR:  conflicting key value violates exclusion constraint
 "room_reservation_room_during_excl"
DETAIL:  Key (room, during)=(123A, ["2010-01-01 14:30:00","2010-01-01
 15:30:00")) conflicts
with existing key (room, during)=(123A, ["2010-01-01 14:00:00","2010-01-01
 15:00:00")).

INSERT INTO room_reservation VALUES
    ('123B', '[2010-01-01 14:30, 2010-01-01 15:30)');
INSERT 0 1
```

8.18.

############################ ## #
################—################################## ########################
################################

#############################

```
CREATE DOMAIN posint AS integer CHECK (VALUE > 0);
CREATE TABLE mytable (id posint);
INSERT INTO mytable VALUES(1);   -- works
INSERT INTO mytable VALUES(-1);  -- fails
```

########mytable.id - 1####posint####integer########## ##################posint######### #(mytable.id - 1)::posint########### #########id#####1################## ##

#######CREATE DOMAIN#######

8.19.

#########OID##PostgreSQL########################## oid################## oid############## reg#############oid########### 8.26################

oid###########4################ ## ####################

oid######################## ## ##

OID############################## ##########oid############################ ######################## ################OID############ ####mytable####### #pg_attribute########################

```
SELECT * FROM pg_attribute WHERE attrelid = 'mytable'::regclass;
```

#################

```
SELECT * FROM pg_attribute
  WHERE attrelid = (SELECT oid FROM pg_class WHERE relname = 'mytable');
```

################################ #######mytable################OID######## #####SELECT######## regclass## ## ########OID#regclass#############OID############

#8.26

##	##	##	###
oid	###	##########	564182
regclass	pg_class	######	pg_type
regcollation	pg_collation	###	"POSIX"
regconfig	pg_ts_config	#######	english
regdictionary	pg_ts_dict	#######	simple
regnamespace	pg_namespace	#####	pg_catalog
regoper	pg_operator	####	+
regoperator	pg_operator	#########	*(integer, integer) or -(NONE,integer)
regproc	pg_proc	###	sum
regprocedure	pg_proc	########	sum(int4)
regrole	pg_authid	####	smithee
regtype	pg_type	######	integer

##################OID##################################### ################## ################################### ####myschema.mytable#regclass#####(############)####

################myschema.mytable#####mytable##########
regproc#regoper## ####
####regprocedure###regoperator########### regoperator#################
#NONE################

##########
SQL############################## ############SQL########################## #
###Foo(F####)##################OID#' "Foo" (int, integer)
'::regprocedure######### ###"Foo"(integer,integer)######## ############
##############

PostgreSQL#######################################OID############ regclass(##
#####OID#########)################ ##########OID##############################
############### #### nextval(regclass)##############OID####################
######

```
nextval('foo')              #####foo####
nextval('FOO')              ####
nextval('"Foo"')            #####Foo####
nextval('myschema.foo')     myschema.foo####
nextval('"myschema".foo')   ####
nextval('foo')              foo#########
```

> ##
>
> ################################regclass#(########)######### #########
> OID## ############(early
> binding)################################## ##############################
> (late binding)############# ######################regclass#####text#######
> ########
>
> ```
> nextval('foo'::text)      foo is looked up at runtime
> ```
>
> to_regclass()############################## #### 9.74########

regclass##################OID#######information_schema##############OID
######### ########OID######pg_relation_size()################ ###########
################

```
SELECT table_schema, table_name,
       pg_relation_size((quote_ident(table_schema) || '.' ||
                         quote_ident(table_name))::regclass)
FROM information_schema.tables
WHERE ...
```

quote_ident()################### ###########

```
SELECT pg_relation_size(table_name)
FROM information_schema.tables
WHERE ...
```

############## ##

#####OID#################### ######1###
############### ######nextval('my_seq'::regclass)##############PostgreSQL
########my_seq################### ## ###

#nextval('my_seq'::text)################ (##########regrole### ######### ############)

##########1########xid################xact####### ###xmin########xmax######## ##### ###########32####### #######64########xid8####### xid##### xid8######### ################################### ###66.1#########

########3#######cid############## ###cmin########cmax############ ########32 #######

################tid################### ###ctid############ ###ID############## ##################################

#########5.6########

8.20. pg_lsn#

pg_lsn##WAL######LSN(Log Sequence Number)################ ####XLogRecPtr### PostgreSQL##############

#####LSN#64bit#####WAL################## LSN#####16/B374D848############ #####8##16######### pg_lsn#####=#>##################### 2##LSN#-#### ####################2##WAL####bytes##### ###########+(pg_lsn,numeric)#-(pg_lsn,numeric)################ #####LSN#pg_lsn#########0/0#FFFFFFFF/FFFFFFFF#####################

8.21.

PostgreSQL################################### ################################ ################### ##############################SQL########################## ########### # 8.27################

#8.27 ######

##	##
any	############################
anyelement	############################(36.2.5###)#
anyarray	############################36.2.5###########
anynonarray	############################36.2.5###########
anyenum	###########################36.2.5###8.7######## ###
anyrange	######################(36.2.5 # 8.17#########)#
anymultirange	######################(36.2.5#8.17#########)#
anycompatible	## ######(36.2.5)#########)#
anycompatiblearray	## ######(36.2.5#########)#
anycompatiblenonarray	## #######(36.2.5#######)#
anycompatiblerange	## #####(36.2.5#8.17#######)#
anycompatiblemultirange	## #####(36.2.5#8.17#######)#

##	##
cstring	########C########################
internal	##################################
language_handler	#############language_handler##############
fdw_handler	#############fdw_handler##############
table_am_handler	#################table_am_handler##########
index_am_handler	################### index_am_handler########
tsm_handler	#################tsm_handler##############
record	###########################
trigger	######trigger##############
event_trigger	##########event_trigger##############
pg_ddl_command	#############DDL##############
void	################
unknown	#####################################

C###

######################void#record##############################trigger## #event_trigger############ ################################## ########## #36.2.5############

internal##SQL################## ##### ########1##internal###########SQL############## ######################## ######################## internal########1#########internal################## ######

#9#

PostgreSQL################################ ##################################### ########### ###### V################################### psql#`\df`#####`\do`###### ################################

#####################################

```
repeat ( text, integer ) → text
```

#####`repeat`#################################### ############################## #######

```
repeat('Pg', 4) → PgPgPgPg
```

###SQL# ################## ###############SQL## ########

9.1.

###############

```
boolean AND boolean → boolean
boolean OR boolean → boolean
NOT boolean → boolean
```

SQL#true#false############`null`#3############## ################

a	***b***	***a* AND *b***	***a* OR *b***
TRUE	TRUE	TRUE	TRUE
TRUE	FALSE	FALSE	TRUE
TRUE	NULL	NULL	TRUE
FALSE	FALSE	FALSE	FALSE
FALSE	NULL	FALSE	NULL
NULL	NULL	NULL	NULL

a	**NOT *a***
TRUE	FALSE
FALSE	TRUE
NULL	NULL

`AND`####`OR`######## ##################################### ##################### ##################################4.2.14###########

9.2.

9.1######################

#9.1

###	##
datatype < *datatype* → boolean	###
datatype > *datatype* → boolean	###
datatype <= *datatype* → boolean	#######
datatype >= *datatype* → boolean	#######
datatype = *datatype* → boolean	###
datatype <> *datatype* → boolean	#####
datatype != *datatype* → boolean	#####

> **##**
>
> <>#SQL################## !=################<>######## ####!=####<>#####
> #################

##################
####################

######################### ####integer >bigint###### ##########################
###

##########################boolean######## ####1 < 2 < 3###########3#####<###
########## ####BETWEEN##################

9.2#################### ######################SQL####################

#9.2

##
datatype BETWEEN *datatype* AND *datatype* → boolean ############## 2 BETWEEN 1 AND 3 → t 2 BETWEEN 3 AND 1 → f
datatype NOT BETWEEN *datatype* AND *datatype* → boolean #####BETWEEN##### 2 NOT BETWEEN 1 AND 3 → f
datatype BETWEEN SYMMETRIC *datatype* AND *datatype* → boolean 2################## 2 BETWEEN SYMMETRIC 3 AND 1 → t
datatype NOT BETWEEN SYMMETRIC *datatype* AND *datatype* → boolean 2################## 2 NOT BETWEEN SYMMETRIC 3 AND 1 → f
datatype IS DISTINCT FROM *datatype* → boolean NULL##################

##
1 IS DISTINCT FROM NULL → t (NULL####) NULL IS DISTINCT FROM NULL → f (NULL####)
datatype IS NOT DISTINCT FROM *datatype* → boolean NULL################ 1 IS NOT DISTINCT FROM NULL → f (NULL####) NULL IS NOT DISTINCT FROM NULL → t (NULL####)
datatype IS NULL → boolean ##NULL######## 1.5 IS NULL → f
datatype IS NOT NULL → boolean ##NULL############ 'null' IS NOT NULL → t
datatype ISNULL → boolean ##NULL################
datatype NOTNULL → boolean ##NULL###################
boolean IS TRUE → boolean ################### true IS TRUE → t NULL::boolean IS TRUE → f (NULL####)
boolean IS NOT TRUE → boolean ####################### true IS NOT TRUE → f NULL::boolean IS NOT TRUE → t (NULL####)
boolean IS FALSE → boolean ################### true IS FALSE → f NULL::boolean IS FALSE → f (NULL####)
boolean IS NOT FALSE → boolean ####################### true IS NOT FALSE → t NULL::boolean IS NOT FALSE → t (NULL####)
boolean IS UNKNOWN → boolean #################### true IS UNKNOWN → f NULL::boolean IS UNKNOWN → t (NULL####)
boolean IS NOT UNKNOWN → boolean ####################### true IS NOT UNKNOWN → t NULL::boolean IS NOT UNKNOWN → f (NULL####)

BETWEEN###################

a BETWEEN *x* AND *y*

#

a >= *x* AND *a* <= *y*

BETWEEN############################ BETWEEN SYMMETRIC##AND######## ############################BETWEEN###### ##############2##################

BETWEEN##

##
BETWEEN####AND######AND####################### ##########BETWEEN###2 ######################## BETWEEN##################################

########NULL##############################null####### ###7 = NULL#null######7 <> NULL###### #############IS [NOT] DISTINCT FROM###########

a IS DISTINCT FROM *b*
a IS NOT DISTINCT FROM *b*

#NULL######IS DISTINCT FROM#<>######## ##########NULL################## NULL########## ####IS NOT DISTINCT FROM##NULL####=############NULL##### ######NULL########## #############NULL#########################

##NULL#NULL####################

expression IS NULL
expression IS NOT NULL

#######################

expression ISNULL
expression NOTNULL

NULL#NULL##################*expression* = NULL########## #NULL############ ########################

###
#############*expression* = NULL##*expression*#NULL########### ################## ##############SQL################### ########### #transform_null_equals################### ##########PostgreSQL#x = NULL##x IS NULL######

######*expression*###########NULL##################NULL######IS NULL##### ########NULL##################NULL######IS NOT NULL##### #########IS NULL#IS NOT NULL######################### ###null##null###################### ########## #####

```
SELECT ROW(1,2.5,'this is a test') = ROW(1, 3, 'not the same');

SELECT ROW(table.*) IS NULL FROM table;  -- detect all-null rows

SELECT ROW(table.*) IS NOT NULL FROM table;  -- detect all-non-null rows

SELECT NOT(ROW(table.*) IS NOT NULL) FROM TABLE; -- detect at least one
 null in rows
```

#############*row* IS DISTINCT FROM NULL###*row* IS NOT DISTINCT FROM NULL# ############### ##############################NULL###############

```
   ###############


boolean_expression IS TRUE
boolean_expression IS NOT TRUE
boolean_expression IS FALSE
boolean_expression IS NOT FALSE
boolean_expression IS UNKNOWN
boolean_expression IS NOT UNKNOWN
```

##################NULL#####NULL############ NULL########################### ## IS UNKNOWN#IS NOT UNKNOWN################################IS NULL#IS NOT NULL#################

9.3##########################

#9.3

##	
	##
	#
num_nonnulls (VARIADIC "any") → integer	
	#NULL########
	num_nonnulls(1, NULL, 2) → 2
num_nulls (VARIADIC "any") → integer	
	NULL#######
	num_nulls(1, NULL, 2) → 1

9.3.

PostgreSQL######################## ####################/#################### #######

9.4######################### #########*numeric_type*################### #smallint#integer#bigint#numeric#real#double precision########### *integral_type*####################smallint#integer#bigint########### ### ############################## integer + numeric##############################

#9.4

###	
	##
	#
numeric_type + *numeric_type* → *numeric_type*	
	#

```
###
      ##
      #
```

```
      2 + 3 → 5
```

```
+ numeric_type → numeric_type
      #########
      + 3.5 → 3.5
```

```
numeric_type - numeric_type → numeric_type
      #
      2 - 3 → -1
```

```
- numeric_type → numeric_type
      ##
      - (-4) → 4
```

```
numeric_type * numeric_type → numeric_type
      #
      2 * 3 → 6
```

```
numeric_type / numeric_type → numeric_type
      ############################
      5.0 / 2 → 2.5000000000000000
      5 / 2 → 2
      (-5) / 2 → -2
```

```
numeric_type % numeric_type → numeric_type
      ######smallint#integer#bigint#numeric#####
      5 % 4 → 1
```

```
numeric ^ numeric → numeric
double precision ^ double precision → double precision
      ##
      2 ^ 3 → 8
      #####################^############
      2 ^ 3 ^ 3 → 512
      2 ^ (3 ^ 3) → 134217728
```

```
|/ double precision → double precision
      ###
      |/ 25.0 → 5
```

```
||/ double precision → double precision
      ###
      ||/ 64.0 → 4
```

```
@ numeric_type → numeric_type
      ###
      @ -5.0 → 5.0
```

```
integral_type & integral_type → integral_type
      ######AND
      91 & 15 → 11
```

#
integral_type | *integral_type* → *integral_type* ######OR 32 | 3 → 35
integral_type # *integral_type* → *integral_type* ########### 17 # 5 → 20
~ *integral_type* → *integral_type* ######NOT ~1 → -2
integral_type << integer → *integral_type* ######### 1 << 4 → 16
integral_type >> integer → *integral_type* ######### 8 >> 2 → 2

9.5############### ################################ ########################### ############ ## double precision###### ################C###

#9.5

#
abs (*numeric_type*) → *numeric_type* ### abs(-17.4) → 17.4
cbrt (double precision) → double precision ### cbrt(64.0) → 4
ceil (numeric) → numeric ceil (double precision) → double precision #################### ceil(42.2) → 43 ceil(-42.8) → -42
ceiling (numeric) → numeric ceiling (double precision) → double precision #####################ceil#### ceiling(95.3) → 96
degrees (double precision) → double precision ######### degrees(0.5) → 28.64788975654116

#
div (*y* numeric, *x* numeric) → numeric *y*/*x*#### #0######### div(9, 4) → 2
erf (double precision) → double precision #### erf(1.0) → 0.8427007929497149
erfc (double precision) → double precision #######1-erf(x)################ erfc(1.0) → 0.15729920705028513
exp (numeric) → numeric exp (double precision) → double precision ###e########### exp(1.0) → 2.7182818284590452
factorial (bigint) → numeric ## factorial(5) → 120
floor (numeric) → numeric floor (double precision) → double precision #################### floor(42.8) → 42 floor(-42.8) → -43
gcd (*numeric_type*, *numeric_type*) → *numeric_type* ####################### ######0##0#### integer#bigint#numeric##### gcd(1071, 462) → 21
lcm (*numeric_type*, *numeric_type*) → *numeric_type* ########################### ######0##0#### integer#bigint#numeric# #### lcm(1071, 462) → 23562
ln (numeric) → numeric ln (double precision) → double precision #### ln(2.0) → 0.6931471805599453
log (numeric) → numeric log (double precision) → double precision 10####### log(100) → 2
log10 (numeric) → numeric log10 (double precision) → double precision 10########log#### log10(1000) → 3

##
log (*b* numeric, *x* numeric) → numeric *b*#####*x*### log(2.0, 64.0) → 6.0000000000000000
min_scale (numeric) → integer ##############################10##### min_scale(8.4100) → 2
mod (*y numeric_type*, *x numeric_type*) → *numeric_type* *y*/*x*#### smallint#integer#bigint#numeric##### mod(9, 4) → 1
pi () → double precision π#### pi() → 3.141592653589793
power (*a* numeric, *b* numeric) → numeric power (*a* double precision, *b* double precision) → double precision *a*#*b*# power(9, 3) → 729
radians (double precision) → double precision ######### radians(45.0) → 0.7853981633974483
round (numeric) → numeric round (double precision) → double precision ########### numeric################# double precision############ ########### ####################(round to nearest even)#### round(42.4) → 42
round (*v* numeric, *s* integer) → numeric *v*#####*s*####### ################### round(42.4382, 2) → 42.44 round(1234.56, -1) → 1230
scale (numeric) → integer ############10##### scale(8.4100) → 4
sign (numeric) → numeric sign (double precision) → double precision #####(-1, 0,#### +1) sign(-8.4) → -1
sqrt (numeric) → numeric sqrt (double precision) → double precision ### sqrt(2) → 1.4142135623730951
trim_scale (numeric) → numeric

##########################10######## trim_scale(8.4100) → 8.41
trunc (numeric) → numeric trunc (double precision) → double precision ############### trunc(42.8) → 42 trunc(-42.8) → -42
trunc (*v* numeric, *s* integer) → numeric *v*######*s*###### trunc(42.4382, 2) → 42.43
width_bucket (*operand* numeric, *low* numeric, *high* numeric, *count* integer) → integer width_bucket (*operand* double precision, *low* double precision, *high* double precision, *count* integer) → integer *low*##*high*################*count*###########*operand*############### ## ############0###*count*+1#### width_bucket(5.35, 0.024, 10.06, 5) → 3
width_bucket (*operand* anycompatible, *thresholds* anycompatiblearray) → integer ######################*operand*################# ###################0# ### *operand*################################## *thresholds*########## ################# ##################

9.6###############

#9.6

random ( ) → double precision 0.0 <= x < 1.0######### random() → 0.897124072839091
random (*min* integer, *max* integer) → integer random (*min* bigint, *max* bigint) → bigint random (*min* numeric, *max* numeric) → numeric ######*min* <= x *max* ########### ##numeric######*min* ###*max* ########### ############# random(1, 10) → 7 random(-0.499, 0.499) → 0.347
random_normal ([*mean* double precision [, *stddev* double precision]]) → double precision ############################ *mean*########0.0#*stddev*########1.0###

#
random_normal(0.0, 1.0) → 0.051285419
setseed(double precision) → void ###random()#random_normal()############(#)##########-1.0##1.0####### ############ setseed(0.12345)

#####[64.812239]#random()#################### ##########################
##pgcrypto############## setseed()#########################random()######
#setseed()######################### ########setseed()##################
#random()## #######random_normal()##
####### ####### 9.6########random()#random_normal()#################### ##
########################### ##############pgcrypto############## setseed()#####
##setseed()#################### ##
######setseed()###

########## 9.7###### ##################################2########

#9.7

#
acos(double precision) → double precision ############## acos(1) → 0
acosd(double precision) → double precision ########## acosd(0.5) → 60
asin(double precision) → double precision ############## asin(1) → 1.5707963267948966
asind(double precision) → double precision ########## asind(0.5) → 30
atan(double precision) → double precision ############## atan(1) → 0.7853981633974483
atand(double precision) → double precision ########## atand(1) → 45
atan2(*y* double precision, *x* double precision) → double precision *y*/*x*############## atan2(1, 0) → 1.5707963267948966
atan2d(*y* double precision, *x* double precision) → double precision *y*/*x*########## atan2d(1, 0) → 90

cos(double precision)→double precision ############ cos(0)→1
cosd(double precision)→double precision ######## cosd(60)→0.5
cot(double precision)→double precision ############ cot(0.5)→1.830487721712452
cotd(double precision)→double precision ######## cotd(45)→1
sin(double precision)→double precision ############ sin(1)→0.8414709848078965
sind(double precision)→double precision ######## sind(30)→0.5
tan(double precision)→double precision ############ tan(1)→1.5574077246549023
tand(double precision)→double precision ######## tand(45)→1

##
##########################radians()#degrees()####### ############ ##sind(30)##############################

9.8################

#9.8

sinh(double precision)→double precision ##### sinh(1)→1.1752011936438014
cosh(double precision)→double precision ##### cosh(0)→1

#
tanh(double precision) → double precision ##### tanh(1) → 0.7615941559557649
asinh(double precision) → double precision ###### asinh(1) → 0.881373587019543
acosh(double precision) → double precision ###### acosh(1) → 0
atanh(double precision) → double precision ###### atanh(0.5) → 0.5493061443340548

9.4.

################################## ########character#####character varying#######text########### ######################text################ ######### ####character varying################ character################ ####text######character###################

SQL## ####### 9.9######### ## PostgreSQL## 9.10###########

> ##
>
> ########||### 9.9## ####### ################text#################

#9.9 SQL#########

##/### ## #
text || text → text 2########### 'Post' || 'greSQL' → PostgreSQL
text || anynonarray → text anynonarray || text → text ###################2########### ########################||######## ######### ###########################text############ 'Value: ' || 42 → Value: 42
btrim(*string* text [, *characters* text]) → text *string*##*characters*## btrim('xyxtrimyyx', 'xyz') → trim
text IS [NOT] [*form*] NORMALIZED → boolean

##/###   ##   #
#######Unicode################## ######*form*################NFC (#### ###)#NFD#NFKC####NFKD### ################UTF8########### ########### ##   U&'\0061\0308bc' IS NFD NORMALIZED → t
bit_length (text) → integer   ##############octet_length#8#####   bit_length('jose') → 32
char_length (text) → integer   character_length (text) → integer   #############   char_length('josé') → 4
lower (text) → text   ##################################   lower('TOM') → tom
lpad (*string* text, *length* integer [, *fill* text]) → text   ##*fill*#######################*string*#*length*######## *string*## #*length*#########################   lpad('hi', 5, 'xy') → xyxhi
ltrim (*string* text [, *characters* text]) → text   *string*##*characters*##   ltrim('zzzytest', 'xyz') → test
normalize (text [, *form*]) → text   #######Unicode######### ######*form*################NFC (###### #)#NFD#NFKC####NFKD### ################UTF8###########   normalize(U&'\0061\0308bc', NFC) → U&'\00E4bc'
octet_length (text) → integer   #############   octet_length('josé') → 5#############UTF8####
octet_length (character) → integer   ############# ############character#####################   octet_length('abc '::character(4)) → 4
overlay (*string* text PLACING *newsubstring* text FROM *start* integer [FOR *count* integer]) → text   *string*#*start*####*count*###*newsubstring*######## *count*##### #*newsubstring*##############   overlay('Txxxxas' placing 'hom' from 2 for 4) → Thomas
position (*substring* text IN *string* text) → integer   *string*##*substring*######################0##############   position('om' in 'Thomas') → 3
rpad (*string* text, *length* integer [, *fill* text]) → text   ##*fill*#######################*string*#*length*######## *string*###*length*# ###################

##/### ## #
rpad('hi', 5, 'xy') → hixyx
rtrim (*string* text [, *characters* text]) → text *string*##*characters*## rtrim('testxxzx', 'xyz') → test
substring (*string* text [FROM *start* integer][FOR *count* integer]) → text *start*########*start*########*string*########### *count*########*count*# ########## #####*start*#*count*############## substring('Thomas' from 2 for 3) → hom substring('Thomas' from 3) → omas substring('Thomas' for 2) → Th
substring (*string* text FROM *pattern* text) → text POSIX######################9.7.3######### substring('Thomas' from '...$') → mas
substring (*string* text SIMILAR *pattern* text ESCAPE *escape* text) → text substring (*string* text FROM *pattern* text FOR *escape* text) → text SQL######################9.7.2######### ######SQL:2003###########2## ####SQL:1999######################### substring('Thomas' similar '%#"o_a#"_' escape '#') → oma
trim ([LEADING | TRAILING | BOTH][*characters* text] FROM *string* text) → text *string*##*characters*## #BOTH############### trim(both 'xyz' from 'yxTomxx') → Tom
trim ([LEADING | TRAILING | BOTH][FROM] *string* text [, *characters* text]) → text ###trim()######## trim(both from 'yxTomxx', 'xyz') → Tom
unicode_assigned (text) → text ################Unicode##################true###########false#### ## ################UTF8############
upper (text) → text ################################## upper('tom') → TOM

9.10##################### ######### 9.9#######SQL##################### ### ###9.7###################12####################

#9.10

##/### ## #
text ^@ text → boolean #######2###################starts_with()########

##/###
##
#

'alphabet' ^@ 'alph' → t

ascii (text) → integer
################### UTF8###########Unicode############ ##############
#######ASCII############
ascii('x') → 120

chr (integer) → text
################ UTF8##########Unicode############## ###############
########ASCII############ chr(0)################################
chr(65) → A

concat (*val1* "any" [, *val2* "any" [, ...]]) → text
#################### NULL#########
concat('abcde', 2, NULL, 22) → abcde222

concat_ws (*sep* text, *val1* "any" [, *val2* "any" [, ...]]) → text
###################### ####################NULL########### #####
NULL##########
concat_ws(',', 'abcde', 2, NULL, 22) → abcde,2,22

format (*formatstr* text [, *formatarg* "any" [, ...]]) → text
######################### 9.4.1######### #####C#####sprintf#######
format('Hello %s, %1$s', 'World') → Hello World, World

initcap (text) → text
############################## ####################################
##
initcap('hi THOMAS') → Hi Thomas

left (*string* text, *n* integer) → text
#######*n*####### *n*##############|*n*|################
left('abcde', 2) → ab

length (text) → integer
##############
length('jose') → 4

md5 (text) → text
###MD5#######16##########
md5('abc') → 900150983cd24fb0d6963f7d28e17f72

parse_ident (*qualified_identifier* text [, *strict_mode* boolean DEFAULT
true]) → text[]
qualified_identifier################################ ###########
######################2########false###################### ########
############################## ############################## ######
##########name[]##############
parse_ident('"SomeSchema".someTable') →
{SomeSchema,sometable}

pg_client_encoding () → name
#######################

##/###   ##   #
pg_client_encoding() → UTF8
quote_ident(text) → text   #########SQL################################### ################ ## ##################### # 41.1#########   quote_ident('Foo bar') → "Foo bar"
quote_literal(text) → text   #########SQL## ############# ################### quote_literal#NULL#####NULL############## ##NULL############quote_nullable############# # 41.1#########   quote_literal(E'O\'Reilly') → 'O''Reilly'
quote_literal(anyelement) → text   ################################ ##############################   quote_literal(42.5) → '42.5'
quote_nullable(text) → text   #########SQL## ######NULL## ##NULL###### ############################### # 41.1#########   quote_nullable(NULL) → NULL
quote_nullable(anyelement) → text   ####################################NULL####NULL###### ########### ###################   quote_nullable(42.5) → '42.5'
regexp_count(*string* text, *pattern* text [, *start* integer [, *flags* text]])   → integer   *string*####POSIX####*pattern*############9.7.3#########   regexp_count('123456789012', '\d\d\d', 2) → 3
regexp_instr(*string* text, *pattern* text [, *start* integer [, *N* integer [,   *endoption* integer [, *flags* text [, *subexpr* integer]]]]]) → integer   POSIX####*pattern*#*N*#########*string*################0######9.7.3#### ######   regexp_instr('ABCDEF', 'c(.)(..)', 1, 1, 0, 'i') → 3   regexp_instr('ABCDEF', 'c(.)(..)', 1, 1, 0, 'i', 2) → 5
regexp_like(*string* text, *pattern* text [, *flags* text]) → boolean   POSIX####*pattern*####*string*################9.7.3#########   regexp_like('Hello World', 'world$', 'i') → t
regexp_match(*string* text, *pattern* text [, *flags* text]) → text[]   *string*####POSIX####*pattern*#######################9.7.3#########   regexp_match('foobarbequebaz', '(bar)(beque)') → {bar,beque}
regexp_matches(*string* text, *pattern* text [, *flags* text]) → setof   text[]   *string*####POSIX####*pattern*##################g################### ################9.7.3#########   regexp_matches('foobarbequebaz', 'ba.', 'g') →

##/###
##
#

{bar}
{baz}

regexp_replace (*string* text, *pattern* text, *replacement* text [, *start* integer] [, *flags* text]) → text
string####POSIX####*pattern*##########g##########################9.7.3#########
regexp_replace('Thomas', '.[mN]a.', 'M') → ThM

regexp_replace (*string* text, *pattern* text, *replacement* text, *start* integer, *N* integer [, *flags* text]) → text
POSIX####*pattern*#*N*###############*N*#0#######################9.7.3#######
regexp_replace('Thomas', '.', 'X', 3, 2) → ThoXas

regexp_split_to_array (*string* text, *pattern* text [, *flags* text]) → text[]
POSIX#############*string*################ 9.7.3#########
regexp_split_to_array('hello world', '\s+') → {hello,world}

regexp_split_to_table (*string* text, *pattern* text [, *flags* text]) → setof text
POSIX#############*string*####### ####9.7.3#########
regexp_split_to_table('hello world', '\s+') →

hello
world

regexp_substr (*string* text, *pattern* text [, *start* integer [, *N* integer [, *flags* text [, *subexpr* integer]]]]) → text
POSIX####*pattern*#*N*#######*string*####################NULL######
9.7.3#########
regexp_substr('ABCDEF', 'c(.)(..)', 1, 1, 'i') → CDEF
regexp_substr('ABCDEF', 'c(.)(..)', 1, 1, 'i', 2) → EF

repeat (*string* text, *number* integer) → text
#####*number*####*string*#######
repeat('Pg', 4) → PgPgPgPg

replace (*string* text, *from* text, *to* text) → text
string#######*from*######*to*###########
replace('abcdefabcdef', 'cd', 'XX') → abXXefabXXef

reverse (text) → text
##############
reverse('abcde') → edcba

right (*string* text, *n* integer) → text
#######*n*####### *n*###############|*n*|################
right('abcde', 2) → de

split_part (*string* text, *delimiter* text, *n* integer) → text

##/###
##
#

string#*delimiter*##########*n*########1################ *n*########|*n*|
#############

split_part('abc~@~def~@~ghi', '~@~', 2) → def

split_part('abc,def,ghi,jkl', ',', -2) → ghi

---

starts_with (*string* text, *prefix* text) → boolean
string#*prefix*##############

starts_with('alphabet', 'alph') → t

---

string_to_array (*string* text, *delimiter* text [, *null_string* text]) →
text[]
string#*delimiter*#############text######## *delimiter*#NULL#
##*string*################ *delimiter*######*string*############
null_string#####NULL#####################NULL#########
array_to_string#########

string_to_array('xx~~yy~~zz', '~~', 'yy') → {xx,NULL,zz}

---

string_to_table (*string* text, *delimiter* text [, *null_string* text]) →
setof text
string#*delimiter*#############text########### *delimiter*#NULL###
###*string*############## *delimiter*######*string*################
null_string#####NULL#####################NULL#########

string_to_table('xx~^~yy~^~zz', '~^~', 'yy') →

```
xx
NULL
zz
```

---

strpos (*string* text, *substring* text) → integer
string#####*substring*#############*substring*###############
#position(*substring* in *string*)###########################

strpos('high', 'ig') → 2

---

substr (*string* text, *start* integer [, *count* integer]) → text
string#*start*###################*count*###################
#substring(*string* from *from* for *count*)######

substr('alphabet', 3) → phabet

substr('alphabet', 3, 2) → ph

---

to_ascii (*string* text) → text

to_ascii (*string* text, *encoding* name) → text

to_ascii (*string* text, *encoding* integer) → text
string#######################ASCII###### *encoding*#########
#################################### #######################
LATIN1#LATIN2#LATIN9#WIN1250################### ##############
##unaccent###############

to_ascii('Karél') → Karel

---

to_bin (integer) → text

to_bin (bigint) → text
#####[63.636364]######16########## ######## 2 ####################

to_bin(2147483647) → 11111111111111111111111111111111

##/### ## #
to_bin(-1234) → 11111111111111111111101100101110
to_hex(integer) → text to_hex(bigint) → text #####[69.512195]######16########## ############## 2 #### 16 ########## to_hex(2147483647) → 7fffffff to_hex(-1234) → fffffb2e
to_oct(integer) → text to_oct(bigint) → text #####[65.789474]######16########## ######### 2 #### 8 ######### to_oct(2147483647) → 17777777777 to_oct(-1234) → 37777775456
translate(*string* text, *from* text, *to* text) → text *from*##########*string*#########*to*#################### ##*from*#*to*## #####*from*######################### translate('12345', '143', 'ax') → a2x5
unistr(text) → text ##########Unicode######### Unicode####*XXXX* (16#4#)#\+*XXXXXX* (16#6#)#\u*XXXX* (16#4#)#\U*XXXXXXXX*(16#8#)######## ################2########## ###### ################## #############UTF-8#########################Unicode################### ########### ################## #####Unicode##########4.1.2.3##################### unistr('d\0061t\+000061') → data unistr('d\u0061t\U00000061') → data

concat#concat_ws###format###Variadic###########VARIADIC################## ##########################36.5.6########## ################################ ## variadic#####NULL#####concat###concat_ws#NULL######format#NULL########## #####

9.21##string_agg#########bytea######### 9.13##############

9.4.1. format

##format##C###sprintf####################################

format(*formatstr* text [, *formatarg* "any" [, ...]])

formatstr################################## ################################## ################## ## ############## #####*formatarg*## ##############

#########%##################

%[*position*][*flags*][*width*]*type*

#####################

position (####)

n$########*n*################ #######*formatstr*########## *position*######
##########################

flags (####)

###########flag#######(-)#######
###################### ###*width*##########################

width (####)

########################## #######################flag#-############ ####
########################### ##
(*)####*n*###############**n*$###########

##############
#abs(*width*)############flag#-##################

type (##)

################################## ################

- s##################### NULL##############
- I#####SQL############################ NULL#######quote_ident#######
- L#####SQL################# NULL##########NULL######quote_nullable#####

#########################%%#####%#####################

##########################

```
SELECT format('Hello %s', 'World');
Result: Hello World

SELECT format('Testing %s, %s, %s, %%', 'one', 'two', 'three');
Result: Testing one, two, three, %

SELECT format('INSERT INTO %I VALUES(%L)', 'Foo bar', E'O\'Reilly');
Result: INSERT INTO "Foo bar" VALUES('O''Reilly')

SELECT format('INSERT INTO %I VALUES(%L)', 'locations', 'C:\Program
 Files');
Result: INSERT INTO locations VALUES('C:\Program Files')
```

width######flag#-##############

```
SELECT format('|%10s|', 'foo');
Result: |       foo|

SELECT format('|%-10s|', 'foo');
Result: |foo       |

SELECT format('|%*s|', 10, 'foo');
Result: |       foo|

SELECT format('|%*s|', -10, 'foo');
Result: |foo       |

SELECT format('|%-*s|', 10, 'foo');
Result: |foo       |
```

```
SELECT format('|%-*s|', -10, 'foo');
Result: |foo       |
```

#####*position*################

```
SELECT format('Testing %3$s, %2$s, %1$s', 'one', 'two', 'three');
Result: Testing three, two, one

SELECT format('|%*2$s|', 'foo', 10, 'bar');
Result: |       bar|

SELECT format('|%1$*2$s|', 'foo', 10, 'bar');
Result: |       foo|
```

##C##sprintf######PostgreSQL#format#################*position*############## ########################## *position*### ### ####format################################ #######

```
SELECT format('Testing %3$s, %2$s, %s', 'one', 'two', 'three');
Result: Testing three, two, three
```

%I ### %L##############SQL################# # 41.1#########

9.5.

###############bytea############################## ########################## #################

SQL### #### 9.11######### ##PostgreSQL## ##################################### 9.12##########

#9.11 SQL#############

##/### ## #
bytea || bytea → bytea 2############### '\x123456'::bytea || '\x789a00bcde'::bytea → \x123456789a00bcde
bit_length (bytea) → integer ##############octet_length#8### bit_length('\x123456'::bytea) → 24
btrim (*bytes* bytea, *bytesremoved* bytea) → bytea *bytes*##*bytesremoved*############################## btrim('\x1234567890'::bytea, '\x9012'::bytea) → \x345678
ltrim (*bytes* bytea, *bytesremoved* bytea) → bytea *bytes*#####*bytesremoved*###################### ltrim('\x1234567890'::bytea, '\x9012'::bytea) → \x34567890
octet_length (bytea) → integer ################## octet_length('\x123456'::bytea) → 3

##/###   ##   #
overlay (*bytes* bytea PLACING *newsubstring* bytea FROM *start* integer [FOR *count* integer]) → bytea   *bytes#start#######count####newsubstring#######* *count#####* *#newsubstring##############*   overlay('\x1234567890'::bytea placing '\002\003'::bytea from 2 for 3) → \x12020390
position (*substring* bytea IN *bytes* bytea) → integer   *bytes##substring#################################0######*   position('\x5678'::bytea in '\x1234567890'::bytea) → 3
rtrim (*bytes* bytea, *bytesremoved* bytea) → bytea   *bytes##bytesremoved#############################*   rtrim('\x1234567890'::bytea, '\x9012'::bytea) → \x12345678
substring (*bytes* bytea [FROM *start* integer] [FOR *count* integer]) → bytea   *start########start########bytes########### count########count## ######## #####start#count##############*   substring('\x1234567890'::bytea from 3 for 2) → \x5678
trim ([LEADING | TRAILING | BOTH] *bytesremoved* bytea FROM *bytes* bytea) → bytea   *bytes##bytesremoved#################################BOTH######### ######*   trim('\x9012'::bytea from '\x1234567890'::bytea) → \x345678
trim ([LEADING | TRAILING | BOTH] [FROM] *bytes* bytea, *bytesremoved* bytea) → bytea   ###trim()########   trim(both from '\x1234567890'::bytea, '\x9012'::bytea) → \x345678

9.12##################### ######### 9.11#######SQL#######################

#9.12

#
bit_count (*bytes* bytea) → bigint   ############################popcount##############   bit_count('\x1234567890'::bytea) → 15
get_bit (*bytes* bytea, *n* bigint) → integer   #######n##############   get_bit('\x1234567890'::bytea, 30) → 1
get_byte (*bytes* bytea, *n* integer) → integer   #######n##############   get_byte('\x1234567890'::bytea, 4) → 144
length (bytea) → integer

length('\x1234567890'::bytea) → 5
length (*bytes* bytea, *encoding* name) → integer #####*encoding*############################ length('jose'::bytea, 'UTF8') → 4
md5 (bytea) → text #######MD5#######16########## md5('Th\000omas'::bytea) → 8ab2d3c9689aaf18b4958c334c82d8b1
set_bit (*bytes* bytea, *n* bigint, *newvalue* integer) → bytea #######n######*newvalue*##### set_bit('\x1234567890'::bytea, 30, 0) → \x1234563890
set_byte (*bytes* bytea, *n* integer, *newvalue* integer) → bytea #######n######*newvalue*##### set_byte('\x1234567890'::bytea, 4, 64) → \x1234567840
sha224 (bytea) → bytea #######SHA-224 ########## sha224('abc'::bytea) → \x23097d223405d8228642a477bda2 55b32aadbce4bda0b3f7e36c9da7
sha256 (bytea) → bytea #######SHA-256 ########## sha256('abc'::bytea) → \xba7816bf8f01cfea414140de5dae2223 b00361a396177a9cb410ff61f20015ad
sha384 (bytea) → bytea #######SHA-384 ########## sha384('abc'::bytea) → \xcb00753f45a35e8bb5a03d699ac65007 272c32ab0eded1631a8b605a43ff5bed 8086072ba1e7cc2358baeca134c825a7
sha512 (bytea) → bytea #######SHA-512 ########## sha512('abc'::bytea) → \xddaf35a193617abacc417349ae204131 12e6fa4e89a97ea20a9eeee64b55d39a 2192992a274fc1a836ba3c23a3feebbd 454d4423643ce80e2a9ac94fa54ca49f
substr (*bytes* bytea, *start* integer [, *count* integer]) → bytea *start*########*bytes*############ *count*########*count*########### #substring(*bytes* from *start* for *count*)###### substr('\x1234567890'::bytea, 3, 2) → \x5678

get_byte#set_byte##############0########## get_bit#set_bit##############
#########0##################15##################

##########md5#16########text########SHA-2###bytea###### ##############
#encode#decode######## #####16#################
##encode(sha256('abc'),'hex')#bytea#####decode(md5('abc'), 'hex')####
####

9.13###### ######## #text##########################bytea#################################

#9.13 ####/###########

##	
convert (*bytes* bytea, *src_encoding* name, *dest_encoding* name) → bytea ######*src_encoding*#######################*dest_encoding*######## ##############23.3.4########## convert('text_in_utf8', 'UTF8', 'LATIN1') → \x746578745f696e5f75746638	
convert_from (*bytes* bytea, *src_encoding* name) → text ######*src_encoding*## ###23.3.4########## convert_from('text_in_utf8', 'UTF8') → text_in_utf8	
convert_to (*string* text, *dest_encoding* name) → bytea text##########################*dest_encoding*#################### ##23.3.4########## convert_to('some_text', 'UTF8') → \x736f6d655f74657874	
encode (*bytes* bytea, *format* text) → text ################################*format*###base64#escape#hex### encode('123\000\001', 'base64') → MTIzAAE=	
decode (*string* text, *format* text) → bytea #################### *format*##encode######	
	decode('MTIzAAE=', 'base64') → \x3132330001

encode#decode#####################

base64

base64###RFC 2045 6.8#[1]###### RFC###########76######### ####MIME CRLF#### ##################### decode############################## ############### ###decode####base64##################

escape

escape#######################8###########(*nnn*)################### #### ############## ##########################3##8############decode######### ##################

hex

hex#########4#######################0##f#16######## encode###a-f#16##### ######## ########8#######encode############# decode###a-f############### ##### decode################16##################

9.21######string_agg#33.4#####################

9.6.

[1] https://datatracker.ietf.org/doc/html/rfc2045#section-6.8

####bit##bit varying#################################### ######bit#######
######bit varying############## ######## 9.1############## 9.14###############
###

#9.14

bit || bit → bit ## B'10001' || B'011' → 10001011
bit & bit → bit ######AND################## B'10001' & B'01101' → 00001
bit | bit → bit ######OR################## B'10001' | B'01101' → 11101
bit # bit → bit ############################## B'10001' # B'01101' → 11100
~ bit → bit ######## ~ B'10001' → 01110
bit << integer → bit ###################### B'10001' << 3 → 01000
bit >> integer → bit ###################### B'10001' >> 2 → 00100

###################### 9.15####################

#9.15

bit_count (bit) → bigint ###########################popcount############## bit_count(B'10111') → 4
bit_length (bit) → integer ################# bit_length(B'10111') → 5
length (bit) → integer ################ length(B'10111') → 5

#
octet_length (bit) → integer ################# octet_length(B'1011111011') → 2
overlay (*bits* bit PLACING *newsubstring* bit FROM *start* integer [FOR *count* integer]) → bit *bits#start#######count####newsubstring#######* *count#####* *#newsubstring##############* overlay(B'01010101010101010' placing B'11111' from 2 for 3) → 0111110101010101010
position (*substring* bit IN *bits* bit) → integer *bits##substring*################################0###### position(B'010' in B'000001101011') → 8
substring (*bits* bit [FROM *start* integer] [FOR *count* integer]) → bit *start########bits#############* *count########count###########* ### *##start#count##############* substring(B'110010111111' from 3 for 2) → 00
get_bit (*bits* bit, *n* integer) → integer *######n*##################(###)#####0####### get_bit(B'101010101010101010', 6) → 1
set_bit (*bits* bit, *n* integer, *newvalue* integer) → bit *######n######newvalue##########*(###)#####0####### set_bit(B'101010101010101010', 6, 0) → 101010001010101010

####bit##################bit############## ####bit(n)###########n#######
#########

```
44::bit(10)                    0000101100
44::bit(3)                     100
cast(-44 as bit(12))           111111010100
'1110'::bit(4)::integer        14
```

###bit##########bit(1)###

9.7.

PostgreSQL################3#################SQL#LIKE##########SIMILAR TO##
##SQL:1999############POSIX######### ######################################
#####################################

###
##############################Perl####Tcl#########################

##

################
################### ##################################

SIMILAR TO#POSIX#####################SIMILAR TO##################
########

LIKE######2##

##3#################################### ##################################

9.7.1. LIKE

string LIKE *pattern* [ESCAPE *escape-character*]
string NOT LIKE *pattern* [ESCAPE *escape-character*]

LIKE######*pattern*#*string*############ ########NOT LIKE##LIKE############
############ #######NOT (*string* LIKE *pattern*)#########

pattern##########################pattern#################LIKE##############
pattern############_#######################%##0################

#######

```
'abc' LIKE 'abc'    true
'abc' LIKE 'a%'     true
'abc' LIKE '_b_'    true
'abc' LIKE 'c'      false
```

LIKE######################## ##
#######

##*pattern*#####################
##################### ########################ESCAPE######################
####################2######

##

standard_conforming_strings######off###
###4.1.2.1#########

###ESCAPE ''########################### ####################################
#####################################

SQL#######ESCAPE###
#ESCAPE######## #######PostgreSQL###############

################################LIKE#####ILIKE############### #####SQL###
##PostgreSQL######

~~####LIKE#####~~*#ILIKE###### ##NOT LIKE###NOT ILIKE###!~~###!~~*######
#########PostgreSQL####### #######LIKE##########################EXPLAIN#
##############

LIKE#ILIKE#NOT LIKE#NOT ILIKE#####PostgreSQL################ ###### ### ANY(######)#############ESCAPE################ ########################### #######

##################################^@#######starts_with#######

9.7.2. SIMILAR TO####

string SIMILAR TO *pattern* [ESCAPE *escape-character*]
string NOT SIMILAR TO *pattern* [ESCAPE *escape-character*]

SIMILAR TO### ######SQL############ ###############LIKE######## SQL#######LIKE#######(POSIX)################### #######

LIKE####SIMILAR TO### ################### ###LIKE####SIMILAR TO###%###_########################### #####################POSIX######.*###.#######

LIKE###############SIMILAR TO########POSIX##############################

- |#######2###################
- *########0##############
- +########1##############
- ?########0#####1############
- {*m*}##########*m*############
- {*m*,}########*m*##############
- {*m*,*n*}########*m*#####*n*##############
- ###()#####1####################
- ####[...]##POSIX##################

SIMILAR TO#######.####################

LIKE################################ ###########ESCAPE##############ESCAPE ''############################

SQL#######ESCAPE###ESCAPE### ###### ########PostgreSQL###############

##################################POSIX#################################### ## #### 9.20## 9.21## 9.22#########

#######

```
'abc' SIMILAR TO 'abc'          true
'abc' SIMILAR TO 'a'            false
'abc' SIMILAR TO '%(b|d)%'      true
'abc' SIMILAR TO '(b|c)%'       false
'-abc-' SIMILAR TO '%\mabc\M%'  true
'xabcy' SIMILAR TO '%\mabc\M%'  false
```

3#########substring#######SQL############################## ##SQL######## ####################

substring(*string* similar *pattern* escape *escape-character*)

##########SQL:1999#####################

substring(*string* from *pattern* for *escape-character*)

#######3#####################

substring(*string*, *pattern*, *escape-character*)

SIMILAR TO##NULL###### ######## ##"#######2################ #### ##

#######################substring#######3################ ####3############## #####|#################### ######################################3########### ##################################### #POSIX#######3###########non-greedy##### #####

SQL#########PostgreSQL################################3################# ### ################################3#################

###"##################

```
substring('foobar' similar '%#"o_b#"%' escape '#')   oob
substring('foobar' similar '#"o_b#"%' escape '#')    NULL
```

9.7.3. POSIX####

9.16##POSIX##############################

#9.16

###	## #
text ~ text → boolean	######################## 'thomas' ~ 't.*ma' → t
text ~* text → boolean	######################## 'thomas' ~* 'T.*ma' → t
text !~ text → boolean	########################## 'thomas' !~ 't.*max' → t
text !~* text → boolean	########################## 'thomas' !~* 'T.*ma' → f

POSIX#####################LIKE###SIMILAR TO############# egrep#sed#### #awk######Unix#################################

LIKE###

####LIKE######################## LIKE######################################
#############################

#######

```
'abcd' ~ 'bc'     true
'abcd' ~ 'a.c'    true — dot matches any character
'abcd' ~ 'a.*d'   true — * repeats the preceding pattern item
'abcd' ~ '(b|x)'  true — | means OR, parentheses group
'abcd' ~ '^a'     true — ^ anchors to start of string
'abcd' ~ '^(b|c)' false — would match except for anchoring
```

POSIX#######################

2#########substring###substring(*string* from *pattern*)######POSIX#######
######################## ####################NULL############################
#########
################################
################

#######

```
substring('foobar' from 'o.b')     oob
substring('foobar' from 'o(.)b')   o
```

regexp_count####POSIX############################ ####
#regexp_count(*string*,*pattern*[,*start*[,*flags*]])########## *pattern#string*####
##################*start*#################################### *flags*####
#############################0################ #####*flags*#i##################
########### ############# 9.24#########

#######

```
regexp_count('ABCABCAXYaxy', 'A.')          3
regexp_count('ABCABCAXYaxy', 'A.', 1, 'i')  4
```

regexp_instr##########POSIX########*N*######################## ########0####
####regexp_instr(*string*, *pattern* [, *start* [, *N* [, *endoption* [,
flags [, *subexpr*]]]]])###### *pattern#string*##########################*start#*
#################################### *N*################*N*############ #######
############## *endoption*#############0############################## #######
##*endoption*#1########################### *flags*################0############
################ ############# 9.24######### ####################*subexpr*#####
######### ########################### ###################### *subexpr*#########0
##################################

#######

```
regexp_instr('number of your street, town zip, FR', '[^,]+', 1, 2)
                                    23
regexp_instr('ABCDEFGHI', '(c..)(...)', 1, 1, 0, 'i', 2)
                                    6
```

regexp_like####POSIX################################true###false###### ##
#regexp_like#*string*,*pattern*[,*flags*]#### *flags*################0############
################ ############# 9.24######### ####################~############
i###############~*#############

#######

```
regexp_like('Hello World', 'world')       false
regexp_like('Hello World', 'world', 'i')  true
```

regexp_match###POSIX##
regexp_match(*string*, *pattern* [, *flags*])######## ###############NULL#### ## ############*pattern*###
############*pattern*#################################*n*######*pattern*#*n*## ## *flags*########
9.24##########

#######

```
SELECT regexp_match('foobarbequebaz', 'bar.*que');
 regexp_match
--------------
 {barbeque}
(1 row)

SELECT regexp_match('foobarbequebaz', '(bar)(beque)');
 regexp_match
--------------
 {bar,beque}
(1 row)
```

###

#################NULL############################regexp_substr()#### ###### ####regexp_substr()#PostgreSQL#####15########### ###############
#####regexp_match()####################

```
SELECT (regexp_match('foobarbequebaz', 'bar.*que'))[1];
 regexp_match
--------------
 barbeque
(1 row)
```

regexp_matches###POSIX## ##
#regexp_match###### ####################################g##############1## ##########*N*####g#############*N*####### ########regexp_match################
#############*pattern*##################################
regexp_matches## 9.24#################################g#######

#######

```
SELECT regexp_matches('foo', 'not there');
 regexp_matches
----------------
(0 rows)

SELECT regexp_matches('foobarbequebazilbarfbonk', '(b[^b]+)(b[^b]+)',
 'g');
 regexp_matches
----------------
 {bar,beque}
 {bazil,barf}
(2 rows)
```

###

##################regexp_match()#################regexp_matches()# #######g################ ####regexp_match()#PostgreSQL#####10######### ### ############################SELECT###regexp_matches()########### ###

```
SELECT col1, (SELECT regexp_matches(col2, '(bar)(beque)')) FROM tab;
```

###regexp_match()##################################NULL###### #SELECT ##

regexp_replace####POSIX################################ ## ##regexp_replace(*source*, *pattern*, *replacement*[, *start*[, *N*]][, *flags*])### #*start*#########*N*#########*flags*#################### *pattern*###### #####*source*############# ###############*replacement*#######*source*######## ## *replacement*####*n*#*n*#1##9################*n*################################ ##### ###\&################################## ################################\ \####### ##*string*#####*pattern*#############*start*########################### ################################## *N*#########################*N*################## ## g###########*N*#############*start*########################## #g####*N*######### ######### *flags*###########################1######################## #######(g## #)# 9.24#########

#######

```
regexp_replace('foobarbaz', 'b..', 'X')
                                   fooXbaz
regexp_replace('foobarbaz', 'b..', 'X', 'g')
                                   fooXX
regexp_replace('foobarbaz', 'b(..)', 'X\1Y', 'g')
                                   fooXarYXazY
regexp_replace('A PostgreSQL function', 'a|e|i|o|u', 'X', 1, 0, 'i')
                                   X PXstgrXSQL fXnctXXn
regexp_replace('A PostgreSQL function', 'a|e|i|o|u', 'X', 1, 3, 'i')
                                   A PostgrXSQL function
```

regexp_split_to_table###POSIX########################### ##regexp_split_to_table(*string*, *pattern* [, *flags*])######## *pattern*####### #####*string*###### ## ############ ## *flags*################## ################################## regexp_split_to_table## 9.24############## #####

regexp_split_to_array####regexp_split_to_array#####text######### ##regexp_split_to_table######### regexp_split_to_array(*string*, *pattern* [, *flags*])######## ######regexp_split_to_table########

#######

```
SELECT foo FROM regexp_split_to_table('the quick brown fox jumps over the
 lazy dog', '\s+') AS foo;
  foo
-------
 the
 quick
 brown
 fox
```

```
 jumps
 over
 the
 lazy
 dog
(9 rows)

SELECT regexp_split_to_array('the quick brown fox jumps over the lazy
 dog', '\s+');
              regexp_split_to_array
-----------------------------------------------
 {the,quick,brown,fox,jumps,over,the,lazy,dog}
(1 row)

SELECT foo FROM regexp_split_to_table('the quick brown fox', '\s*') AS
 foo;
 foo
-----
 t
 h
 e
 q
 u
 i
 c
 k
 b
 r
 o
 w
 n
 f
 o
 x
(16 rows)
```

#################regexp## ####
##########regexp#################################### Perl#####################
#########

regexp_substr####POSIX######################## ########NULL######
regexp_substr*string*,*pattern*[,*start*[,*N*[,*flags*[,*subexpr*]]]])###########
pattern#*string*##########################*start*#############################
######### *N*################*N*############### ########################### *flags*#
###############0############################## ############## 9.24########## #####
################*subexpr*################ ######################## ############
############ *subexpr*#########0################################

#######

```
regexp_substr('number of your street, town zip, FR', '[^,]+', 1, 2)
                                    town zip
regexp_substr('ABCDEFGHI', '(c..)(...)', 1, 1, 'i', 2)
                                   FGH
```

9.7.3.1.

PostgreSQL######Henry Spencer############################## ###################
####################

POSIX 1003.2#############RE###2##############RE####ERE########egrep##########
#####RE####BRE########ed########### PostgreSQL################POSIX#########

#Perl#Tcl### ######POSIX#######RE####RE####ARE######ARE#ERE#############BRE###

> **##**
>
> PostgreSQL###########ARE############ ####RE########9.7.3.4###ERE#####BRE############## ####POSIX1003.2###

#####|#######1#################### #########1####################

############################ ##

############################## ################################## ## ###### 9.17#################################### # 9.18###

############################ ### ####### 9.19#########################

#9.17

###	##
(*re*)	####*re* ###########*re*############## ###################
(?:*re*)	##ARE###
.	###1##########
[*chars*]	###### *chars*#####1######## ####9.7.3.2# ##########
k	####*k*###\\##############
c	###*c*###### ################## ######## 9.7.3.3######### #ARE###ERE#BRE#####*c*# ########
{	##################{######## ######### ##*bound*###########
x	###*x*##########1##### *x*########

RE#########\###############

> **##**
>
> ##standard_conforming_strings######off#################################2###4.1.2.1##########

#9.18

####	###
*	####0############

####	###
+	####1###########
?	####0####1#######
{m}	#######m########
{m,}	####m#########
{m,n}	####m###n######### m#n#############
*?	*#############
+?	+#############
??	?#############
{m}?	{m}#############
{m,}?	{m,}#############
{m,n}?	{m,n}#############

{...}###################### ######m#n#########10#######0##255##############

############ARE## ## #9.7.3.5#########

> ##
>
> ############################**###### ########################^#|#########
> ######

#9.19 ######

##	##
^	#########
$	#########
(?=re)	#######re########################## #ARE####
(?!re)	#######re########################## #ARE####
(?<=re)	######re########################## #ARE####
(?<!re)	#####re########################## #ARE####

#####################9.7.3.3##

9.7.3.2.

########[]########## ###############1########################## ####^######### #############1######### #####2###-#############2############################ ###ASCII####[0-9]############# ###a-c-e###########2######### ############### ####################################

######]############################^########## -########################### #####2############ -################[.#.]#################### #######[#### ########################ARE## ###\#ERE# BRE#####################ARE#########################

##[.#.]##############
########## ############################## ##################################1#
##########################ch#############[[.ch.]]*c#chchcc########5#######
###

> **##**
>
> ######PostgreSQL################## ########################

########[=#=]## ##############
##[.#.]################ ####[[=o=]]#[[=^=]]###[o^]##########o#^##########
##################

##########[:#:]###################################### ######################
POSIX#################### alnum#######alpha#####blank#######cntrl####
###digit#####graph#############lower######print#############punct###
###space#####upper######xdigit#16###### ################7-bit ASCII##########
################## ######ASCII###23.2######
##########################LC_CTYPE#####23.1######### #ASCII#################
################################ ####C###########ASCII######################
############PostgreSQL##alnum###########(_)######word########7-bit ASCII
########ascii##############

########2#############[[:<:]]#[[:>:]]#####################################
######word################################
######## ####POSIX 1003.2###
######## ##

9.7.3.3.

########\###################### ##
\##############################ARE####### ERE#########################\######
#####################################\######### ####ERE#ARE#######1#####

#################RE############################ ##### 9.20######

#################################### ##### 9.21######

9.22######

#####*n*#########################*n*################ 9.23########## ##
##([bc])\1#bb####cc########bc#cb#########RE###################### #######
########### ################## ### ###
##(^\d)\1#22#######

#9.20

#####	##
\a	C############
\b	C###########
\B	################################\#####
\c*X*	####*X*###########5####*X*###########0## ###
\e	#####ESC##############033###8####### ##

#####	##
\f	C########
\n	C#######
\r	C#######
\t	C########
\u*wxyz*	####*wxyz*####4##16######16##### #0x*wxyz*#####
\U*stuvwxyz*	####*stuvwxyz*####8##16######16##### #0x*stuvwxyz*#####
\v	C########
\x*hhh*	####*hhh*####16###########16####0x*hhh*# #########16################
\0	####0#NULL########
xy	####*xy*####2##8################0*xy*#### #
xyz	####*xyz*####3##8################0*xyz*# ####

16######0-9#a-f#A-F### 8######0-7###

ASCII###(0-127)## #########UTF-8## ###########################\u1234###U+1234####### ######################### ########################### ## ##################

########################### ####\135#ASCII#]####\135#################

#9.21

#####	##
\d	[[:digit:]]#################
\s	[[:space:]]###################
\w	[[:word:]]###################
\D	[^[:digit:]]##################
\S	[^[:space:]]####################
\W	[^[:word:]]####################

####[a-c\d]#[a-c[:digit:]]######

#9.22

#####	##
\A	################^#########9.7.3.5####### ####
\m	###############
\M	###############
\y	####################
\Y	#########################

#####	##
\Z	###############$########9.7.3.5###### ####

######[[:<:]]#[[:>:]]##################################

#9.23

#####	##
\m	####m#############m########
\mnn	####m#########nn############mnn10##### ########################mnn########

> ##
>
> 8## ###0###8 ############ ################################### ######################## ##8###########

9.7.3.4.

################################

RE##2######################## RE#***:###########RE####ARE######## (PostgreSQL# RE#ARE####################################flags###########ERE#BRE########### ##) RE#***=###########RE###################################

ARE##################(?xyz)########RE####################xyz#1########## ## ############################— ###########################flags################# ########### ############## 9.24###### #####################flags################# ######

#9.24 ARE###########

#####	##
b	###RE#BRE
c	## ########
e	###RE#ERE
i	################9.7.3.5################# #####################
m	n########
n	###########9.7.3.5####
p	###############9.7.3.5####
q	###RE##############################
s	##################
t	################
w	######################9.7.3.5####
x	########

#################)######### ARE######***:####################

####################RE######x############################## #######RE#######
################RE################# ##############RE############### ########3
#########

- ###\####################
- ########################
- (?:###########################

#####################################

####ARE############(?#*ttt*)##################*ttt*################# ########
#####(?:######################### ##
############

###***=################RE####################################

9.7.3.5.

RE#######1######################RE######################## ########1#########
##
#

RE#########################

- ##
- RE#######################
- {*m*}####{*m*}?###
- ##########{*m*,*n*}#*m*#*n*##
- ############{*m*,*n*}?#*m*#*n*##
- #######|########RE##
- |########2############RE##########

###RE############### #########RE########
################################### ##
###########RE################################

############

```
SELECT SUBSTRING('XY1234Z', 'Y*([0-9]{1,3})');
Result: 123
SELECT SUBSTRING('XY1234Z', 'Y*?([0-9]{1,3})');
Result: 1
```

#######Y*###########RE############# ####Y################################
#Y123###### ################123###### 2#######Y*?##########RE############## ###
#Y##################################Y1###### ###[0-9]{1,3}##################
#############################1#############

######RE##################################RE################################
##

{1,1}###{1,1}?###########RE#################################### RE##########
#################################### ##
################

```
SELECT regexp_match('abc01234xyz', '(.*)(\d+)(.*)');
Result: {abc0123,4,xyz}
```

###############.*####################\d+#################### ###############
#########

```
SELECT regexp_match('abc01234xyz', '(.*?)(\d+)(.*)');
Result: {abc,0,""}
```

###################RE## RE#######
##################

```
SELECT regexp_match('abc01234xyz', '(?:(.*?)(\d+)(.*)){1,1}');
Result: {abc,01234,xyz}
```

RE##

################################
####bb*#abbbc####3#######(week|wee)(night|knights)#weeknights####10#
#######abc####(.*).*##################3###########bc####(a*)*###########
RE#####################

##############
################################x#[xX]################################ #######
###########[x]#[xX]######[^x]#[^xX]####################################

#################.#^############RE##
##########^#$## ####ARE#####
#\A#\Z####################### ##########\D#\W##################
#PostgreSQL 14################################ #########[^[:digit:]]###
#[^[:word:]]#########

#####################.############################^#$########

#######################^#$######################.################ #############
##################

9.7.3.6.

#####RE################ ###################256#######RE##############POSIX####
#######RE##############

ARE#######POSIX ERE##############\############################ #####ARE#
###POSIX ERE################################***#####BRE###ERE#POSIX#######
##

###ARE###Perl########## #############Perl############### ###########\b#\B####
###/#
################################

9.7.3.7.

BRE#ERE################ BRE######|#+#?########################### #######
#\{#\}####{#}########## ##############\(#\)####(#)########## ^##RE########
######################## $##RE################################ ###*#RE########
##############################^############### ####1##################BRE###
###\<#\>#####[[:<:]]#[[:>:]]######

9.7.3.8. ##SQL###XQuery####

SQL:2008#####SQL##########XQuery##################################

- `LIKE_REGEX`
- `OCCURRENCES_REGEX`
- `POSITION_REGEX`
- `SUBSTRING_REGEX`
- `TRANSLATE_REGEX`

PostgreSQL###################### # 9.25###

#9.25

##SQL	PostgreSQL
string `LIKE_REGEX` *pattern*	`regexp_like(`*string*`,` *pattern*`)` or *string* `~` *pattern*
`OCCURRENCES_REGEX(`*pattern* `IN` *string*`)`	`regexp_count(`*string*`,` *pattern*`)`
`POSITION_REGEX(`*pattern* `IN` *string*`)`	`regexp_instr(`*string*`,` *pattern*`)`
`SUBSTRING_REGEX(`*pattern* `IN` *string*`)`	`regexp_substr(`*string*`,` *pattern*`)`
`TRANSLATE_REGEX(`*pattern* `IN` *string* `WITH` *replacement*`)`	`regexp_replace(`*string*`,` *pattern*`,` *replacement*`)`

PostgreSQL##########################SQL#############SQL################## ##################################

##SQL#############ARE#######XQuery############ ###POSIX##########XQuery#####################

- XQuery################### ##########`[a-z-[aeiou]]`#######################################
- XQuery########`\c#\C#\i#\I`############
- `\p{UnicodeProperty}`#########`\P{UnicodeProperty}`####XQuery##########################
- POSIX##################`COLLATE`##############`\w`## 9.21################## XQuery########Unicode####################### ####Unicode#####################################
- SQL###XQuery#######POSIX##########newline################ ################ ####ASCII NL#`\n`############## ###SQL#CR #`\r`##CRLF #`\r\n`##Windows###### ###LINE SEPARATOR (U+2028)####Unicode###################### #####SQL#### ##.#`\s`#`\r\n`#2######1###############
- # 9.20#################XQuery#`\n`#`\r`#`\t`############
- XQuery################`[:`*name*`:]`##############

- XQuery########################## 9.22####################
- 9.7.3.4#############XQuery#########
- XQuery################POSIX## 9.24############################## i#q########### ################
  - XQuery#s#######################m#^#$#######################POSIX#n#p#w### #############POSIX#s#m############### ###############POSIX############ ##XQuery##################
  - XQuery#x##################POSIX#################### POSIX#x############## ############## POSIX#######################

9.8.

PostgreSQL#################/## ################################## # 9.26############ ########################## #########2##############################

#9.26

##	
	##
	#
to_char (timestamp, text) → text	
to_char (timestamp with time zone, text) → text	
	##############################
	to_char(timestamp '2002-04-20 17:31:12.66', 'HH12:MI:SS') → 05:31:12
to_char (interval, text) → text	
	############################
	to_char(interval '15h 2m 12s', 'HH24:MI:SS') → 15:02:12
to_char (*numeric_type*, text) → text	
	##########################integer#bigint#numeric#real#double precision#######
	to_char(125, '999') → 125
	to_char(125.8::real, '999D9') → 125.8
	to_char(-125.8, '999D99S') → 125.80-
to_date (text, text) → date	
	##########################
	to_date('05 Dec 2000', 'DD Mon YYYY') → 2000-12-05
to_number (text, text) → numeric	
	##########################
	to_number('12,454.8-', '99G999D9S') → -12454.8
to_timestamp (text, text) → timestamp with time zone	
	############################## ## 9.33#to_timestamp(double precision)########
	to_timestamp('05 Dec 2000', 'DD Mon YYYY') → 2000-12-05 00:00:00-05

> ###
>
> to_timestamp#to_date################################### ########## ### ####to_number## ################

to_char### ############## ################### ## ##### ## #########

9.27####/##############################

#9.27 ##/################

####	##
HH	# (01–12)
HH12	# (01–12)
HH24	# (00–23)
MI	# (00–59)
SS	# (00–59)
MS	### (000–999)
US	##### (000000–999999)
FF1	10##1# (0–9)
FF2	100##1# (00–99)
FF3	### (000–999)
FF4	10##1### (0000–9999)
FF5	100##1### (00000–99999)
FF6	##### (000000–999999)
SSSS, SSSSS	##0###### (0–86399)
AM#am#PM###pm	##/############
A.M.#a.m.#P.M.###p.m.	##/############
Y,YYY	#######4####
YYYY	##4####
YYY	###3#
YY	###2#
Y	###1#
IYYY	ISO 8601#####4####
IYY	ISO 8601######3#
IY	ISO 8601######2#
I	ISO 8601######1#
BC#bc# AD####ad	##############
B.C.#b.c.# A.D.####a.d.	##############
MONTH	##########9################
Month	##############9################

####	##
month	##########9################
MON	###############3###################
Mon	###################3###################
mon	###############3###################
MM	###(01–12)
DAY	###########9################
Day	###############9################
day	###########9################
DY	###################3###################
Dy	###################3###################
dy	################3###################
DDD	####### (001–366)
IDDD	ISO 8601##########001–371### ###### ISO######
DD	####### (01–31)
D	########1#######7###
ID	ISO 8601#########1#######7###
W	###### (1–5)############1##
WW	######## (1–53)########1##
IW	ISO 8601###############01–53;######### #####1##
CC	###2##21###2001-01-01#####
J	########4714#11#24################### ###B.7###
Q	###
RM	############I–XII#I#1##
rm	############i–xii#i#1##
TZ	upper case time-zone abbreviation
tz	lower case time-zone abbreviation
TZH	time-zone###
TZM	time-zone##
OF	time-zone offset from UTC (*HH* or *HH*:*MM*)

####FMMonth#FM#######Month##### ## # 9.28####/##################

#9.28 ##/###################

###	##	#
FM###	##########0############## ###	FMMonth
TH###	###########	DDTH####12TH

###	##	#
th###	##########	DDth####12th
FX###	######################### ###	FX Month DD Day
TM###	######lc_time############ ############	TMMonth
SP###	##########	DDSP

##/#####################

- FM###################################### PostgreSQL###FM################## ####Oracle###FM##################FM#################################

- FM################TM############

- to_timestamp#to_date################### ###MON#Mon#mon################ ## TM#######################################23.2####

- FX################to_timestamp#to_date################################ ###### ####to_timestamp('2000 JUN', 'YYYY MON')#to_timestamp('2000 - JUN', 'YYYY-MON')#######to_timestamp('2000 JUN','FXYYYY MON')# ######## ###to_timestamp######################## FX########1############ ####

- FX################to_timestamp#to_date################################ #"non-letter/non-digit character"## # ####to_timestamp('2000JUN', 'YYYY///MON')#to_timestamp('2000/JUN', 'YYYY MON')#######to_timestamp('2000//JUN', 'YYYY/MON')############ ##################################

  FX## ######################### ######################## #####to_timestamp('2000/JUN', 'FXYYYY MON')### #####to_timestamp('2000/JUN', 'FXYYYY MON')######################### ##J################

- TZH######################## FX## ## ################ ##########TZH## ######TZH############ ############################ ### ##to_timestamp('2000 -10', 'YYYY TZH')##-10#TZH###### ##to_timestamp('2000 -10', 'YYYY TZH')##10#TZH#######

- to_char##################################### ########################### ## ####'"Hello Year "YYYY'##YYYY## ###############Year##Y######## to_date#to_number#to_timestamp######## ###########################"XX"#2##########XX#####################

###

PostgreSQL 12################"non-letter or non-digit"################ ################ #####to_timestamp('2000y6m1d', 'yyyy-MM-DD')####### ######################"letter characters"############# ### ##to_timestamp('2000y6m1d', 'yyyytMMtDDt')#to_timestamp('2000y6m1d', 'yyyy"y"MM"m"DD"d"')##y#m#d########

- ###############'\"YYYY Month\"'########################### ############## ## ##

- to_timestamp####to_date#YYY###4####################4##########2020##### ###############95####1995######

- to_timestamp###to_date#######BC##### #######BC################AD#### ## #########BC 1########

- to_timestamp#to_date###4############YYYY########## YYYY############## ################ ##########4######### ####20000#####to_date('200001130', 'YYYYMMDD')#4############   #####to_date('20000-1130', 'YYYY-MMDD')#to_date('20000Nov30', 'YYYYMonDD')#########################

- to_timestamp###to_date####YYY#YYYY#####Y,YYY##########CC############## ########### CC#YY####Y################################ ################### ################

- to_timestamp###to_date#############DAY#D################################ ################ ########Q###############

- to_timestamp###to_date##### ISO 8601#################################### ######

  - ############# ####to_date('2006-42-4', 'IYYY-IW-ID')####2006-10-19# ##### ############1#############

  - #############to_date('2006-291', 'IYYY-IDDD')#2006-10-19######

  ISO 8601## ISO 8601####### ############################## #############ISO#############

  ##

  to_date######ISO####################to_char##########YYYY-MM-DD (IYYY-IDDD)####################### ####IYYY-MM-DD################## ################### (########9.9.1##########)

- to_timestamp########MS##########US######################### ## ##to_timestamp('12.3', 'SS.MS')#3#######300###################12 + 0.3#### ####### ###SS.MS########12.3#12.30#12.300################### 3########## #12:003#########################12 + 0.003 = 12.003########

  ############# to_timestamp('15:12:02.020.001230', 'HH24:MI:SS.MS.US')#15##12##2#+20###+1230##### = 2.021230####

- to_char(..., 'ID')#######extract(isodow from ...)######## ##to_char(..., 'D')#######extract(dow from ...)##############

- to_char(interval)####HH#HH12#12################### ###0###36#######12#### ##### ##HH24##############interval######23###########

9.29#############################

#9.29

####	##
9	###################

####	##
0	##################
.######	###
, (###)	########
PR	########
S	##############
L	############
D	###########
G	#################
MI	### < 0###############
PL	### > 0##############
SG	##########/######
RN	########1#3999#
TH###th	#####
V	################
EEEE	##########

#######################

- 0#################################### 9#################################### ############################ #to_number()######2##################
- ######################################to_char()######################
- ######S#L#D#G############################3###########lc_monetary## #lc_numeric##### ##3##############
- to_char()## S## ####9#######################
- SG#PL####MI#################### ####to_char(-12, 'MI9999')#'- 12'#### ##to_char(-12, 'S9999')#' -12'###### #Oracle#####9###MI#########9###MI# ###############
- TH################
- PL#SG####TH#PostgreSQL######
- to_number#####L####TH##10######## ### ####TH#2######### #######
- V#to_char#########10^*n*##### ###*n*#V####### V#to_number################## to_char###to_number#####V#################99.9V99 ########
- EEEE(######)## ###(####9.99EEEE##########)#

####FM99.99#FM#######99.99####### # 9.30####################

#9.30

###	##	#
FM###	#########0##############	FM99.99

###	##	#
TH###	###########	999TH
th###	###########	999th

9.31##to_char#################

#9.31 to_char##

#	##
to_char(current_timestamp, 'Day, DD HH12:MI:SS')	'Tuesday , 06 05:39:18'
to_char(current_timestamp, 'FMDay, FMDD HH12:MI:SS')	'Tuesday, 6 05:39:18'
to_char(current_timestamp AT TIME ZONE 'UTC', 'YYYY-MM-DD"T"HH24:MI:SS"Z"')	'2022-12-06T05:39:18Z', ISO 8601 extended format
to_char(-0.1, '99.99')	' -.10'
to_char(-0.1, 'FM9.99')	'-.1'
to_char(-0.1, 'FM90.99')	'-0.1'
to_char(0.1, '0.9')	' 0.1'
to_char(12, '9990999.9')	' 0012.0'
to_char(12, 'FM9990999.9')	'0012.'
to_char(485, '999')	' 485'
to_char(-485, '999')	'-485'
to_char(485, '9 9 9')	' 4 8 5'
to_char(1485, '9,999')	' 1,485'
to_char(1485, '9G999')	' 1 485'
to_char(148.5, '999.999')	' 148.500'
to_char(148.5, 'FM999.999')	'148.5'
to_char(148.5, 'FM999.990')	'148.500'
to_char(148.5, '999D999')	' 148,500'
to_char(3148.5, '9G999D999')	' 3 148,500'
to_char(-485, '999S')	'485-'
to_char(-485, '999MI')	'485-'
to_char(485, '999MI')	'485 '
to_char(485, 'FM999MI')	'485'
to_char(485, 'PL999')	'+485'
to_char(485, 'SG999')	'+485'
to_char(-485, 'SG999')	'-485'
to_char(-485, '9SG99')	'4-85'
to_char(-485, '999PR')	'<485>'
to_char(485, 'L999')	'DM 485'
to_char(485, 'RN')	' CDLXXXV'
to_char(485, 'FMRN')	'CDLXXXV'
to_char(5.2, 'FMRN')	'V'

#	##
to_char(482, '999th')	' 482nd'
to_char(485, '"Good number:"999')	'Good number: 485'
to_char(485.8, '"Pre:"999" Post:" .999')	'Pre: 485 Post: .800'
to_char(12, '99V999')	' 12000'
to_char(12.4, '99V999')	' 12400'
to_char(12.45, '99V9')	' 125'
to_char(0.0004859, '9.99EEEE')	' 4.86e-04'

9.9. ##/########

9.33####/## # 9.32###+#*################## ###### ##########9.8######### 8.5#######/#################################

9.1#############/########## ## ## ## ##TimeZone#######################################UTC##########UTC####### ## ######################TimeZone##################

###time####timestamp#######################2#########1##time with time zone####timestamp with time zone#########1##time without time zone#### #timestamp without time zone######## ########################## ###+#*### ####2#########date + integer#integer + date####################

#9.32 ##/#####

###	## #
date + integer → date	####### date '2001-09-28' + 7 → 2001-10-05
date + interval → timestamp	######### date '2001-09-28' + interval '1 hour' → 2001-09-28 01:00:00
date + time → timestamp	####### date '2001-09-28' + time '03:00' → 2001-09-28 03:00:00
interval + interval → interval	###### interval '1 day' + interval '1 hour' → 1 day 01:00:00
timestamp + interval → timestamp	############## timestamp '2001-09-28 01:00' + interval '23 hours' → 2001-09-29 00:00:00
time + interval → time	#########

```
###
      ##
      #
```

```
      time '01:00' + interval '3 hours' → 04:00:00
```

```
- interval → interval
      ##########
      - interval '23 hours' → -23:00:00
```

```
date - date → integer
      #############
      date '2001-10-01' - date '2001-09-28' → 3
```

```
date - integer → date
      ########
      date '2001-10-01' - 7 → 2001-09-24
```

```
date - interval → timestamp
      ###########
      date '2001-09-28' - interval '1 hour' → 2001-09-27 23:00:00
```

```
time - time → interval
      #####
      time '05:00' - time '03:00' → 02:00:00
```

```
time - interval → time
      ###########
      time '05:00' - interval '2 hours' → 03:00:00
```

```
timestamp - interval → timestamp
      ###############
      timestamp '2001-09-28 23:00' - interval '23 hours' →
      2001-09-28 00:00:00
```

```
interval - interval → interval
      #######
      interval '1 day' - interval '1 hour' → 1 day -01:00:00
```

```
timestamp - timestamp → interval
      ##########justify_hours()####24###########
      timestamp '2001-09-29 03:00' - timestamp '2001-07-27 12:00'
      → 63 days 15:00:00
```

```
interval * double precision → interval
      ############
      interval '1 second' * 900 → 00:15:00
      interval '1 day' * 21 → 21 days
      interval '1 hour' * 3.5 → 03:30:00
```

```
interval / double precision → interval
      ############
      interval '1 hour' / 1.5 → 00:40:00
```

#9.33 ##/#######

age (timestamp, timestamp) → interval ################################## age(timestamp '2001-04-10', timestamp '1957-06-13') → 43 years 9 mons 27 days
age (timestamp) → interval current_date############## age(timestamp '1957-06-13') → 62 years 6 mons 10 days
clock_timestamp ( ) → timestamp with time zone ####################9.9.5#### clock_timestamp() → 2019-12-23 14:39:53.662522-05
current_date → date ######9.9.5### current_date → 2019-12-23
current_time → time with time zone ######9.9.5#### current_time → 14:39:53.662522-05
current_time (integer) → time with time zone #############9.9.5#### current_time(2) → 14:39:53.66-05
current_timestamp → timestamp with time zone ##########################9.9.5#### current_timestamp → 2019-12-23 14:39:53.662522-05
current_timestamp (integer) → timestamp with time zone ##################################9.9.5#### current_timestamp(0) → 2019-12-23 14:39:53-05
date_add (timestamp with time zone, interval [, text]) → timestamp with time zone interval#timestamp with timezone#####3########################## ###########TimeZone#################### 2###########timestamp with timezone+interval######## date_add('2021-10-31 00:00:00+02'::timestamptz, '1 day'::interval, 'Europe/Warsaw') → 2021-10-31 23:00:00+00
date_bin (interval, timestamp, timestamp) → timestamp ##########################9.9.3######### date_bin('15 minutes', timestamp '2001-02-16 20:38:40', timestamp '2001-02-16 20:05:00') → 2001-02-16 20:35:00
date_part (text, timestamp) → double precision ##################extract#####9.9.1#### date_part('hour', timestamp '2001-02-16 20:38:40') → 20
date_part (text, interval) → double precision ################extract#####9.9.1####

date_part('month', interval '2 years 3 months') → 3
date_subtract(timestamp with time zone, interval [, text]) → timestamp with time zone interval#timestamp with timezone#####3##################### #############TimeZone#################### 2##########timestamp with timezone-interval######## date_subtract('2021-11-01 00:00:00+01'::timestamptz, '1 day'::interval, 'Europe/Warsaw') → 2021-10-30 22:00:00+00
date_trunc(text, timestamp) → timestamp ############9.9.2### date_trunc('hour', timestamp '2001-02-16 20:38:40') → 2001-02-16 20:00:00
date_trunc(text, timestamp with time zone, text) → timestamp with time zone #########################9.9.2## date_trunc('day', timestamptz '2001-02-16 20:38:40+00', 'Australia/Sydney') → 2001-02-16 13:00:00+00
date_trunc(text, interval) → interval ############9.9.2### date_trunc('hour', interval '2 days 3 hours 40 minutes') → 2 days 03:00:00
extract(*field* from timestamp) → numeric ##################9.9.1#### extract(hour from timestamp '2001-02-16 20:38:40') → 20
extract(*field* from interval) → numeric ###############9.9.1#### extract(month from interval '2 years 3 months') → 3
isfinite(date) → boolean ######+/-############ isfinite(date '2001-02-16') → true
isfinite(timestamp) → boolean ##########+/-############ isfinite(timestamp 'infinity') → false
isfinite(interval) → boolean #####[64.285714]#######+/-############ ###############±######## isfinite(interval '4 hours') → true
justify_days(interval) → interval 30################## justify_days(interval '1 year 65 days') → 1 year 2 mons 5 days
justify_hours(interval) → interval #######24#######

justify_hours(interval '50 hours 10 minutes') → 2 days 02:10:00
justify_interval(interval) → interval justify_days###justify_hours######################### justify_interval(interval '1 mon -1 hour') → 29 days 23:00:00
localtime → time ######9.9.5#### localtime → 14:39:53.662522
localtime (integer) → time ############9.9.5#### localtime(0) → 14:39:53
localtimestamp → timestamp #########################9.9.5#### localtimestamp → 2019-12-23 14:39:53.662522
localtimestamp (integer) → timestamp ################################9.9.5#### localtimestamp(2) → 2019-12-23 14:39:53.66
make_date (*year* int, *month* int, *day* int) → date ####################BC####### make_date(2013, 7, 15) → 2013-07-15
make_interval ([*years* int [, *months* int [, *weeks* int [, *days* int [, *hours* int [, *mins* int [, *secs* double precision]]]]]]]) → interval ### make_interval(days => 10) → 10 days
make_time (*hour* int, *min* int, *sec* double precision) → time ################ make_time(8, 15, 23.5) → 08:15:23.5
make_timestamp (*year* int, *month* int, *day* int, *hour* int, *min* int, *sec* double precision) → timestamp ##########################BC####### make_timestamp(2013, 7, 15, 8, 15, 23.5) → 2013-07-15 08:15:23.5
make_timestamptz (*year* int, *month* int, *day* int, *hour* int, *min* int, *sec* double precision [, *timezone* text]) → timestamp with time zone ################################BC####### *timezone*################ ####### ############Europe/London####### make_timestamptz(2013, 7, 15, 8, 15, 23.5) → 2013-07-15 08:15:23.5+01 make_timestamptz(2013, 7, 15, 8, 15, 23.5, 'America/New_York') → 2013-07-15 13:15:23.5+01
now() → timestamp with time zone

###########################9.9.5#### now() → 2019-12-23 14:39:53.662522-05
statement_timestamp() → timestamp with time zone ###################9.9.5#### statement_timestamp() → 2019-12-23 14:39:53.662522-05
timeofday() → text ########clock_timestamp######text##########9.9.5#### timeofday() → Mon Dec 23 14:39:53.662522 2019 EST
transaction_timestamp() → timestamp with time zone ###########################9.9.5#### transaction_timestamp() → 2019-12-23 14:39:53.662522-05
to_timestamp(double precision) → timestamp with time zone Unix#######1970-01-01 00:00:00+00########timestamp with time zone### to_timestamp(1284352323) → 2010-09-13 04:32:03+00

##########OVERLAPS SQL###############

```
(start1, end1) OVERLAPS (start2, end2)
(start1, length1) OVERLAPS (start2, length2)
```

#####2### ###2#####################
####/##/##########################   ##################################
##OVERLAPS######################### #######start <= time < end################
####start#end#################### ################2#######################

```
SELECT (DATE '2001-02-16', DATE '2001-12-21') OVERLAPS
       (DATE '2001-10-30', DATE '2002-10-30');
Result: true
SELECT (DATE '2001-02-16', INTERVAL '100 days') OVERLAPS
       (DATE '2001-10-30', DATE '2002-10-30');
Result: false
SELECT (DATE '2001-10-29', DATE '2001-10-30') OVERLAPS
       (DATE '2001-10-30', DATE '2001-10-31');
Result: false
SELECT (DATE '2001-10-30', DATE '2001-10-30') OVERLAPS
       (DATE '2001-10-30', DATE '2001-10-31');
Result: true
```

timestamp###timestamp with time zone###interval#############interval###
######interval########################## #####################timestamp######
##
####3#31##1#######4#30#######3#31##2#######5#31####### ###########timestamp###
################# ######################## ##################################
DST###########timestamp with time zone################interval '1
day'###############interval '24 hours'########################### #######
America/Denver####################

```
SELECT timestamp with time zone '2005-04-02 12:00:00-07' + interval '1
 day';
Result: 2005-04-03 12:00:00-06
```

```
SELECT timestamp with time zone '2005-04-02 12:00:00-07' + interval '24
 hours';
Result: 2005-04-03 13:00:00-06
```

#####America/Denver####2005-04-03 02:00:00##########1#############

###############age#####months################# PostgreSQL################## ################## ####age('2004-06-01', '2004-04-30')#4######1 mon 1 day#####5##31####4##30######5##########1 mon 2 days#######

############################ ##########################EXTRACT(EPOCH FROM ...)##########################2############# ########################## ######### #-#########################(24###)#####/#/############## age##########/ #/############################## ######################### #####timezone = 'US/ Eastern'########2##################

```
SELECT EXTRACT(EPOCH FROM timestamptz '2013-07-01 12:00:00') -
       EXTRACT(EPOCH FROM timestamptz '2013-03-01 12:00:00');
Result: 10537200.000000
SELECT (EXTRACT(EPOCH FROM timestamptz '2013-07-01 12:00:00') -
        EXTRACT(EPOCH FROM timestamptz '2013-03-01 12:00:00'))
        / 60 / 60 / 24;
Result: 121.9583333333333333
SELECT timestamptz '2013-07-01 12:00:00' - timestamptz '2013-03-01
 12:00:00';
Result: 121 days 23:00:00
SELECT age(timestamptz '2013-07-01 12:00:00', timestamptz '2013-03-01
 12:00:00');
Result: 4 mons
```

9.9.1. EXTRACT, date_part

EXTRACT(*field* FROM *source*)

extract#####/########################
source#timestamp#date#time#interval############## #timestamp#time######### ############# *field*#source################################## ################ ################### #####1###########date##########1#########time#########
extract###numeric########

###############

century

interval###########100#####

```
SELECT EXTRACT(CENTURY FROM TIMESTAMP '2000-12-16 12:21:13');
Result: 20
SELECT EXTRACT(CENTURY FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 21
SELECT EXTRACT(CENTURY FROM DATE '0001-01-01 AD');
Result: 1
SELECT EXTRACT(CENTURY FROM DATE '0001-12-31 BC');
Result: -1
SELECT EXTRACT(CENTURY FROM INTERVAL '2001 years');
Result: 20
```

day

#####1–31## interval######

```
SELECT EXTRACT(DAY FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 16
SELECT EXTRACT(DAY FROM INTERVAL '40 days 1 minute');
Result: 40
```

decade

#######10######

```
SELECT EXTRACT(DECADE FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 200
```

dow

####0#######6######

```
SELECT EXTRACT(DOW FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 5
```

extract########to_char(...,'D')####################

doy

#########1–365/366#

```
SELECT EXTRACT(DOY FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 47
```

epoch

timestamp with time zone########1970-01-01 00:00:00 UTC#####
##########date#timestamp############################1970-01-01 00:00:00###
###interval####################

```
SELECT EXTRACT(EPOCH FROM TIMESTAMP WITH TIME ZONE '2001-02-16
 20:38:40.12-08');
Result: 982384720.120000
SELECT EXTRACT(EPOCH FROM TIMESTAMP '2001-02-16 20:38:40.12');
Result: 982355920.120000
SELECT EXTRACT(EPOCH FROM INTERVAL '5 days 3 hours');
Result: 442800.000000
```

to_timestamp######timestamp with time zone############

```
SELECT to_timestamp(982384720.12);
Result: 2001-02-17 04:38:40.12+00
```

to_timestamp#date####timestamp###
##################UTC############################

hour

########TIMESTAMP####0–23#INTERVAL########

```
SELECT EXTRACT(HOUR FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 20
```

isodow

####1#######7######

```
SELECT EXTRACT(ISODOW FROM TIMESTAMP '2001-02-18 20:38:40');
Result: 7
```

######dow###### ###ISO 8601############

isoyear

#########ISO 8601#####

```
SELECT EXTRACT(ISOYEAR FROM DATE '2006-01-01');
Result: 2005
SELECT EXTRACT(ISOYEAR FROM DATE '2006-01-02');
Result: 2006
```

####ISO 8601#####1#4#####################1#######12####ISO################## ### #####week##############

julian

###################### ############################ ###B.7##########

```
SELECT EXTRACT(JULIAN FROM DATE '2006-01-01');
Result: 2453737
SELECT EXTRACT(JULIAN FROM TIMESTAMP '2006-01-01 12:00');
Result: 2453737.50000000000000000000
```

microseconds

###############1,000,000###################

```
SELECT EXTRACT(MICROSECONDS FROM TIME '17:12:28.5');
Result: 28500000
```

millennium

interval###########1000#####

```
SELECT EXTRACT(MILLENNIUM FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 3
SELECT EXTRACT(MILLENNIUM FROM INTERVAL '2001 years');
Result: 2
```

1900######2########3######2001#1#1##########

milliseconds

###############1000#######################

```
SELECT EXTRACT(MILLISECONDS FROM TIME '17:12:28.5');
Result: 28500.000
```

minute

(0–59)

```
SELECT EXTRACT(MINUTE FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 38
```

month

#######(1~12)#interval#######12######(0~11)

```
SELECT EXTRACT(MONTH FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 2
SELECT EXTRACT(MONTH FROM INTERVAL '2 years 3 months');
Result: 3
SELECT EXTRACT(MONTH FROM INTERVAL '2 years 13 months');
Result: 1
```

quarter

##############(1–4)

```
SELECT EXTRACT(QUARTER FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 1
```

second

###########

```
SELECT EXTRACT(SECOND FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 40.000000
SELECT EXTRACT(SECOND FROM TIME '17:12:28.5');
Result: 28.500000
```

timezone

####UTC################UTC################UTC########### (########PostgreSQL############UTC########)

timezone_hour

##############

timezone_minute

##############

week

ISO 8601#### ####ISO##############1#4###########1######## ##################### ###1#######

ISO#########1###########52######53#########12############################ ## ####2005-01-01#2004###53#####2006-01-01#2005###52######## #2012-12-31#2013###1######### ##############isoyear######week########### ###

```
SELECT EXTRACT(WEEK FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 7
```

year

#######AD########################AD####BC################

```
SELECT EXTRACT(YEAR FROM TIMESTAMP '2001-02-16 20:38:40');
Result: 2001
```

interval########extract####interval############################ ###########
INTERVAL############################## ############

```
SELECT INTERVAL '80 minutes';
Result: 01:20:00
SELECT EXTRACT(MINUTES FROM INTERVAL '80 minutes');
Result: 20
```

> **##**
>
> #####[61.785216]#####+/-#######extract###########
> #epoch#julian#year#isoyear#decade#century#millennium#####+/-######
> ### ##############NULL####### PostgreSQL#9.6##########################
> ########### ##########+/-Infinity####extract##############timestamp#####
> #epoch#julian#year#isoyear#decade#century#millennium#####+/Infinity
> ###### epoch#hour#day#year#decade#century####millennium####
> #interval#########/+/Infinity###### ###########NULL####### PostgreSQL
> #9.6################################

extract################# ##/####################9.8##########

date_part#######Ingres############SQL#extract########

```
date_part('field', source)
```

###*field*################################ date_part###########extract####
##########date_part###double precision#########
extract################

```
SELECT date_part('day', TIMESTAMP '2001-02-16 20:38:40');
Result: 16
SELECT date_part('hour', INTERVAL '4 hours 3 minutes');
Result: 4
```

9.9.2. date_trunc

date_trunc############trunc###########

```
date_trunc(field, source [, time_zone ])
```

source######timestamp#timestamp with time zone####interval###### #date#
#time###########timestamp####interval########### *field*##################
#######timestamp#timestamp with time zone####interval#############
######################1########

field###############

microseconds
milliseconds
second
minute
hour

day
week
month
quarter
year
decade
century
millennium

####timestamp with time zone################################## ################### ##############TimeZone################################ #*time_zone*########### #####8.5.3################

timestamp without time zone####interval###################### ######## #######

###########America/New_York#######

```
SELECT date_trunc('hour', TIMESTAMP '2001-02-16 20:38:40');
Result: 2001-02-16 20:00:00
SELECT date_trunc('year', TIMESTAMP '2001-02-16 20:38:40');
Result: 2001-01-01 00:00:00
SELECT date_trunc('day', TIMESTAMP WITH TIME ZONE '2001-02-16
 20:38:40+00');
Result: 2001-02-16 00:00:00-05
SELECT date_trunc('day', TIMESTAMP WITH TIME ZONE '2001-02-16
 20:38:40+00', 'Australia/Sydney');
Result: 2001-02-16 08:00:00-05
SELECT date_trunc('hour', INTERVAL '3 days 02:47:33');
Result: 3 days 02:00:00
```

9.9.3. date_bin

##date_bin##############################(*stride*)#########

date_bin(*stride*, *source*, *origin*)

source#timestamp####timestamp with time zone####### #date####timestamp# ########### *stride*#interval####### #######timestamp####timestamp with time zone###*source*###################

#######

```
SELECT date_bin('15 minutes', TIMESTAMP '2020-02-11 15:44:17', TIMESTAMP
 '2001-01-01');
Result: 2020-02-11 15:30:00
SELECT date_bin('15 minutes', TIMESTAMP '2020-02-11 15:44:17', TIMESTAMP
 '2001-01-01 00:02:30');
Result: 2020-02-11 15:32:30
```

#####(1##1####)###########date_trunc############### ####date_bin############# #######

stride##

9.9.4. AT TIME ZONE and AT LOCAL

AT TIME ZONE############time stamp *without* time zone##time stamp *with* time zone##### #time with time zone#################### # 9.34###########

#9.34 AT TIME ZONE and AT LOCAL Variants

#
timestamp without time zone AT TIME ZONE *zone* → timestamp with time zone ## timestamp '2001-02-16 20:38:40' at time zone 'America/Denver' → 2001-02-17 03:38:40+00
timestamp without time zone AT LOCAL → timestamp with time zone #####[61.445783]## #### ##TimeZone################ ######## timestamp '2001-02-16 20:38:40' at local → 2001-02-17 03:38:40+00
timestamp with time zone AT TIME ZONE *zone* → timestamp without time zone ## timestamp with time zone '2001-02-16 20:38:40-05' at time zone 'America/Denver' → 2001-02-16 18:38:40
timestamp with time zone AT LOCAL → timestamp without time zone #####[63.768116]## ###### ## #############TimeZone# ############### timestamp with time zone '2001-02-16 20:38:40-05' at local → 2001-02-16 18:38:40
time with time zone AT TIME ZONE *zone* → time with time zone #######*with* time zone############# ####################UTC########### ######### time with time zone '05:34:17-05' at time zone 'UTC' → 10:34:17+00
time with time zone AT LOCAL → time with time zone #####[68.020305]#######*with* time zone############# #################### UTC#################### ############################## ########### #######TimeZone######UTC########### ###########TimeZone#UTC############## time with time zone '05:34:17-05' at local → 10:34:17+00

##############*zone*###'America/Los_Angeles'##############INTERVAL '-08:00'#################### #######8.5.3######################## ######## UTC##

#######AT LOCAL##AT TIME ZONE *local*############# ####*local*###### #TimeZone####

################TimeZone##America/Los_Angeles#########

```
SELECT TIMESTAMP '2001-02-16 20:38:40' AT TIME ZONE 'America/Denver';
Result: 2001-02-16 19:38:40-08
SELECT TIMESTAMP WITH TIME ZONE '2001-02-16 20:38:40-05' AT TIME ZONE
 'America/Denver';
```

```
Result: 2001-02-16 18:38:40
SELECT TIMESTAMP '2001-02-16 20:38:40' AT TIME ZONE 'Asia/Tokyo' AT TIME
 ZONE 'America/Chicago';
Result: 2001-02-16 05:38:40
SELECT TIMESTAMP WITH TIME ZONE '2001-02-16 20:38:40-05' AT LOCAL;
Result: 2001-02-16 17:38:40
SELECT TIME WITH TIME ZONE '20:38:40-05' AT LOCAL;
Result: 17:38:40
```

#####[65.640194]#######################TimeZone############### 2######time stamp with time zone#################without a time zone####### ####TimeZone###### ############### 3#################### ##################################### #TimeZone############## 2 ######### ############ ########### ############## # #######TimeZone######################## 3 ###################### 4######### ####TimeZone#####################################

#######5########### ## ############ ##'America/Los_Angeles'##

##timezone(*zone*, *timestamp*)##SQL#####*timestamp* AT TIME ZONE *zone*######

#####[90.370370]###timezone(*zone*, *timestamp*)##SQL#####*timestamp* AT TIME ZONE *zone*######

#####[57.819905]###timezone(*zone*, *timestamp*)##SQL#####*timestamp* AT TIME ZONE *zone*###### #######timezone(*timestamp*)##SQL#####*timestamp AT LOCAL*# #####

#####[50.746269]###timezone(*zone*, *timestamp*)##SQL#####*timestamp* AT TIME ZONE *zone*###### #######timezone(*time*)##SQL#####*time AT LOCAL*######

9.9.5. #####/##

PostgreSQL############################ ######SQL############################## ####

```
CURRENT_DATE
CURRENT_TIME
CURRENT_TIMESTAMP
CURRENT_TIME(precision)
CURRENT_TIMESTAMP(precision)
LOCALTIME
LOCALTIMESTAMP
LOCALTIME(precision)
LOCALTIMESTAMP(precision)
```

CURRENT_TIME###CURRENT_TIMESTAMP####################LOCALTIME## #LOCALTIMESTAMP###################

CURRENT_TIME#CURRENT_TIMESTAMP#LOCALTIME####LOCALTIMESTAMP########### ## ############################## ##

#######

```
SELECT CURRENT_TIME;
Result: 14:39:53.662522-05
SELECT CURRENT_DATE;
Result: 2019-12-23
SELECT CURRENT_TIMESTAMP;
Result: 2019-12-23 14:39:53.662522-05
```

```
SELECT CURRENT_TIMESTAMP(2);
Result: 2019-12-23 14:39:53.66-05
SELECT LOCALTIMESTAMP;
Result: 2019-12-23 14:39:53.662522
```

```
#######################################################  ####################
#########################################################################
```

> **##**
>
> ```
> ####################################
> ```

PostgreSQL### ###SQL#################

```
transaction_timestamp()
statement_timestamp()
clock_timestamp()
timeofday()
now()
```

```
transaction_timestamp()#CURRENT_TIMESTAMP###########################
statement_timestamp()#####################################################
######  statement_timestamp()###transaction_timestamp()###############
#################################### clock_timestamp()##################
###SQL##############  timeofday()#PostgreSQL#########  clock_timestamp()#
###############timestamp    with    time    zone############text######
##now()#transaction_timestamp()##########PostgreSQL#####
```

```
######/############### now################################################ ##
##########################
```

```
SELECT CURRENT_TIMESTAMP;
SELECT now();
SELECT TIMESTAMP 'now';  -- but see tip below
```

> **###**
>
> ```
> ##########DEFAULT#######################3############### #####now####
> #############timestamp######################################## ###2####
> ################################  ######################################
> #8.5.1.4########
> ```

9.9.6.

```
################################
```

```
pg_sleep ( double precision )
pg_sleep_for ( interval )
pg_sleep_until ( timestamp with time zone )
```

```
pg_sleep#################################### pg_sleep_for#interval######
############### pg_sleep_until######################## ##########
```

```
SELECT pg_sleep(1.5);
SELECT pg_sleep_for('5 minutes');
```

```
SELECT pg_sleep_until('tomorrow 03:00');
```

> **##**
>
> ##########################0.01######### ######################### ####### ###################### ###pg_sleep_until############################## #############

> **##**
>
> pg_sleep## ##

9.10.

####8.7## ######## 9.35##### ##### ##############################

```
CREATE TYPE rainbow AS ENUM ('red', 'orange', 'yellow', 'green', 'blue',
 'purple');
```

#9.35 #########

##	## # ##
enum_first (anyenum) → anyenum	############### enum_first(null::rainbow) → red
enum_last (anyenum) → anyenum	############### enum_last(null::rainbow) → purple
enum_range (anyenum) → anyarray	######################## enum_range(null::rainbow) → {red,orange,yellow,green,blue,purple}
enum_range (anyenum, anyenum) → anyarray	############################ ################### #########NULL##### ################ #########NULL################# enum_range('orange'::rainbow, 'green'::rainbow) → {orange,yellow,green} enum_range(NULL, 'green'::rainbow) → {red,orange,yellow,green} enum_range('orange'::rainbow, NULL) → {orange,yellow,green,blue,purple}

enum_range## ####NULL########################## ##

9.11.

point#box#lseg#line#path#polygon####circle########PostgreSQL############ ############### 9.36## 9.37##### 9.38##########

#9.36

#
geometric_type + point → *geometric_type*   ##############point############## point#box#path#circle#######   box '(1,1),(0,0)' + point '(2,0)' → (3,1),(2,0)
path + path → path   2#########################NULL######   path '[(0,0),(1,1)]' + path '[(2,2),(3,3),(4,4)]' → [(0,0),(1,1),(2,2),(3,3),(4,4)]
geometric_type - point → *geometric_type*   ##############point############## point#box#path#circle#######   box '(1,1),(0,0)' - point '(2,0)' → (-1,1),(-2,0)
geometric_type * point → *geometric_type*   ##########2###point## 2##   #point##x###############   ############## point#box [a] path#circle#######   path '((0,0),(1,0),(1,1))' * point '(3.0,0)' → ((0,0),(3,0),(3,3))   path '((0,0),(1,0),(1,1))' * point(cosd(45), sind(45))   → ((0,0),(0.7071067811865475,0.7071067811865475),(0,1.414213562373095))
geometric_type / point → *geometric_type*   ##########2###point## 2##   #point##x###############   ############# point#box[a]#path#circle#######   path '((0,0),(1,0),(1,1))' / point '(2.0,0)' → ((0,0),(0.5,0),(0.5,0.5))   path '((0,0),(1,0),(1,1))' / point(cosd(45), sind(45))   → ((0,0),(0.7071067811865476,-0.7071067811865476),(1.4142135623730951,0))
@-@ *geometric_type* → double precision   ########lseg#path#######   @-@ path '[(0,0),(1,0),(1,1)]' → 2
@@ *geometric_type* → point   ######### box#lseg#polygon#circle#######   @@ box '(2,2),(0,0)' → (1,1)
# *geometric_type* → integer   ######## path#polygon#######   # path '((1,0),(0,1),(-1,0))' → 3
geometric_type # *geometric_type* → point

###############NULL###### lseg#line####### lseg '[(0,0),(1,1)]' # lseg '[(1,0),(0,1)]' → (0.5,0.5)
box # box → box 2############## #######NULL###### box '(2,2),(-1,-1)' # box '(1,1),(-2,-2)' → (1,1),(-1,-1)
geometric_type ## *geometric_type* → point ##########2##########2################### ############## (point, box), (point, lseg), (point, line), (lseg, box), (lseg, lseg), (line, lseg). point '(0,0)' ## lseg '[(2,0),(0,2)]' → (1,1)
geometric_type <-> *geometric_type* → double precision ################ 7#########point################################## ## (box#lseg)#(lseg#line)#(polygon#circle)(##########)# circle '<(0,0),1>' <-> circle '<(5,0),1>' → 3
geometric_type @> *geometric_type* → boolean #########2################ ################ (box#point)# (box#box)# (path#point)# (polygon#point)# (polygon#polygon)# (circle#point)# (circle#circle)# circle '<(0,0),2>' @> point '(1,1)' → t
geometric_type <@ *geometric_type* → boolean #########2############################ ################ (point# box)# (point# lseg)# (point# line)# (point# path)# (point# polygon)# (point# circle)# (box# box)# (lseg# box)# (lseg# line)# (polygon# polygon)# (circle# circle)# point '(1,1)' <@ circle '<(0,0),2>' → t
geometric_type && *geometric_type* → boolean #################### ################ box#polygon#circle####### box '(1,1),(0,0)' && box '(2,2),(0,0)' → t
geometric_type << *geometric_type* → boolean ############2############### point#box#polygon#circle####### circle '<(0,0),1>' << circle '<(5,0),1>' → t
geometric_type >> *geometric_type* → boolean ############2############### point#box#polygon#circle####### circle '<(5,0),1>' >> circle '<(0,0),1>' → t
geometric_type &< *geometric_type* → boolean #########2##################### box#polygon#circle####### box '(1,1),(0,0)' &< box '(2,2),(0,0)' → t
geometric_type &> *geometric_type* → boolean #########2##################### box#polygon#circle####### box '(3,3),(0,0)' &> box '(2,2),(0,0)' → t
geometric_type <<| *geometric_type* → boolean ############2############### point#box#polygon#circle####### box '(3,3),(0,0)' <<| box '(5,5),(3,4)' → t

#
geometric_type |>> *geometric_type* → boolean ############2############### point#box#polygon#circle####### box '(5,5),(3,4)' |>> box '(3,3),(0,0)' → t
geometric_type &<| *geometric_type* → boolean #########2#################### box#polygon#circle####### box '(1,1),(0,0)' &<| box '(2,2),(0,0)' → t
geometric_type |&> *geometric_type* → boolean #########2#################### box#polygon#circle####### box '(3,3),(0,0)' |&> box '(2,2),(0,0)' → t
box <^ box → boolean #########2############ ############### box '((1,1),(0,0))' <^ box '((2,2),(1,1))' → t
box >^ box → boolean #########2############ ############### box '((2,2),(1,1))' >^ box '((1,1),(0,0))' → t
geometric_type ?# *geometric_type* → boolean ################## ################ (box# box)# (lseg# box)# (lseg# lseg)# (lseg# line)# (line# box)# (line# line)# (path# path)# lseg '[(-1,0),(1,0)]' ?# box '(2,2),(-2,-2)' → t
?- line → boolean ?- lseg → boolean ###### ?- lseg '[(-1,0),(1,0)]' → t
point ?- point → boolean ########### ####y################ point '(1,0)' ?- point '(0,0)' → t
?| line → boolean ?| lseg → boolean ###### ?| lseg '[(-1,0),(1,0)]' → f
point ?| point → boolean ########### ####x################ point '(0,1)' ?| point '(0,0)' → t
line ?-| line → boolean lseg ?-| lseg → boolean #######2######## lseg '[(0,0),(0,1)]' ?-| lseg '[(0,0),(1,0)]' → t
line ?|| line → boolean lseg ?|| lseg → boolean ######

#
lseg '[(-1,0),(1,0)]' ?|| lseg '[(-1,2),(1,2)]' → t
geometric_type ~= *geometric_type* → boolean ########## point#box#polygon#circle####### polygon '((0,0),(1,1))' ~= polygon '((1,1),(0,0))' → t

[a]box########################## ##

##

#######~=####point#box#polygon####circle########################### ########=##########=##<=###### #########

##

PostgreSQL#14##################point <<| point###point |>> point##### #<^###>^######## ####################################

#9.37

#
area (*geometric_type*) → double precision ######## box#path#circle####### ##path####################NULL### ### ##path########################### area(box '(2,2),(0,0)') → 4
center (*geometric_type*) → point ######### box#circle####### center(box '(1,2),(0,0)') → (0.5,1)
diagonal (box) → lseg ################## (lseg(box)######) diagonal(box '(1,2),(0,0)') → [(1,2),(0,0)]
diameter (circle) → double precision ########## diameter(circle '<(0,0),2>') → 4
height (box) → double precision ########### height(box '(1,2),(0,0)') → 2
isclosed (path) → boolean ##### isclosed(path '((0,0),(1,1),(2,0))') → t
isopen (path) → boolean #####

```
##
      ##
      #
```

```
      isopen(path '[(0,0),(1,1),(2,0)]') → t
```

```
length ( geometric_type ) → double precision
      ######## lseg#path#######
      length(path '((-1,0),(1,0))') → 4
```

```
npoints ( geometric_type ) → integer
      ####### path#polygon#######
      npoints(path '[(0,0),(1,1),(2,0)]') → 3
```

```
pclose ( path ) → path
      ##############
      pclose(path '[(0,0),(1,1),(2,0)]') → ((0,0),(1,1),(2,0))
```

```
popen ( path ) → path
      ##############
      popen(path '((0,0),(1,1),(2,0))') → [(0,0),(1,1),(2,0)]
```

```
radius ( circle ) → double precision
      ##########
      radius(circle '<(0,0),2>') → 2
```

```
slope ( point, point ) → double precision
      2##################
      slope(point '(0,0)', point '(2,1)') → 0.5
```

```
width ( box ) → double precision
      ##########
      width(box '(1,2),(0,0)') → 1
```

#9.38

```
##
      ##
      #
```

```
box ( circle ) → box
      ##############
      box(circle '<(0,0),2>') →
      (1.414213562373095,1.414213562373095),
      (-1.414213562373095,-1.414213562373095)
```

```
box ( point ) → box
      ############
      box(point '(1,0)') → (1,0),(1,0)
```

```
box ( point, point ) → box
      2################
      box(point '(0,1)', point '(1,0)') → (1,1),(0,0)
```

```
box ( polygon ) → box
      ##############
      box(polygon '((0,0),(1,1),(2,0))') → (2,1),(0,0)
```

```
##
      ##
      #
```

```
bound_box ( box, box ) → box
      2###############
      bound_box(box '(1,1),(0,0)', box '(4,4),(3,3)') → (4,4),
      (0,0)
```

```
circle ( box ) → circle
      ###############
      circle(box '(1,1),(0,0)') → <(0.5,0.5),0.7071067811865476>
```

```
circle ( point, double precision ) → circle
      ##############
      circle(point '(0,0)', 2.0) → <(0,0),2>
```

```
circle ( polygon ) → circle
      ########### ####################################
      circle(polygon '((0,0),(1,3),(2,0))') →
      <(1,1),1.6094757082487299>
```

```
line ( point, point ) → line
      2##############
      line(point '(-1,0)', point '(1,0)') → {0,-1,0}
```

```
lseg ( box ) → lseg
      ##################
      lseg(box '(1,0),(-1,0)') → [(1,0),(-1,0)]
```

```
lseg ( point, point ) → lseg
      2############
      lseg(point '(-1,0)', point '(1,0)') → [(-1,0),(1,0)]
```

```
path ( polygon ) → path
      ####################
      path(polygon '((0,0),(1,1),(2,0))') → ((0,0),(1,1),(2,0))
```

```
point ( double precision, double precision ) → point
      ##########
      point(23.4, -44.5) → (23.4,-44.5)
```

```
point ( box ) → point
      ############
      point(box '(1,0),(-1,0)') → (0,0)
```

```
point ( circle ) → point
      ###########
      point(circle '<(0,0),2>') → (0,0)
```

```
point ( lseg ) → point
      ###########
      point(lseg '[(-1,0),(1,0)]') → (0,0)
```

```
point ( polygon ) → point
      ############ ###############
```

point(polygon '((0,0),(1,1),(2,0))') → (1,0.3333333333333333)
polygon (box) → polygon ###4########### polygon(box '(1,1),(0,0)') → ((0,0),(0,1),(1,1),(1,0))
polygon (circle) → polygon ##12########### polygon(circle '<(0,0),2>') → ((-2,0), (-1.7320508075688774,0.9999999999999999), (-1.0000000000000002,1.7320508075688772), (-1.2246063538223773e-16,2), (0.9999999999999996,1.7320508075688774), (1.732050807568877,1.0000000000000007), (2,2.4492127076447545e-16), (1.7320508075688776,-0.9999999999999994), (1.0000000000000009,-1.7320508075688767), (3.673819061467132e-16,-2), (-0.9999999999999987,-1.732050807568878), (-1.7320508075688767,-1.0000000000000009))
polygon (integer, circle) → polygon ##*n*########### polygon(4, circle '<(3,0),1>') → ((2,0),(3,1), (4,1.2246063538223773e-16),(3,-1))
polygon (path) → polygon #################### polygon(path '((0,0),(1,1),(2,0))') → ((0,0),(1,1),(2,0))

####point###0#1##############point#################### ####t.p#point### ##SELECT p[0] FROM t#####X###########UPDATE t SET p[1] = ...#Y###### ### ####box####lseg#####2##point##############

9.12.

IP#############cidr#inet## 9.1############### 9.39## 9.40#################### ##

####cidr#####inet##################inet##########cidr###### #inet#cidr## ## ##inet##cidr################ ##### ########################cidr##################

#9.39 IP#######

inet << inet → boolean #################### ########4#################### ####2######### ###

```
###
      ##
      #
```

```
      inet '192.168.1.5' << inet '192.168.1/24' → t
      inet '192.168.0.5' << inet '192.168.1/24' → f
      inet '192.168.1/24' << inet '192.168.1/24' → f
```

```
inet <<= inet → boolean
      ##########################
      inet '192.168.1/24' <<= inet '192.168.1/24' → t
```

```
inet >> inet → boolean
      ####################
      inet '192.168.1/24' >> inet '192.168.1.5' → t
```

```
inet >>= inet → boolean
      #########################
      inet '192.168.1/24' >>= inet '192.168.1/24' → t
```

```
inet && inet → boolean
      ######################
      inet '192.168.1/24' && inet '192.168.1.80/28' → t
      inet '192.168.1/24' && inet '192.168.2.0/28' → f
```

```
~ inet → inet
      ###########
      ~ inet '192.168.1.6' → 63.87.254.249
```

```
inet & inet → inet
      ##########
      inet '192.168.1.6' & inet '0.0.0.255' → 0.0.0.6
```

```
inet | inet → inet
      ##########
      inet '192.168.1.6' | inet '0.0.0.255' → 192.168.1.255
```

```
inet + bigint → inet
      ################
      inet '192.168.1.6' + 25 → 192.168.1.31
```

```
bigint + inet → inet
      ################
      200 + inet '::ffff:fff0:1' → ::ffff:255.240.0.201
```

```
inet - bigint → inet
      #################
      inet '192.168.1.43' - 36 → 192.168.1.7
```

```
inet - inet → bigint
      2##############
      inet '192.168.1.43' - inet '192.168.1.19' → 24
      inet '::1' - inet '::ffff:1' → -4294901760
```

#9.40 IP######

#
abbrev (inet) → text ################ ####inet##################text################ ################################ abbrev(inet '10.1.0.0/32') → 10.1.0.0
abbrev (cidr) → text ################ ## 8.22####### ### abbrev(cidr '10.1.0.0/16') → 10.1/16
broadcast (inet) → inet ################################## broadcast(inet '192.168.1.5/24') → 192.168.1.255/24
family (inet) → integer ##############IPv4##4##IPv6##6### family(inet '::1') → 6
host (inet) → text IP############################ host(inet '192.168.1.0/24') → 192.168.1.0
hostmask (inet) → inet ########################## hostmask(inet '192.168.23.20/30') → 0.0.0.3
inet_merge (inet, inet) → cidr ############################## inet_merge(inet '192.168.1.5/24', inet '192.168.2.5/24') → 192.168.0.0/22
inet_same_family (inet, inet) → boolean #######IP#################### inet_same_family(inet '192.168.1.5/24', inet '::1') → f
masklen (inet) → integer ################ masklen(inet '192.168.1.5/24') → 24
netmask (inet) → inet ############################ netmask(inet '192.168.1.5/24') → 255.255.255.0
network (inet) → cidr ################################## ######cidr############## network(inet '192.168.1.5/24') → 192.168.1.0/24
set_masklen (inet, integer) → inet #######inet####### ############## set_masklen(inet '192.168.1.5/24', 16) → 192.168.1.5/16
set_masklen (cidr, integer) → cidr

##	
	##
	#
	########cidr######## ####################0########
	set_masklen(cidr '192.168.1.0/24', 16) → 192.168.0.0/16
text (inet) → text	
	#######IP######################### ####text######################
	text(inet '192.168.1.5') → 192.168.1.5/32

###
##abbrev#host####text#########IP####################

MAC########macaddr#macaddr8### 9.1############ 9.41#################### ### ###IP#########################~#&#|(NOT#AND#OR)########

#9.41 MAC######

##	
	##
	#
trunc (macaddr) → macaddr	
	###################### #########PostgreSQL########################### #######
	trunc(macaddr '12:34:56:78:90:ab') → 12:34:56:00:00:00
trunc (macaddr8) → macaddr8	
	#########5############ #########PostgreSQL########################### #######
	trunc(macaddr8 '12:34:56:78:90:ab:cd:ef') → 12:34:56:00:00:00:00:00
macaddr8_set7bit (macaddr8) → macaddr8	
	7#######1#####EUI-64###########IPv6################
	macaddr8_set7bit(macaddr8 '00:34:56:ab:cd:ef') → 02:34:56:ff:fe:ab:cd:ef

9.13.

9.42## 9.43#### 9.44############################PostgreSQL##############12########

#9.42

###	
	##
	#
tsvector @@ tsquery → boolean	
tsquery @@ tsvector → boolean	
	tsvector#tsquery######## ####################
	to_tsvector('fat cats ate rats') @@ to_tsquery('cat & rat') → t

text @@ tsquery → boolean #######to_tsvector()##########tsquery####### 'fat cats ate rats' @@ to_tsquery('cat & rat') → t
tsvector || tsvector → tsvector 2##tsvector###### ##################2####################### 'a:1 b:2'::tsvector || 'c:1 d:2 b:3'::tsvector → 'a':1 'b':2,5 'c':3 'd':4
tsquery && tsquery → tsquery 2##tsquery## 'fat | rat'::tsquery && 'cat'::tsquery → ('fat' | 'rat') & 'cat'
tsquery || tsquery → tsquery 2##tsquery## 'fat | rat'::tsquery || 'cat'::tsquery → 'fat' | 'rat' | 'cat'
!! tsquery → tsquery tsquery####################################### !! 'cat'::tsquery → !'cat'
tsquery <-> tsquery → tsquery 2## to_tsquery('fat') <-> to_tsquery('rat') → 'fat' <-> 'rat'
tsquery @> tsquery → boolean ###tsquery#2######### ## ####### 'cat'::tsquery @> 'cat & rat'::tsquery → f
tsquery <@ tsquery → boolean ###tsquery#2########## ## ######## 'cat'::tsquery <@ 'cat & rat'::tsquery → t 'cat'::tsquery <@ '!cat & rat'::tsquery → t

############# 9.1###########tsvector###tsquery########## ################## ##

#9.43

array_to_tsvector (text[]) → tsvector ######tsvector###### ################################ ######## #NULL########## array_to_tsvector('{fat,cat,rat}'::text[]) → 'cat' 'fat' 'rat'
get_current_ts_config () → regconfig

#
#default_text_search_config##################OID###### get_current_ts_config() → english
length (tsvector) → integer tsvector############# length('fat:2,4 cat:3 rat:5A'::tsvector) → 3
numnode (tsquery) → integer tsquery#################### numnode('(fat & rat) | cat'::tsquery) → 5
plainto_tsquery ([*config* regconfig,] *query* text) → tsquery ##############################tsquery###### ###################################### ################ ##################################### plainto_tsquery('english', 'The Fat Rats') → 'fat' & 'rat'
phraseto_tsquery ([*config* regconfig,] *query* text) → tsquery ##############################tsquery###### ###################################### ################ ##################################### phraseto_tsquery('english', 'The Fat Rats') → 'fat' <-> 'rat' phraseto_tsquery('english', 'The Cat and Rats') → 'cat' <2> 'rat'
websearch_to_tsquery ([*config* regconfig,] *query* text) → tsquery ##############################tsquery###### ###################### ## #or##OR#####################NOT########### ############### ###### ##web################ websearch_to_tsquery('english', '"fat rat" or cat dog') → 'fat' <-> 'rat' | 'cat' & 'dog'
querytree (tsquery) → text tsquery##################### #####T############################### querytree('foo & ! bar'::tsquery) → 'foo'
setweight (*vector* tsvector, *weight* "char") → tsvector *vector########weight####### setweight('fat:2,4 cat:3 rat:5B'::tsvector, 'A') → 'cat':3A 'fat':2A,4A 'rat':5A
setweight (*vector* tsvector, *weight* "char", *lexemes* text[]) → tsvector *vector#####lexemes#####weight####### ############################ ###### ########################## setweight('fat:2,4 cat:3 rat:5,6B'::tsvector, 'A', '{cat,rat}') → 'cat':3A 'fat':2,4 'rat':5A,6A
strip (tsvector) → tsvector ######tsvector####### strip('fat:2,4 cat:3 rat:5A'::tsvector) → 'cat' 'fat' 'rat'
to_tsquery ([*config* regconfig,] *query* text) → tsquery ##############################tsquery###### #####tsquery######### ###########

#
to_tsquery('english', 'The & Fat & Rats') → 'fat' & 'rat'
to_tsvector ([*config* regconfig,] *document* text) → tsvector ##############################tsvector####### ############# to_tsvector('english', 'The Fat Rats') → 'fat':2 'rat':3
to_tsvector ([*config* regconfig,] *document* json) → tsvector to_tsvector ([*config* regconfig,] *document* jsonb) → tsvector ######################JSON########tsvector####### ############## ######### ## ####jsonb## ##JSON### to_tsvector('english', '{"aa": "The Fat Rats", "b": "dog"}'::json) → 'dog':5 'fat':2 'rat':3 to_tsvector('english', '{"aa": "The Fat Rats", "b": "dog"}'::jsonb) → 'dog':1 'fat':4 'rat':5
json_to_tsvector ([*config* regconfig,] *document* json, *filter* jsonb) → tsvector jsonb_to_tsvector ([*config* regconfig,] *document* jsonb, *filter* jsonb) → tsvector *filter*###########JSON##############################tsvector####### ######################### ## ####jsonb####JSON## #### *filter*#0##############jsonb############# "string" (########### ##)#"numeric" (###########)#"boolean" (###########)#"key" (########## #)#"all" (#######)# ########*filter*##################JSON########## ## json_to_tsvector('english', '{"a": "The Fat Rats", "b": 123}'::json, '["string", "numeric"]') → '123':5 'fat':2 'rat':3 json_to_tsvector('english', '{"cat": "The Fat Rats", "dog": 123}'::json, '"all"') → '123':9 'cat':1 'dog':7 'fat':4 'rat':5
ts_delete (*vector* tsvector, *lexeme* text) → tsvector *vector*######*lexeme*####### *lexeme*######################lexeme##### #### ts_delete('fat:2,4 cat:3 rat:5A'::tsvector, 'fat') → 'cat':3 'rat':5A
ts_delete (*vector* tsvector, *lexemes* text[]) → tsvector *vector*##*lexemes*############### *lexemes*############################ ####### *vector*######################## ts_delete('fat:2,4 cat:3 rat:5A'::tsvector, ARRAY['fat','rat']) → 'cat':3
ts_filter (*vector* tsvector, *weights* "char"[]) → tsvector *vector*##*weights*############# ts_filter('fat:2,4 cat:3b,7c rat:5A'::tsvector, '{a,b}') → 'cat':3B 'rat':5A
ts_headline ([*config* regconfig,] *document* text, *query* tsquery [, *options* text]) → text

document##*query*################## *document*#tsvector############# ######## ## 12.3.4###### ################*options*############ ts_headline('The fat cat ate the rat.', 'cat') → The fat cat ate the rat.
ts_headline ([*config* regconfig,] *document* json, *query* tsquery [, *options* text]) → text ts_headline ([*config* regconfig,] *document* jsonb, *query* tsquery [, *options* text]) → text JSON *document*##########*query*################## ###12.3.4######## ts_headline('{"cat":"raining cats and dogs"}'::jsonb, 'cat') → {"cat": "raining cats and dogs"}
ts_rank ([*weights* real[],] *vector* tsvector, *query* tsquery [, *normalization* integer]) → real *vector*#*query*#################### ###12.3.3######## ts_rank(to_tsvector('raining cats and dogs'), 'cat') → 0.06079271
ts_rank_cd ([*weights* real[],] *vector* tsvector, *query* tsquery [, *normalization* integer]) → real #############*vector*#*query*#################### ###12.3.3######## ts_rank_cd(to_tsvector('raining cats and dogs'), 'cat') → 0.1
ts_rewrite (*query* tsquery, *target* tsquery, *substitute* tsquery) → tsquery *query*#####*target*#*substitute*######## ###12.4.2.1######## ts_rewrite('a & b'::tsquery, 'a'::tsquery, 'foo|bar'::tsquery) → 'b' & ('foo' | 'bar')
ts_rewrite (*query* tsquery, *select* text) → tsquery SELECT#####################*query*########## ###12.4.2.1######## SELECT ts_rewrite('a & b'::tsquery, 'SELECT t,s FROM aliases') → 'b' & ('foo' | 'bar')
tsquery_phrase (*query1* tsquery, *query2* tsquery) → tsquery ########*query1*#*query2*###################### #<->########## tsquery_phrase(to_tsquery('fat'), to_tsquery('cat')) → 'fat' <-> 'cat'
tsquery_phrase (*query1* tsquery, *query2* tsquery, *distance* integer) → tsquery #######*distance*#######*query1*#*query2*###################### tsquery_phrase(to_tsquery('fat'), to_tsquery('cat'), 10) → 'fat' <10> 'cat'
tsvector_to_array (tsvector) → text[] tsvector############# tsvector_to_array('fat:2,4 cat:3 rat:5A'::tsvector) → {cat,fat,rat}

#

unnest (tsvector) → setof record (*lexeme* text, *positions* smallint[], *weights* text) 1####1####tsvector########### select * from unnest('cat:3 fat:2,4 rat:5A'::tsvector) →
 	
`lexeme | positions | weights`
`--------+-----------+---------`
`cat  | {3}  | {D}`
`fat  | {2,4}  | {D,D}`
`rat  | {5}  | {A}`

##

######regconfig################################default_text_search_config## ############

9.44####################################

#9.44

#
ts_debug ([*config* regconfig,] *document* text) → setof record (*alias* text, *description* text, *token* text, *dictionaries* regdictionary[], *dictionary* regdictionary, *lexemes* text[]) ######################*document*#################################### ##### ###12.8.1######## ts_debug('english', 'The Brightest supernovaes') → (asciiword,"Word, all ASCII",The, {english_stem},english_stem,{}) ...
ts_lexize (*dict* regdictionary, *token* text) → text[] ##NULL###### ## #12.8.3######## ts_lexize('english_stem', 'stars') → {star}
ts_parse (*parser_name* text, *document* text) → setof record (*tokid* integer, *token* text) #############*document*############# ###12.8.2######## ts_parse('default', 'foo - bar') → (1,foo) ...
ts_parse (*parser_oid* oid, *document* text) → setof record (*tokid* integer, *token* text) OID############*document*############# ###12.8.2######## ts_parse(3722, 'foo - bar') → (1,foo) ...
ts_token_type (*parser_name* text) → setof record (*tokid* integer, *alias* text, *description* text) ################################### ###12.8.2########

#
ts_token_type('default') → (1,asciiword,"Word, all ASCII") ...
ts_token_type(*parser_oid* oid) → setof record(*tokid* integer, *alias* text, *description* text) OID################################ ###12.8.2######## ts_token_type(3722) → (1,asciiword,"Word, all ASCII") ...
ts_stat(*sqlquery* text [, *weights* text]) → setof record(*word* text, *ndoc* integer, *nentry* integer) ###tsvector###########*sqlquery*############################## ## #12.4.4######## ts_stat('SELECT vector FROM apod') → (foo,10,15) ...

9.14. UUID##

PostgreSQL###UUID##########1######

gen_random_uuid () → uuid

##########4(####)UUID###### ###UUID################################

uuid-ossp######UUID################################

#########UUID #################

uuid_extract_timestamp (uuid) → timestamp with time zone

##########UUID#####1##timestamp with time zone###### ###############NULL ###### ################UUID#################### ###UUID###############

uuid_extract_version (uuid) → smallint

###########RFC 4122[2]##############UUID############## ##################NULL### ### ####gen_random_uuid########UUID#########4######

PostgreSQL## 9.1###UUID####################

9.15. XML##

####################xml############ xml########8.13######### xml######### #xmlparse###xmlserialize########################

##########PostgreSQL#configure --with-libxml####################

9.15.1. XML#####

SQL#####XML############################## ################################### ###XML######################

9.15.1.1. xmltext

[2] https://datatracker.ietf.org/doc/html/rfc4122

```
xmltext ( text ) → xml
```

#######xmltext#######################XML###### ######(&)########(< >)#### (“”)#######################

##

```
SELECT xmltext('< foo & bar >');
         xmltext
-------------------------
 &lt; foo &amp; bar &gt;
```

9.15.1.2. xmlcomment

```
xmlcomment ( text ) → xml
```

##xmlcomment#############XML#######XML####### ######--###########-###### ######################XML############# ###NULL######NULL######

##

```
SELECT xmlcomment('hello');

  xmlcomment
--------------
 <!--hello-->
```

9.15.1.3. xmlconcat

```
xmlconcat ( xml [, ...] ) → xml
```

##xmlconcat#####XML#########XML################## NULL#######NULL###### ###########NULL######

##

```
SELECT xmlconcat('<abc/>', '<bar>foo</bar>');

      xmlconcat
----------------------
 <abc/><bar>foo</bar>
```

XML########################## ##########XML version###########version###### ##########version######## ########standalone######yes################### ##### ###standalone################no################### ###########standalone######### standalone#########standalone##################version 1.0#version######### ###XML# XML######version################ encoding####################

##

```
SELECT xmlconcat('<?xml version="1.1"?><foo/>', '<?xml version="1.1"
 standalone="no"?><bar/>');

             xmlconcat
-----------------------------------
 <?xml version="1.1"?><foo/><bar/>
```

9.15.1.4. xmlelement

```
xmlelement ( NAME name [, XMLATTRIBUTES ( attvalue [ AS attname ]
 [, ...] ) ] [, content [, ...]] ) → xml
```

xmlelement##################XML######## ######name#attname############ ###### attvalue#content######PostgreSQL############### XMLATTRIBUTES##### XML###########content##############

##

```
SELECT xmlelement(name foo);

 xmlelement
------------
 <foo/>

SELECT xmlelement(name foo, xmlattributes('xyz' as bar));

    xmlelement
------------------
 <foo bar="xyz"/>

SELECT xmlelement(name foo, xmlattributes(current_date as bar), 'cont',
 'ent');

             xmlelement
--------------------------------------
 <foo bar="2007-01-26">content</foo>
```

###XML##################_xHHHH_#################################HHHH#16###### ####Unicode######### #######

```
SELECT xmlelement(name "foo$bar", xmlattributes('xyz' as "a&b"));

            xmlelement
----------------------------------
 <foo_x0024_bar a_x0026_b="xyz"/>
```

#############

```
CREATE TABLE test (a xml, b xml);
SELECT xmlelement(name test, xmlattributes(a, b)) FROM test;
```

##################

```
SELECT xmlelement(name test, xmlattributes('constant'), a, b) FROM test;
SELECT xmlelement(name test, xmlattributes(func(a, b))) FROM test;
```

############################## #########xml########XML######### #######

```
SELECT xmlelement(name foo, xmlattributes('xyz' as bar),
                            xmlelement(name abc),
                            xmlcomment('test'),
                            xmlelement(name xyz));

                  xmlelement
----------------------------------------------
 <foo bar="xyz"><abc/><!--test--><xyz/></foo>
```

############XML################ #######<#>####&#################### ###### ######bytea#########xmlbinary#########base64####16############## ############## ####D.3.1.3###########SQL:2006#########PostgreSQL#####################

9.15.1.5. xmlforest

```
xmlforest ( content [ AS name ] [, ...] ) → xml
```

xmlforest##################XML################## xmlelement####*name*##### ############*content*###################

##

```
SELECT xmlforest('abc' AS foo, 123 AS bar);

          xmlforest
------------------------------
 <foo>abc</foo><bar>123</bar>


SELECT xmlforest(table_name, column_name)
FROM information_schema.columns
WHERE table_schema = 'pg_catalog';

                                xmlforest
-----------------------------------------------------------------------
 <table_name>pg_authid</table_name><column_name>rolname</column_name>
 <table_name>pg_authid</table_name><column_name>rolsuper</column_name>
 ...
```

######################
##

###XML##########xmlelement################ #######xml################## XML#################

XML######################XML################## #####xmlelement# #xmlforest####################

9.15.1.6. xmlpi

```
xmlpi ( NAME name [, content ] ) → xml
```

xmlpi##XML########## xmlelement####*name*##################*content*####### ############## *content*###########?>####################

##

```
SELECT xmlpi(name php, 'echo "hello world";');

            xmlpi
-----------------------------
 <?php echo "hello world";?>
```

9.15.1.7. xmlroot

```
xmlroot ( xml, VERSION {text|NO VALUE} [, STANDALONE {YES|NO|NO
 VALUE} ] ) → xml
```

xmlroot##XML################## version################version##########standalone ###################standalone###########

```
SELECT xmlroot(xmlparse(document '<?xml version="1.1"?><content>abc</
content>'),
               version '1.0', standalone yes);

                xmlroot
----------------------------------------
 <?xml version="1.0" standalone="yes"?>
 <content>abc</content>
```

9.15.1.8. xmlagg

xmlagg (xml) → xml

##################xmlagg######### ###xmlconcat############################## ################################ ##############9.21#########

##

```
CREATE TABLE test (y int, x xml);
INSERT INTO test VALUES (1, '<foo>abc</foo>');
INSERT INTO test VALUES (2, '<bar/>');
SELECT xmlagg(x) FROM test;
        xmlagg
----------------------
 <foo>abc</foo><bar/>
```

#############4.2.7###########ORDER BY################### #######

```
SELECT xmlagg(x ORDER BY y DESC) FROM test;
        xmlagg
----------------------
 <bar/><foo>abc</foo>
```

###

```
SELECT xmlagg(x) FROM (SELECT * FROM test ORDER BY y DESC) AS tab;
        xmlagg
----------------------
 <bar/><foo>abc</foo>
```

9.15.2. XML##

##############xml############

9.15.2.1. IS DOCUMENT

xml IS DOCUMENT → boolean

#IS DOCUMENT###XML#####XML###NULL## ##NULL###### ################8.13#########

9.15.2.2. IS NOT DOCUMENT

xml IS NOT DOCUMENT → boolean

#IS NOT DOCUMENT###XML#####XML#######################################
NULL####NULL######

9.15.2.3. XMLEXISTS

```
XMLEXISTS ( text PASSING [BY {REF|VALUE}] xml [BY {REF|VALUE}] ) → boolean
```

##xmlexists#####XML############XPath 1.0############## ######################
######################## #########NULL######NULL###### ################NULL
##########XML#####XML############

#:

```
SELECT xmlexists('//town[text() = ''Toronto'']' PASSING BY VALUE
 '<towns><town>Toronto</town><town>Ottawa</town></towns>');

 xmlexists
------------
 t
(1 row)
```

PostgreSQL#BY REF##BY VALUE##########D.3.2################

SQL####xmlexists###XML##################D.3.1###########PostgreSQL#XPath
1.0###########

9.15.2.4. xml_is_well_formed

```
xml_is_well_formed ( text ) → boolean
xml_is_well_formed_document ( text ) → boolean
xml_is_well_formed_content ( text ) → boolean
```

#######text############################## xml_is_well_formed_document######
##########xml_is_well_formed_content################
xml_is_well_formed##xmloption######DOCUMENT############CONTENT#########
############ ####xml_is_well_formed####xml##############################
#XMLPARSE##############################

##

```
SET xmloption TO DOCUMENT;
SELECT xml_is_well_formed('<>');
 xml_is_well_formed
--------------------
 f
(1 row)

SELECT xml_is_well_formed('<abc/>');
 xml_is_well_formed
--------------------
 t
(1 row)

SET xmloption TO CONTENT;
SELECT xml_is_well_formed('abc');
 xml_is_well_formed
--------------------
 t
```

```
(1 row)

SELECT xml_is_well_formed_document('<pg:foo xmlns:pg="http://
postgresql.org/stuff">bar</pg:foo>');
 xml_is_well_formed_document
-----------------------------
 t
(1 row)

SELECT xml_is_well_formed_document('<pg:foo xmlns:pg="http://
postgresql.org/stuff">bar</my:foo>');
 xml_is_well_formed_document
-----------------------------
 f
(1 row)
```

#####################################

9.15.3. XML###

####xml#########PostgreSQL#XPath 1.0#######xpath###xpath_exists######
#XMLTABLE########

9.15.3.1. xpath

xpath (*xpath* text, *xml* xml [, *nsarray* text[]]) → xml[]

##xpath##XML#*xml*####XPath 1.0#*xpath*(########)###### ####XPath###############
#XML######### ###XPath##################################

2########XML#####################################

############################## ######################text##################
################## ##URI### ############
#XML#####################################XML#####xpath##################
###

#:

```
SELECT xpath('/my:a/text()', '<my:a xmlns:my="http://example.com">test</
my:a>',
             ARRAY[ARRAY['my', 'http://example.com']]);

 xpath
--------
 {test}
(1 row)
```

#####(##)################################

```
SELECT xpath('//mydefns:b/text()', '<a xmlns="http://
example.com"><b>test</b></a>',
             ARRAY[ARRAY['mydefns', 'http://example.com']]);

 xpath
--------
 {test}
(1 row)
```

9.15.3.2. xpath_exists

```
xpath_exists ( xpath text, xml xml [, nsarray text[] ] ) → boolean
```

##xpath_exists##xpath########## ######XPath 1.0#######XML################ ## ################################## #XMLEXISTS#######

##

```
SELECT xpath_exists('/my:a/text()', '<my:a xmlns:my="http://
example.com">test</my:a>',
                     ARRAY[ARRAY['my', 'http://example.com']]);

 xpath_exists
--------------
 t
(1 row)
```

9.15.3.3. xmltable

```
XMLTABLE (
    [ XMLNAMESPACES ( namespace_uri AS namespace_name [, ...] ), ]
    row_expression PASSING [BY {REF|VALUE}] document_expression [BY
 {REF|VALUE}]
    COLUMNS name { type [PATH column_expression]
 [DEFAULT default_expression] [NOT NULL | NULL]
                  | FOR ORDINALITY }
            [, ...]
) → setof record
```

xmltable#######XML#######XPath################################ ########### ###########FROM####################

######XMLNAMESPACES######################### ###*namespace_uri*#text# ##*namespace_name*######### ################XML########## ################### ###########

###*row_expression*#######XPath 1.0#(text#####)##XML######### #*document_expression*################## ######xmltable########## *document_expression*#NULL#####*row_expression*########################### #########

document_expression#*row_expression*################## ######XML######### ######################### D.3.2###########BY REF##BY VALUE##################

SQL####xmltable###XML################D.3.1###########PostgreSQL##XPath 1.0### #######

###COLUMNS#################### ######################### ######################### ####FOR ORDINALITY######################integer######## ##########NULL#### #############

FOR ORDINALITY###########*row_expression*##########################1######## ###### FOR ORDINALITY#############1####

> **##**
>
> XPath 1.0## ## #D.3.1.2########

##*column_expression*#XPath 1.0###*row_expression*#######################
#*row_expression*########################### *column_expression* ############
##################

##XPath###XML##XPath 1.0##################################xml###PostgreSQL##
#############PostgreSQL###################### ################################
##1###0########true###false######

##XPath#######XML###########PostgreSQL##xml################################
############ [3]

xml###########XML## ###############
XML##################################### ################################
#XPath 1.0 string############# #PostgreSQL##################

##XML######### ####################
######## NULL######## ####xsi:nil######### ###################text()2######
#####text()######################### XPath 1.0 string######XML#####XML#####
#########################

#############D.3.1.3##############SQL#################

##*default_expression*###############
#NULL####### ##########################

xmltable####################*default_expression*#########################
#####stable#######immutable############################
#*default_expression*###nextval######################

###NOT NULL############# NOT NULL###*column_expression*#########DEFAULT##
######*default_expression*######NULL##################

##

```
CREATE TABLE xmldata AS SELECT
xml $$
<ROWS>
  <ROW id="1">
    <COUNTRY_ID>AU</COUNTRY_ID>
    <COUNTRY_NAME>Australia</COUNTRY_NAME>
  </ROW>
  <ROW id="5">
    <COUNTRY_ID>JP</COUNTRY_ID>
    <COUNTRY_NAME>Japan</COUNTRY_NAME>
    <PREMIER_NAME>Shinzo Abe</PREMIER_NAME>
    <SIZE unit="sq_mi">145935</SIZE>
  </ROW>
  <ROW id="6">
    <COUNTRY_ID>SG</COUNTRY_ID>
    <COUNTRY_NAME>Singapore</COUNTRY_NAME>
    <SIZE unit="sq_km">697</SIZE>
  </ROW>
</ROWS>
$$ AS data;

SELECT xmltable.*
  FROM xmldata,
       XMLTABLE('//ROWS/ROW'
                PASSING data
                COLUMNS id int PATH '@id',
```

[3] ## XPath########################## ########### ################## XPath 1.0#string###

```
                        ordinality FOR ORDINALITY,
                        "COUNTRY_NAME" text,
                        country_id text PATH 'COUNTRY_ID',
                        size_sq_km float PATH 'SIZE[@unit = "sq_km"]',
                        size_other text PATH
                             'concat(SIZE[@unit!="sq_km"], " ",
 SIZE[@unit!="sq_km"]/@unit)',
                        premier_name text PATH 'PREMIER_NAME' DEFAULT 'not
 specified');

 id | ordinality | COUNTRY_NAME | country_id | size_sq_km |  size_other  |
 premier_name
----+------------+--------------+------------+------------+--------------
+---------------
  1 |          1 | Australia    | AU         |            |              |
 not specified
  5 |          2 | Japan        | JP         |            | 145935 sq_mi |
 Shinzo Abe
  6 |          3 | Singapore    | SG         |        697 |              |
 not specified
```

##########text()##########XPath################XML##################

```
CREATE TABLE xmlelements AS SELECT
xml $$
  <root>
   <element>  Hello<!-- xyxxz -->2a2<?aaaaa?> <!--x-->  bbb<x>xxx</x>CC
 </element>
  </root>
$$ AS data;

SELECT xmltable.*
  FROM xmlelements, XMLTABLE('/root' PASSING data COLUMNS element text);
         element
-------------------------
   Hello2a2   bbbxxxCC
```

#######XMLNAMESPACES#####XML#######XPath#############################

```
WITH xmldata(data) AS (VALUES ('
<example xmlns="http://example.com/myns" xmlns:B="http://example.com/b">
 <item foo="1" B:bar="2"/>
 <item foo="3" B:bar="4"/>
 <item foo="4" B:bar="5"/>
</example>'::xml)
)
SELECT xmltable.*
  FROM XMLTABLE(XMLNAMESPACES('http://example.com/myns' AS x,
                              'http://example.com/b' AS "B"),
             '/x:example/x:item'
                PASSING (SELECT data FROM xmldata)
                COLUMNS foo int PATH '@foo',
                  bar int PATH '@B:bar');
 foo | bar
-----+-----
   1 |   2
   3 |   4
   4 |   5
(3 rows)
```

9.15.4. XML#########

####################XML######### ####XML###################

```
table_to_xml ( table regclass, nulls boolean,
               tableforest boolean, targetns text ) → xml
query_to_xml ( query text, nulls boolean,
               tableforest boolean, targetns text ) → xml
cursor_to_xml ( cursor refcursor, count integer, nulls boolean,
                tableforest boolean, targetns text ) → xml
```

table_to_xml######*table*######################### regclass############# ##8.19########### query_to_xml###### #*query*################################ cursor_to_xml######*cursor*######## ################# ##

tableforest#########XML#############

```
<tablename>
  <row>
    <columnname1>data</columnname1>
    <columnname2>data</columnname2>
  </row>

  <row>
    ...
  </row>

  ...
</tablename>
```

tableforest###############XML#######

```
<tablename>
  <columnname1>data</columnname1>
  <columnname2>data</columnname2>
</tablename>

<tablename>
  ...
</tablename>

...
```

##table#############row###### ##

####################### ########XML####################### ########### ###################cursor_to_xml################## #######XML########### #xmlelement#######################

#########xmlelement###################

#####*nulls*####NULL################## #########NULL###############

```
<columnname xsi:nil="true"/>
```

###xsi#XML#############XML######### ################### #######NULL###### ############

#####*targetns*#######XML########### ############################

################################XML###########

```
table_to_xmlschema ( table regclass, nulls boolean,
                     tableforest boolean, targetns text ) → xml
query_to_xmlschema ( query text, nulls boolean,
                     tableforest boolean, targetns text ) → xml
cursor_to_xmlschema ( cursor refcursor, nulls boolean,
                      tableforest boolean, targetns text ) → xml
```

####XML########XML################################

#######XML################XML##################################### #########
######################

```
table_to_xml_and_xmlschema ( table regclass, nulls boolean,
                             tableforest boolean, targetns text ) → xml
query_to_xml_and_xmlschema ( query text, nulls boolean,
                             tableforest boolean, targetns text ) → xml
```

##

```
schema_to_xml ( schema name, nulls boolean,
                tableforest boolean, targetns text ) → xml
schema_to_xmlschema ( schema name, nulls boolean,
                      tableforest boolean, targetns text ) → xml
schema_to_xml_and_xmlschema ( schema name, nulls boolean,
                              tableforest boolean, targetns text ) → xml

database_to_xml ( nulls boolean,
                  tableforest boolean, targetns text ) → xml
database_to_xmlschema ( nulls boolean,
                        tableforest boolean, targetns text ) → xml
database_to_xml_and_xmlschema ( nulls boolean,
                                tableforest boolean, targetns text ) → xml
```

############################ #########################USAGE#################
##

###########################
##

#########################

```
<schemaname>

table1-mapping

table2-mapping

...

</schemaname>
```

#########################*tableforest*###########

##########################

```
<dbname>

<schema1name>
```

```
  ...
</schema1name>

<schema2name>
  ...
</schema2name>

...

</dbname>
```

#######################

############################ 9.1##################
#table_to_xml_and_xmlschema##HTML#############XSLT############ ########
###########XML###########

#9.1 SQL/XML###HTML#####XSLT#######

```
<?xml version="1.0"?>
<xsl:stylesheet version="1.0"
    xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema"
    xmlns="http://www.w3.org/1999/xhtml"
>

  <xsl:output method="xml"
      doctype-system="http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"
      doctype-public="-//W3C/DTD XHTML 1.0 Strict//EN"
      indent="yes"/>

  <xsl:template match="/*">
    <xsl:variable name="schema" select="//xsd:schema"/>
    <xsl:variable name="tabletypename"
                  select="$schema/xsd:element[@name=name(current())]/
@type"/>
    <xsl:variable name="rowtypename"
                  select="$schema/xsd:complexType[@name=$tabletypename]/
xsd:sequence/xsd:element[@name='row']/@type"/>

    <html>
      <head>
        <title><xsl:value-of select="name(current())"/></title>
      </head>
      <body>
        <table>
          <tr>
            <xsl:for-each select="$schema/xsd:complexType[@name=
$rowtypename]/xsd:sequence/xsd:element/@name">
              <th><xsl:value-of select="."/></th>
            </xsl:for-each>
          </tr>

          <xsl:for-each select="row">
            <tr>
              <xsl:for-each select="*">
                <td><xsl:value-of select="."/></td>
              </xsl:for-each>
            </tr>
          </xsl:for-each>
        </table>
      </body>
    </html>
```

```
  </xsl:template>

</xsl:stylesheet>
```

9.16. JSON######

###############

- JSON################
- SQL/JSON####
- ######SQL/JSON######

SQL####JSON#######################PostgreSQL*SQL/JSON*############### ########## ######## #####SQL#######SQL/JSON NULL#####JSON############################ #####JSON##RFC 7159[4]############################

SQL/JSON#########################SQL#######JSON##############################

- JSON##SQL#######
- ###########JSON###############
- SQL/JSON##########SQL/JSON##########JSON##########

SQL/JSON###########[sqltr-19075-6]######## PostgreSQL##########JSON######## ##8.14########

9.16.1. JSON#########

9.45#JSON####(8.14###)############## #### 9.1###########jsonb########json# ######## ######8.14.4##############B#################### ######JSON#### #json_agg########JSON##########json_object_agg###########jsonb# #jsonb_agg#jsonb_object_agg#####9.21########

#9.45 `json`#`jsonb`###

###	
##	
#	
`json -> integer` → `json`	
`jsonb -> integer` → `jsonb`	
JSON###*n*############ ################################	
`'[{"a":"foo"},{"b":"bar"},{"c":"baz"}]'::json -> 2` → `{"c":"baz"}`	
`'[{"a":"foo"},{"b":"bar"},{"c":"baz"}]'::json -> -3` → `{"a":"foo"}`	
`json -> text` → `json`	
`jsonb -> text` → `jsonb`	
#######JSON##################	
`'{"a": {"b":"foo"}}'::json -> 'a'` → `{"b":"foo"}`	
`json ->> integer` → `text`	
`jsonb ->> integer` → `text`	
JSON###*n*######text##########	

[4] https://datatracker.ietf.org/doc/html/rfc7159

#
'[1,2,3]'::json ->> 2 → 3
json ->> text → text jsonb ->> text → text #######JSON###########text######### '{"a":1,"b":2}'::json ->> 'b' → 2
json #> text[] → json jsonb #> text[] → jsonb #########JSON## '{"a": {"b": ["foo","bar"]}}'::json #> '{a,b,1}' → "bar"
json #>> text[] → text jsonb #>> text[] → text #########JSON########text######### '{"a": {"b": ["foo","bar"]}}'::json #>> '{a,b,1}' → bar

##
JSON#############################/##/#################NULL############### ###########

###jsonb#################### ##### 9.46###### 8.14.4############jsonb######### ##################################

#9.46 ##jsonb###

#
jsonb @> jsonb → boolean ###JSON############ #######8.14.3########## '{"a":1, "b":2}'::jsonb @> '{"b":2}'::jsonb → t
jsonb <@ jsonb → boolean ###JSON############# '{"b":2}'::jsonb <@ '{"a":1, "b":2}'::jsonb → t
jsonb ? text → boolean #############################JSON########### '{"a":1, "b":2}'::jsonb ? 'b' → t '["a", "b", "c"]'::jsonb ? 'b' → t
jsonb ?| text[] → boolean ## '{"a":1, "b":2, "c":3}'::jsonb ?| array['b', 'd'] → t
jsonb ?& text[] → boolean ## '["a", "b", "c"]'::jsonb ?& array['a', 'b'] → t

jsonb || jsonb → jsonb 2##jsonb####### 2########################## ###########2######### ######### ##############################2############# ############# ############################ '["a", "b"]'::jsonb || '["a", "d"]'::jsonb → ["a", "b", "a", "d"] '{"a": "b"}'::jsonb || '{"c": "d"}'::jsonb → {"a": "b", "c": "d"} '[1, 2]'::jsonb || '3'::jsonb → [1, 2, 3] '{"a": "b"}'::jsonb || '42'::jsonb → [{"a": "b"}, 42] ## '[1, 2]'::jsonb || jsonb_build_array('[3, 4]'::jsonb) → [1, 2, [3, 4]]
jsonb - text → jsonb #########JSON##########################JSON######### '{"a": "b", "c": "d"}'::jsonb - 'a' → {"c": "d"} '["a", "b", "c", "b"]'::jsonb - 'b' → ["a", "c"]
jsonb - text[] → jsonb ################################## '{"a": "b", "c": "d"}'::jsonb - '{a,c}'::text[] → {}
jsonb - integer → jsonb #################################### JSON################ '["a", "b"]'::jsonb - 1 → ["a"]
jsonb #- text[] → jsonb ### '["a", {"b":1}]'::jsonb #- '{1,b}' → ["a", {}]
jsonb @? jsonpath → boolean #####JSON#######JSON################?####SQL###JSON############## ############### '{"a":[1,2,3,4,5]}'::jsonb @? '$.a[*] ? (@ > 2)' → t
jsonb @@ jsonpath → boolean ###########JSON#####JSON#####################SQL###JSON######### ############ #################NULL######## '{"a":[1,2,3,4,5]}'::jsonb @@ '$.a[*] > 2' → t

##

jsonpath####@?###@@################ ###########################JSON## ############ #####jsonpath############################## ############## JSON#####################

9.47##json####jsonb#################### ######################### #RETURNING######## ###json#jsonb#bytea#####text#char#varchar##### #json############################## #######json########

#9.47 JSON####

to_json (anyelement) → json to_jsonb (anyelement) → jsonb SQL##json####jsonb###### ################################JSON### ############ #######SQL#####json#############################[a] #### ######JSON######## ######NULL############JSON################## ################# to_json('Fred said "Hi."'::text) → "Fred said \"Hi.\"" to_jsonb(row(42, 'Fred said "Hi."'::text)) → {"f1": 42, "f2": "Fred said \"Hi.\""}
array_to_json (anyarray [, boolean]) → json SQL###JSON######## ## #to_json###### array_to_json('{{1,5},{99,100}}'::int[]) → [[1,5],[99,100]]
json_array ([{ *value_expression* [FORMAT JSON] } [, ...]] [{ NULL | ABSENT } ON NULL] [RETURNING *data_type* [FORMAT JSON [ENCODING UTF8]]]) json_array ([*query_expression*] [RETURNING *data_type* [FORMAT JSON [ENCODING UTF8]]]) JSON#######*value_expression*######*query_expression*########### #### *query_expression*########SELECT############### ABSENT ON NULL##########NULL######## *query_expression*################### json_array(1,true,json '{"a":null}') → [1, true, {"a":null}] json_array(SELECT * FROM (VALUES(1),(2)) t) → [1, 2]
row_to_json (record [, boolean]) → json SQL####JSON############ ### ###to_json###### row_to_json(row(1,'foo')) → {"f1":1,"f2":"foo"}
json_build_array (VARIADIC "any") → json jsonb_build_array (VARIADIC "any") → jsonb ################JSON###variadic############ ######to_json### #to_jsonb########## json_build_array(1, 2, 'foo', 4, 5) → [1, 2, "foo", 4, 5]
json_build_object (VARIADIC "any") → json jsonb_build_object (VARIADIC "any") → jsonb variadic#######JSON############ ########################### ########## ########## ####to_json####to_jsonb########## json_build_object('foo', 1, 2, row(3,'bar')) → {"foo" : 1, "2" : {"f1":3,"f2":"bar"}}
json_object ([{ *key_expression* { VALUE | ':' } *value_expression* [FORMAT JSON [ENCODING UTF8]] }[, ...]] [{ NULL | ABSENT } ON NULL] [{ WITH | WITHOUT } UNIQUE [KEYS]] [RETURNING *data_type* [FORMAT JSON [ENCODING UTF8]]]) ##########/####JSON############ ##/########################## ### *key_expression*##text########JSON############# NULL##### ##JSON######################## WITH UNIQUE KEYS############ #*key_expression*########## ABSENT ON NULL##########NULL#####

#
#*value_expression*########### NULL ON NULL#################### ##NULL########## `json_object('code' VALUE 'P123', 'title': 'Jaws')` → `{"code" : "P123", "title" : "Jaws"}`

`json_object(text[])` → `json`
`jsonb_object(text[])` → `jsonb`
########JSON############ #############1########################### #######2#######2###2################################## ######JSON### ########
`json_object('{a, 1, b, "def", c, 3.5}')` → `{"a" : "1", "b" : "def", "c" : "3.5"}`
`json_object('{{a, 1}, {b, "def"}, {c, 3.5}}')` → `{"a" : "1", "b" : "def", "c" : "3.5"}`

`json_object(`*keys* `text[],` *values* `text[])` → `json`
`jsonb_object(`*keys* `text[],` *values* `text[])` → `jsonb`
####json_object#2################### ##########1########
`json_object('{a,b}', '{1,2}')` → `{"a": "1", "b": "2"}`

`json(`*expression* `[ FORMAT JSON [ ENCODING UTF8 ]] [ { WITH | WITHOUT } UNIQUE [ KEYS ]] )` → `json`
##########text###bytea####UTF8#########JSON######## *expression*# NULL####SQL#NULL######## WITH UNIQUE########*expression*######### ##############
`json('{"a":123, "b":[true,"foo"], "a":"bar"}')` → `{"a":123, "b":[true,"foo"], "a":"bar"}`

###### `json_scalar(`*expression*`)` 	
##########SQL######JSON########### ###NULL####SQL#NULL####### ### ################JSON################ ########JSON##########
`json_scalar(123.45)` → `123.45`
`json_scalar(CURRENT_TIMESTAMP)` → `"2022-05-10T10:51:04.62128-04:00"`

`json_serialize(`*expression* `[ FORMAT JSON [ ENCODING UTF8 ]] [ RETURNING` *data_type* `[ FORMAT JSON [ ENCODING UTF8 ]]])`
######SQL/JSON#################### *expression*#####JSON########### #UTF8########bytea### RETURNING####################bytea### ##### #text###
`json_serialize('{ "a" : 1 } ' RETURNING bytea)` → `\x7b20226122203a2031207d20`

[a] ####hstore####hstore##json#########JSON#########hstore#################JSON##############

9.48###JSON#########SQL/JSON##############

#9.48 SQL/JSON######

######### ## #
expression `IS [ NOT ] JSON [ { VALUE | SCALAR | ARRAY | OBJECT } ] [ { WITH | WITHOUT } UNIQUE [ KEYS ] ]`

```
#########
      ##
      #
```

```
      ######expression########JSON#################### SCALAR#ARRAY###
      #OBJECT##############JSON######################## WITH UNIQUE KEYS#
      ##########expression################################


      SELECT js,
        js IS JSON "json?",
        js IS JSON SCALAR "scalar?",
        js IS JSON OBJECT "object?",
        js IS JSON ARRAY "array?"
      FROM (VALUES
            ('123'), ('"abc"'), ('{"a": "b"}'), ('[1,2]'),('abc'))
       foo(js);
           js     | json? | scalar? | object? | array?
      ------------+-------+---------+---------+--------
       123        | t     | t       | f       | f
       "abc"      | t     | t       | f       | f
       {"a": "b"} | t     | f       | t       | f
       [1,2]      | t     | f       | f       | t
       abc        | f     | f       | f       | f


      SELECT js,
        js IS JSON OBJECT "object?",
        js IS JSON ARRAY "array?",
        js IS JSON ARRAY WITH UNIQUE KEYS "array w. UK?",
        js IS JSON ARRAY WITHOUT UNIQUE KEYS "array w/o UK?"
      FROM (VALUES ('[{"a":"1"},
       {"b":"2","b":"3"}]')) foo(js);
      -[ RECORD 1 ]-+--------------------
      js            | [{"a":"1"},        +
                    |  {"b":"2","b":"3"}]
      object?       | f
      array?        | t
      array w. UK?  | f
      array w/o UK? | t
```

9.49#json#jsonb##################

#9.49 JSON####

```
##
      ##
      #
```

```
 json_array_elements(json)→setof json
 jsonb_array_elements(jsonb)→setof jsonb
      #######JSON###JSON###########
      select * from json_array_elements('[1,true, [2,false]]')→


         value
      -----------
       1
       true
       [2,false]
```

```
 json_array_elements_text(json)→setof text
 jsonb_array_elements_text(jsonb)→setof text
```

```
##
      ##
      #
      #######JSON###text##########
      select * from json_array_elements_text('["foo", "bar"]') →


         value
      -----------
       foo
       bar
```

```
json_array_length(json) → integer
jsonb_array_length(jsonb) → integer
      #######JSON###########
      json_array_length('[1,2,3,{"f1":1,"f2":[5,6]},4]') → 5
      jsonb_array_length('[]') → 0
```

```
json_each(json) → setof record(key text, value json)
jsonb_each(jsonb) → setof record(key text, value jsonb)
      #######JSON#######################
      select * from json_each('{"a":"foo", "b":"bar"}') →


       key | value
      -----+-------
       a   | "foo"
       b   | "bar"
```

```
json_each_text(json) → setof record(key text, value text)
jsonb_each_text(jsonb) → setof record(key text, value text)
      #######JSON####################### ####value#text####
      select * from json_each_text('{"a":"foo", "b":"bar"}') →


       key | value
      -----+-------
       a   | foo
       b   | bar
```

```
json_extract_path(from_json json, VARIADIC path_elems text[]) → json
jsonb_extract_path(from_json jsonb, VARIADIC path_elems text[]) →
      jsonb
      ##########JSON################ #####>################variadic##########
      #############
      json_extract_path('{"f2":{"f3":1},"f4":
      {"f5":99,"f6":"foo"}}', 'f4', 'f6') → "foo"
```

```
json_extract_path_text(from_json json, VARIADIC path_elems text[])
      → text
jsonb_extract_path_text(from_json jsonb, VARIADIC path_elems text[]
      ) → text
      ##########JSON#########text########## ##########>>##########
      json_extract_path_text('{"f2":{"f3":1},"f4":
      {"f5":99,"f6":"foo"}}', 'f4', 'f6') → foo
```

```
json_object_keys(json) → setof text
```

##
##
#

jsonb_object_keys(jsonb) → setof text
#######JSON##################
select * from json_object_keys('{"f1":"abc","f2":{"f3":"a", "f4":"b"}}') →

```
 json_object_keys
------------------
 f1
 f2
```

---

json_populate_record(*base* anyelement, *from_json* json) → anyelement
jsonb_populate_record(*base* anyelement, *from_json* jsonb) → anyelement
#######JSON#######*base*################## JSON#######################
######## ######################## ###########*base*#####NULL########
##################NULL############### ####*base*#NULL################
########
JSON######SQL######################

- #######JSON#NULL##SQL#NULL#######
- ####json#####jsonb####JSON##############
- ######(#)##JSON##JSON##
  ###########
- ###########JSON##JSON##########################JSON############
  #######
- ########JSON##
- #########JSON################################

#########JSON#####################json###jsonb##FROM##############
######## FROM##json_populate_record############## ################
##################
create type subrowtype as (d int, e text); create type myrowtype as (a int, b text[], c subrowtype);
select * from json_populate_record(null::myrowtype, '{"a": 1, "b": ["2", "a b"], "c": {"d": 4, "e": "a b c"}, "x": "foo"}') →

```
 a |   b       |      c
---+-----------+-------------
 1 | {2,"a b"} | (4,"a b c")
```

---

jsonb_populate_record_valid(*base* anyelement, *from_json* json) → boolean
#####jsonb_populate_record######## #######JSON########
#jsonb_populate_record#############true#########false######
create type jsb_char2 as (a char(2));
select jsonb_populate_record_valid(NULL::jsb_char2, '{"a": "aaa"}'); →

```
 jsonb_populate_record_valid
-----------------------------
 f
(1 row)
```

##
##
#

```
select * from jsonb_populate_record(NULL::jsb_char2, '{"a":
"aaa"}') q; →

ERROR:  value too long for type character(2)
```

```
select jsonb_populate_record_valid(NULL::jsb_char2, '{"a":
"aa"}'); →

 jsonb_populate_record_valid
-----------------------------
 t
(1 row)
```

```
select * from jsonb_populate_record(NULL::jsb_char2, '{"a":
"aa"}') q; →

 a
----
 aa
(1 row)
```

`json_populate_recordset(`*base* `anyelement,` *from_json* `json) → setof anyelement`

`jsonb_populate_recordset(`*base* `anyelement,` *from_json* `jsonb) → setof anyelement`

######JSON######*base*#################### JSON#########
#json[b]_populate_record##############

```
create type twoints as (a int, b int);
select * from json_populate_recordset(null::twoints,
'[{"a":1,"b":2}, {"a":3,"b":4}]') →

 a | b
---+---
 1 | 2
 3 | 4
```

`json_to_record(json) → record`

`jsonb_to_record(jsonb) → record`

######JSON######AS################### #record#################### #####AS######################## ##json[b]_populate_record######### ######JSON################### ######################NULL########

```
create type myrowtype as (a int, b text);
select * from json_to_record('{"a":1,"b":[1,2,3],"c":
[1,2,3],"e":"bar","r": {"a": 123, "b": "a b c"}}') as x(a
int, b text, c int[], d text, r myrowtype) →

 a |    b    |    c    | d |       r
---+---------+---------+---+---------------
 1 | [1,2,3] | {1,2,3} |   | (123,"a b c")
```

`json_to_recordset(json) → setof record`

jsonb_to_recordset (jsonb) → setof record #######JSON###AS################### #record####################### #AS######################## ##json[b]_populate_record########JSON ############ select * from json_to_recordset('[{"a":1,"b":"foo"}, {"a":"2","c":"bar"}]') as x(a int, b text) → a | b ---+----- 1 | foo 2 |
jsonb_set (*target* jsonb, *path* text[], *new_value* jsonb [, *create_if_missing* boolean]) → jsonb *path*########*new_value*#####*target*###### *create_if_missing*##### #######*path*############*new_value*######## ######################## #######*target*############### ##################*path*#########JSON# ########### ########################*create_if_missing*########### ################################ jsonb_set('[{"f1":1,"f2":null},2,null,3]', '{0,f1}', '[2,3,4]', false) → [{"f1": [2, 3, 4], "f2": null}, 2, null, 3] jsonb_set('[{"f1":1,"f2":null},2]', '{0,f3}', '[2,3,4]') → [{"f1": 1, "f2": null, "f3": [2, 3, 4]}, 2]
jsonb_set_lax (*target* jsonb, *path* text[], *new_value* jsonb [, *create_if_missing* boolean [, *null_value_treatment* text]]) → jsonb *new_value*#NULL#####jsonb_set######## ### #######*null_value_treatment*######## *null_value_treatment*##'raise_exception'#'use_json_null'#'delete_key'#'ret ############## ######'use_json_null'### jsonb_set_lax('[{"f1":1,"f2":null},2,null,3]', '{0,f1}', null) → [{"f1": null, "f2": null}, 2, null, 3] jsonb_set_lax('[{"f1":99,"f2":null},2]', '{0,f3}', null, true, 'return_target') → [{"f1": 99, "f2": null}, 2]
jsonb_insert (*target* jsonb, *path* text[], *new_value* jsonb [, *insert_after* boolean]) → jsonb *new_value*#####*target*###### *path*#############*insert_after*######## #####*new_value*############*insert_after*############## *path*######## ###*new_value*######## ########### #####################*target*############### path###############*path*# ######JSON############ ## ################ jsonb_insert('{"a": [0,1,2]}', '{a, 1}', '"new_value"') → {"a": [0, "new_value", 1, 2]} jsonb_insert('{"a": [0,1,2]}', '{a, 1}', '"new_value"', true) → {"a": [0, 1, "new_value", 2]}
json_strip_nulls (json) → json jsonb_strip_nulls (jsonb) → jsonb

#####JSON###NULL#################### ###############NULL####### ## `json_strip_nulls('[{"f1":1, "f2":null}, 2, null, 3]')` → `[{"f1":1},2,null,3]`

`jsonb_path_exists` (*target* `jsonb`, *path* `jsonpath` [, *vars* `jsonb` [, *silent* `boolean`]]) → `boolean`
#####[65.015480]#JSON######JSON#################### *vars*#########
##JSON####################jsonpath#################### *silent*#
########true#######@?#@@#################### ######JSON #######
# JSON ################### ################SQL###JSON########### 	
################### *vars*#############JSON#################### 	
#jsonpath################## *silent*#######true######@?####@@#####
##############
`jsonb_path_exists('{"a":[1,2,3,4,5]}', '$.a[*] ? (@ >= $min && @ <= $max)', '{"min":2, "max":4}')` → `t`

`jsonb_path_match` (*target* `jsonb`, *path* `jsonpath` [, *vars* `jsonb` [, *silent* `boolean`]]) → `boolean`
#####[55.623722]#####JSON##JSON################ ################# #
##Boolean######null###### ######*vars*#*silent*###jsonb_path_exists#
######### ###########JSON#####JSON############################ 	
############NULL######SQL###JSON######################## #####
#*vars*#*silent*####jsonb_path_exists###########
`jsonb_path_match('{"a":[1,2,3,4,5]}', 'exists($.a[*] ? (@ >= $min && @ <= $max))', '{"min":2, "max":4}')` → `t`

`jsonb_path_query` (*target* `jsonb`, *path* `jsonpath` [, *vars* `jsonb` [, *silent* `boolean`]]) → `setof jsonb`
###########JSON##JSON#############JSON######## SQL###JSON
#######*target*######JSON####### ######################## ##
#true#false#null####### ######*vars*#*silent*###jsonb_path_exists##
#########
`select * from jsonb_path_query('{"a":[1,2,3,4,5]}', '$.a[*] ? (@ >= $min && @ <= $max)', '{"min":2, "max":4}')`
→
 	
`jsonb_path_query`
`------------------`
`2`
`3`
`4`

`jsonb_path_query_array` (*target* `jsonb`, *path* `jsonpath` [, *vars* `jsonb` [, *silent* `boolean`]]) → `jsonb`
###########JSON##JSON#############JSON####JSON######### #####
#jsonb_path_query######
`jsonb_path_query_array('{"a":[1,2,3,4,5]}', '$.a[*] ? (@ >= $min && @ <= $max)', '{"min":2, "max":4}')` → `[2, 3, 4]`

`jsonb_path_query_first` (*target* `jsonb`, *path* `jsonpath` [, *vars* `jsonb` [, *silent* `boolean`]]) → `jsonb` ###########JSON##JSON#############JSON######## ########NULL### ##### #jsonb_path_query######

##	## #
	jsonb_path_query_first('{"a":[1,2,3,4,5]}', '$.a[*] ? (@ >= $min && @ <= $max)', '{"min":2, "max":4}') → 2
jsonb_path_exists_tz (*target* jsonb, *path* jsonpath [, *vars* jsonb [, *silent* boolean]]) → boolean jsonb_path_match_tz (*target* jsonb, *path* jsonpath [, *vars* jsonb [, *silent* boolean]]) → boolean jsonb_path_query_tz (*target* jsonb, *path* jsonpath [, *vars* jsonb [, *silent* boolean]]) → setof jsonb jsonb_path_query_array_tz (*target* jsonb, *path* jsonpath [, *vars* jsonb [, *silent* boolean]]) → jsonb jsonb_path_query_first_tz (*target* jsonb, *path* jsonpath [, *vars* jsonb [, *silent* boolean]]) → jsonb	##_tz#################### ### ########2015-08-02###########################TimeZone######## ## ################(stable)## ####### ######(immutable)## jsonb_path_exists_tz('["2015-08-01 12:00:00-05"]', '$[*] ? (@.datetime() < "2015-08-02".datetime())') → t
jsonb_pretty (jsonb) → text	#####JSON####################### jsonb_pretty('[{"f1":1,"f2":null}, 2]') → [{ "f1": 1, "f2": null }, 2]
json_typeof (json) → text jsonb_typeof (jsonb) → text	#######JSON################## ############ object#array#string#number#boolean#null# #null####SQL#NULL#### #################### json_typeof('-123.4') → number json_typeof('null'::json) → null json_typeof(NULL::json) IS NULL → t

9.16.2. SQL/JSON####

#####[86.280488]#SQL/JSON#####XML##SQL#########XPath###JSON############### ## PostgreSQL######jsonpath##############8.14.7######################## ## ####SQL/JSON#####JSON############### XPath###XML#######################PostgreSQL#######jsonpath############8.14.7####################

#####[69.728601]#JSON#####################*path engine*######### #########JSON ###############SQL/JSON######### ####SQL/JSON######################### ### ###JSON############################ ######JSON##################JSON########

######### ###########NULL#false#################### ####SQL/JSON############
##############

####jsonpath#################### ##
###########JSON#######################JSON#############

#####[67.279412]########*context item*##JSON###############$####### ##############
########### ######JSON############################## ########################
###########JSON#######################$######### ###########$###########
###JSON#######1################################1############## ##############
#################0##1##################

###############GPS#######JSON##########

```
SELECT '{
  "track": {
    "segments": [
      {
        "location":   [ 47.763, 13.4034 ],
        "start time": "2018-10-14 10:05:14",
        "HR": 73
      },
      {
        "location":   [ 47.706, 13.2635 ],
        "start time": "2018-10-14 10:39:21",
        "HR": 135
      }
    ]
  }
}' AS json \gset
```

############psql####&######################## ######psql#:'json'#############
#######JSON#######

#####[88.297872]#################### *.key* ###############JSON################
.key ################JSON################### #:

```
=> select jsonb_path_query(:'json', '$.track.segments');

 jsonb_path_query
------------------------------------------------------------
------------------------------------------------------------
----------------------------------------------
 [{"HR": 73, "location": [47.763, 13.4034], "start time": "2018-10-14
 10:05:14"}, {"HR": 135, "location": [47.706, 13.2635], "start time":
 "2018-10-14 10:39:21"}]
```

#####[83.068783]################[*]######## ################################
####################[*]#########

```
=> select jsonb_path_query(:'json', '$.track.segments[*].location');
 jsonb_path_query
-------------------
 [47.763, 13.4034]
 [47.706, 13.2635]
```

##########JSON###($)#######.track#####"track"###############JSON########
##.segments##############"segments"#########JSON#######[*]##############
############.location####################"location"#########JSON#######
####################"location"##################.location###########
######################

##################[]############################ JSON########0##############
#####

```
=> select jsonb_path_query(:'json', '$.track.segments[0].location');
 jsonb_path_query
-------------------
 [47.763, 13.4034]
```

#####[92.000000]############9.16.2.3############jsonpath#################### ##
######################### ####################

```
=> select jsonb_path_query(:'json', '$.track.segments.size()');
 jsonb_path_query
------------------
 2
```

#####jsonpath#######################9.16.2.3#########

#####[79.638009]###########SQL#WHERE######################## ################
################# ###########SQL#WHERE######################## #############
##############

? (*condition*)

#####[94.711538]################################### ##########################
############# SQL/JSON#3#############true#false#unknown###### unknown###
SQL#NULL#########is unknown######### #########true####################
##

#####[92.896175]###################### 9.51########## ########################@###
########################### ########################@##############

#####[77.310924]#####130### ###########130
######################### ##################

```
=> select jsonb_path_query(:'json', '$.track.segments[*].HR ? (@ > 130)');
 jsonb_path_query
------------------
 135
```

#####[92.796610]## #######
######################################

```
=> select jsonb_path_query(:'json', '$.track.segments[*] ? (@.HR >
 130)."start time"');
   jsonb_path_query
-----------------------
 "2018-10-14 10:39:21"
```

#####[84.422111]############################ ##################################
######### ################################ ##################################
###########

```
=> select jsonb_path_query(:'json', '$.track.segments[*] ? (@.location[1]
 < 13.4) ? (@.HR > 130)."start time"');
   jsonb_path_query
-----------------------
 "2018-10-14 10:39:21"
```

################################ ##

```
=> select jsonb_path_query(:'json', '$.track.segments[*] ? (@.location[1]
 < 13.4).HR ? (@ > 130)');
 jsonb_path_query
------------------
 135
```

#####[69.518717]### ######### ################################ ## #############

```
=> select jsonb_path_query(:'json', '$.track ? (exists(@.segments[*] ?
 (@.HR > 130))).segments.size()');
 jsonb_path_query
------------------
 2
```

9.16.2.1. Deviations from the SQL Standard

#####[95.035461]#PostgreSQL#SQL/JSON########SQL/JSON#############

9.16.2.1.1. Boolean Predicate Check Expressions

######SQL#########PostgreSQL################SQL#################### SQL###### ##########JSON############################3###true#false#unknown###### ##### ## SQL ###############

```
=> select jsonb_path_query(:'json', '$.track.segments ?(@[*].HR > 130)');
                                jsonb_path_query
---------------------------------------------------------
----------------------
 {"HR": 135, "location": [47.706, 13.2635], "start time": "2018-10-14
 10:39:21"}
```

##################true##################

```
=> select jsonb_path_query(:'json', '$.track.segments[*].HR > 130');
 jsonb_path_query
------------------
 true
```

> ##
>
> ##############@@#######jsonb_path_match##########@?######
> #jsonb_path_exists#################

9.16.2.1.2. Regular Expression Interpretation

#####[92.146597]#9.16.2.4########like_regex##################################

9.16.2.2.

#####[88.043478]#JSON###################JSON################### ################ ################################ SQL/JSON#################2########### #### ##JSON###################JSON################### ############################## #################### SQL/JSON################2###########

- #####[54.922280]####(lax)##########— ################################ ######### ######SQL/JSON############## ######## ####### - ################################ #### ####################################
- ##(strict)### — ####################

#####[87.521368]#########JSON####################JSON################### ##### ###################SQL/JSON######################SQL/JSON################## # ################################SQL/JSON################### ###1############### ######### ################### ###############JSON#########################JSON# ###################### ############################SQL/JSON################### ###########SQL/JSON########################### ################################ ##########SQL/JSON############# ###1################### ####################

- ##############type()#size()#######
- ########JSON############## ################################## ################ ##1##################

#####[88.652482]#########GPS## ###### ########GPS###

```
=> select jsonb_path_query(:'json', 'lax $.track.segments.location');
 jsonb_path_query
-------------------
 [47.763, 13.4034]
 [47.706, 13.2635]
```

########################### JSON ##

```
=> select jsonb_path_query(:'json', 'strict $.track.segments.location');
ERROR:  jsonpath member accessor can only be applied to an object
```

#####################segments#####################

```
=> select jsonb_path_query(:'json', 'strict
 $.track.segments[*].location');
 jsonb_path_query
-------------------
 [47.763, 13.4034]
 [47.706, 13.2635]
```

#####.**####################HR##2####

```
=> select jsonb_path_query(:'json', 'lax $.**.HR');
 jsonb_path_query
------------------
 73
 135
 73
 135
```

#####[92.039801]####.**#####segments################### ###.HR############### ########### ##############.**######################## ########HR##############

```
=> select jsonb_path_query(:'json', 'strict $.**.HR');
 jsonb_path_query
------------------
 73
 135
```

################################## ############ ###location###########

```
=> select jsonb_path_query(:'json', 'lax $.track.segments[*].location');
 jsonb_path_query
-------------------
 [47.763, 13.4034]
 [47.706, 13.2635]
(2 rows)
```

###################### ##

```
=> select jsonb_path_query(:'json', 'lax $.track.segments[*].location ?
(@[*] > 15)');
 jsonb_path_query
------------------
 47.763
 47.706
(2 rows)
```

###############path####################### #######################

```
=> select jsonb_path_query(:'json', 'strict $.track.segments[*].location ?
(@[*] > 15)');
 jsonb_path_query
-------------------
 [47.763, 13.4034]
 [47.706, 13.2635]
(2 rows)
```

9.16.2.3. SQL/JSON##########

#####[57.425743]## 9.50#jsonpath################## ########################### ## ####### 9.50##jsonpath########## ########## ## ### ################################### ###.###()#.###()###############

#9.50 jsonpath#######

###/####   ##   #
number + *number* → *number*   ##   jsonb_path_query('[2]', '$[0] + 3') → 5
+ *number* → *number*   ##################################   jsonb_path_query_array('{"x": [2,3,4]}', '+ $.x') → [2, 3, 4]
number - *number* → *number*   ##   jsonb_path_query('[2]', '7 - $[0]') → 5
- *number* → *number*   #########################   jsonb_path_query_array('{"x": [2,3,4]}', '- $.x') → [-2, -3, -4]

###/####   ##   #
number * *number* → *number*   ##   jsonb_path_query('[4]', '2 * $[0]') → 8
number / *number* → *number*   ##   jsonb_path_query('[8.5]', '$[0] / 2') → 4.2500000000000000
number % *number* → *number*   ######   jsonb_path_query('[32]', '$[0] % 10') → 2
value . type() → *string*   JSON#####json_typeof####   jsonb_path_query_array('[1, "2", {}]', '$[*].type()') → ["number", "string", "object"]
value . size() → *number*   JSON####################1#   jsonb_path_query('{"m": [11, 15]}', '$.m.size()') → 2
value . boolean() → *boolean*   ######JSON##########################   jsonb_path_query_array('[1, "yes", false]',   '$[*].boolean()') → [true, true, false]
value . string() → *string*   ######JSON##############################   jsonb_path_query_array('[1.23, "xyz", false]',   '$[*].string()') → ["1.23", "xyz", "false"]   jsonb_path_query('"2023-08-15"', '$.datetime().string()') → "2023-08-15"
value . double() → *number*   JSON#######################   jsonb_path_query('{"len": "1.9"}', '$.len.double() * 2') → 3.8
number . ceiling() → *number*   #########################   jsonb_path_query('{"h": 1.3}', '$.h.ceiling()') → 2
number . floor() → *number*   #########################   jsonb_path_query('{"h": 1.7}', '$.h.floor()') → 1
number . abs() → *number*   ##########   jsonb_path_query('{"z": -0.3}', '$.z.abs()') → 0.3
value . bigint() → *bigint*   ######JSON ######################

###/#### ## #
jsonb_path_query('{"len": "9876543219"}', '$.len.bigint()') → 9876543219
value . decimal([*precision* [, *scale*]]) → *decimal* ######JSON########################## precision#scale############ ## jsonb_path_query('1234.5678', '$.decimal(6, 2)') → 1234.57
value . integer() → *integer* ######JSON################## jsonb_path_query('{"len": "12345"}', '$.len.integer()') → 12345
value . number() → *numeric* ######JSON################# jsonb_path_query('{"len": "123.45"}', '$.len.number()') → 123.45
string . datetime() → *datetime_type* (see note) ########### jsonb_path_query('["2015-8-1", "2015-08-12"]', '$[*] ? (@.datetime() < "2015-08-2".datetime())') → "2015-8-1"
string . datetime(*template*) → *datetime_type* (see note) ###to_timestamp#################### jsonb_path_query_array('["12:30", "18:40"]', '$[*].datetime("HH24:MI")') → ["12:30:00", "18:40:00"]
string . date() → *date* #####[61.764706]############ ################### jsonb_path_query('"2023-08-15"', '$.date()') → "2023-08-15"
string . time() → *time without time zone* #####[67.307692]############ ######################## jsonb_path_query('"12:34:56"', '$.time()') → "12:34:56"
string . time(*precision*) → *time without time zone* ## jsonb_path_query('"12:34:56.789"', '$.time(2)') → "12:34:56.79"
string . time_tz() → *time with time zone* #####[69.387755]############ ############################ jsonb_path_query('"12:34:56 +05:30"', '$.time_tz()') → "12:34:56+05:30"
string . time_tz(*precision*) → *time with time zone* ############################ ################## jsonb_path_query('"12:34:56.789 +05:30"', '$.time_tz(2)') → "12:34:56.79+05:30"
string . timestamp() → *timestamp without time zone* #####[63.157895]############ ##################################

###/#### ## #
jsonb_path_query('"2023-08-15 12:34:56"', '$.timestamp()') → "2023-08-15T12:34:56"
string . timestamp(*precision*) → *timestamp without time zone* ############# ################### ################## jsonb_path_query('"2023-08-15 12:34:56.789"', '$.timestamp(2)') → "2023-08-15T12:34:56.79"
string . timestamp_tz() → *timestamp with time zone* #####[66.666667]########### ################################ jsonb_path_query('"2023-08-15 12:34:56 +05:30"', '$.timestamp_tz()') → "2023-08-15T12:34:56+05:30"
string . timestamp_tz(*precision*) → *timestamp with time zone* ################################ ################## jsonb_path_query('"2023-08-15 12:34:56.789 +05:30"', '$.timestamp_tz(2)') → "2023-08-15T12:34:56.79+05:30"
object . keyvalue() → *array* ###3################################/#### "key"#"value"#"id"# "id"# ##/####################### jsonb_path_query_array('{"x": "20", "y": 32}', '$.keyvalue()') → [{"id": 0, "key": "x", "value": "20"}, {"id": 0, "key": "y", "value": 32}]

##

datetime()#datetime(*template*)#####date#timetz#time#timestamptz# ####timestamp### ###################

datetime()##########date#timetz#time#timestamptz#timestamp#ISO## ############## ############################

datetime(*template*)#######################################

datetime()#datetime(*template*)#to_timestampSQL#########(##9.8)#### ###3######### ##

#####[58.453473]################################ date##timestamp### #timestamptz######### timestamp#timestamptz##time#timetz###### ### ###################TimeZone####################jsonpath######### ### ##########/################################ date##timestamp## #timestamptz##timestamp#timestamptz##time#timetz####### ######## ###################TimeZone#########timezone###jsonpath############ ## #######/############/##############TimeZone###################### ####### #######timezone#####jsonpath############

9.51###################

#9.51 jsonpath#######

##/# ## # *value* == *value* → boolean ###################JSON########## jsonb_path_query_array('[1, "a", 1, 3]', '$[*] ? (@ == 1)') → [1, 1] jsonb_path_query_array('[1, "a", 1, 3]', '$[*] ? (@ == "a")') → ["a"]
value != *value* → boolean *value* <> *value* → boolean ##### jsonb_path_query_array('[1, 2, 1, 3]', '$[*] ? (@ != 1)') → [2, 3] jsonb_path_query_array('["a", "b", "c"]', '$[*] ? (@ <> "b")') → ["a", "c"]
value < *value* → boolean #### jsonb_path_query_array('[1, 2, 3]', '$[*] ? (@ < 2)') → [1]
value <= *value* → boolean #### jsonb_path_query_array('["a", "b", "c"]', '$[*] ? (@ <= "b")') → ["a", "b"]
value > *value* → boolean ###### jsonb_path_query_array('[1, 2, 3]', '$[*] ? (@ > 2)') → [3]
value >= *value* → boolean #### jsonb_path_query_array('[1, 2, 3]', '$[*] ? (@ >= 2)') → [2, 3]
true → boolean JSON### jsonb_path_query('[{"name": "John", "parent": false}, {"name": "Chris", "parent": true}]', '$[*] ? (@.parent == true)') → {"name": "Chris", "parent": true}
false → boolean JSON### jsonb_path_query('[{"name": "John", "parent": false}, {"name": "Chris", "parent": true}]', '$[*] ? (@.parent == false)') → {"name": "John", "parent": false}
null → *value* JSON##null#SQL####null####################### jsonb_path_query('[{"name": "Mary", "job": null}, {"name": "Michael", "job": "driver"}]', '$[*] ? (@.job == null) .name') → "Mary"

##/# ## #
boolean && *boolean* → boolean ##AND jsonb_path_query('[1, 3, 7]', '$[*] ? (@ > 1 && @ < 5)') → 3
boolean || *boolean* → boolean ##OR jsonb_path_query('[1, 3, 7]', '$[*] ? (@ < 1 || @ > 5)') → 7
! *boolean* → boolean ##NOT jsonb_path_query('[1, 3, 7]', '$[*] ? (!(@ < 5))') → 7
boolean is unknown → boolean #####unknown############# jsonb_path_query('[-1, 2, 7, "foo"]', '$[*] ? ((@ > 0) is unknown)') → "foo"
string like_regex *string* [flag *string*] → boolean ########2################################ ######flag############## ###9.16.2.4########## jsonb_path_query_array('["abc", "abd", "aBdC", "abdacb", "babc"]', '$[*] ? (@ like_regex "^ab.*c")') → ["abc", "abdacb"] jsonb_path_query_array('["abc", "abd", "aBdC", "abdacb", "babc"]', '$[*] ? (@ like_regex "^ab.*c" flag "i")') → ["abc", "aBdC", "abdacb"]
string starts with *string* → boolean 2######1########################## jsonb_path_query('["John Smith", "Mary Stone", "Bob Johnson"]', '$[*] ? (@ starts with "John")') → "John Smith"
exists (*path_expression*) → boolean ###########SQL/JSON################## ############unknown######2## #################################### jsonb_path_query('{"x": [1, 2], "y": [2, 4]}', 'strict $.* ? (exists (@ ? (@[*] > 2)))') → [2, 4] jsonb_path_query_array('{"value": 41}', 'strict $? (exists (@.name)) .name') → []

9.16.2.4. SQL/JSON####

SQL/JSON#####like_regex################################ #######SQL/JSON#####
##

```
$[*] ? (@ like_regex "^[aeiou]" flag "i")
```

######flag#################### i################m#^#$########s#.########q####
################################

SQL/JSON###########XQuery#######LIKE_REGEX########### PostgreSQL###
#LIKE_REGEX############## #####like_regex######9.7.3########POSIX##########
########9.7.3.8########SQL/JSON###################

SQL/JSON###################SQL/JSON##XQuery#######POSIX#################### ######

like_regex########8.14.7####################JSON######################## ###### ### ############root#################### ####

```
$.* ? (@ like_regex "^\\d+$")
```

9.16.3. SQL/JSON Query Functions

####### 9.52######SQL/JSON##JSON_EXISTS()#JSON_###)#JSON_VALUE()##JSON### ########## ##########_##SQL/JSON############_################## ##_###### #######9.16.2######### ##_#################### ##################PASSING###### ########## ######_#######jsonb##################jsonb########

#9.52 SQL/JSON Query Functions

######### ## #
####### JSON_EXISTS (*context_item*, *path_expression* [PASSING { *value* AS *varname* } [, ...]] [{ TRUE | FALSE | UNKNOWN | ERROR } ON ERROR]) → boolean • ######_#######SQL/JSON##_#################################### ###### • ######ON########_######################### ############################### ### ########include# boolean##############SQL null###UNKNOWN######## ON### ####################boolean######## ## JSON_EXISTS(jsonb '{"key1": [1,2,3]}', 'strict $.key1[*] ? (@ > $x)' PASSING 2 AS x) → t JSON_EXISTS(jsonb '{"a": [1,2,3]}', 'lax $.a[5]' ERROR ON ERROR) → f JSON_EXISTS(jsonb '{"a": [1,2,3]}', 'strict $.a[5]' ERROR ON ERROR) → ERROR: jsonpath array subscript is out of bounds
####### JSON_QUERY (*context_item*, *path_expression* [PASSING { *value* AS *varname* } [, ...]] [RETURNING *data_type* [FORMAT JSON [ENCODING UTF8]]] [{ WITHOUT | WITH { CONDITIONAL | [UNCONDITIONAL] } } [ARRAY] WRAPPER] [{ KEEP | OMIT } QUOTES [ON SCALAR STRING]] [{ ERROR | NULL | EMPTY { [ARRAY] | OBJECT } | DEFAULT *expression* } ON EMPTY] [{ ERROR | NULL | EMPTY { [ARRAY] | OBJECT } | DEFAULT *expression* } ON ERROR]) → jsonb • ######SQL/JSON##_########_################

```
########
      ##
      #
```

- ###################jsonb###########RETURNING#########################
  ##

- ############################WITH##########make############# ######JSON#
  ##### ##############WITHOUT########################## WITH############
  ###WITH UNCONDITIONAL################ ########################### ####
  #######################WITH CONDITIONAL########### WITHOUT###########
  ###########################

- #####################################JSON####### ####KEEP QUOTES#######
  ######## ###OMIT QUOTES################## ######JSON########WITH#######
  ########OMIT QUOTES########

- ######ON EMPTY#####_########################## ON#######_##########
  ##RETURNING###############_################ON EMPTY##################
  ########

- ######ON EMPTY#ON## SQL include####
  ######null######EMPTY[##]###########EMPTY######################jsonb##
  #RETURNING################ ON EMPTY###ON#########################SQL
  NULL####

```
      ##
      JSON_QUERY(jsonb '[1,[2,3],null]', 'lax $[*][$off]' PASSING
      1 AS off WITH CONDITIONAL WRAPPER) → [3]
      JSON_QUERY(jsonb '{"a": "[1, 2]"}', 'lax $.a' OMIT QUOTES)
      → [1, 2]
      JSON_QUERY(jsonb '{"a": "[1, 2]"}', 'lax $.a' RETURNING
      int[] OMIT QUOTES ERROR ON ERROR) →


      ERROR:  malformed array literal: "[1, 2]"
      DETAIL:  Missing "]" after array dimensions.
```

```
#######

JSON_VALUE (
      context_item, path_expression
      [ PASSING { value AS varname } [, ...]]
      [ RETURNING data_type ]
      [ { ERROR | NULL | DEFAULT expression } ON EMPTY ]
      [ { ERROR | NULL | DEFAULT expression } ON ERROR ]) → text
```

- ######SQL/JSON##_########_################

- ######JSON_VALUE)################SQL/JSON#################### #######
  ############## ################################JSON_###########

- ## ####RETURNING#########
  ##################

- ######ON#####ON EMPTY##########JSON_##################################
  #############

- #########_VALUE##############JSON##JSON_####OMIT QUOTES##############
  ###########

######## ## #
JSON_VALUE(jsonb '"123.45"', '$' RETURNING float) → 123.45 JSON_VALUE(jsonb '"03:04 2015-02-01"', '$.datetime("HH24:MI YYYY-MM-DD")' RETURNING date) → 2015-02-01 JSON_VALUE(jsonb '[1,2]', 'strict $[$off]' PASSING 1 as off) → 2 JSON_VALUE(jsonb '[1,2]', 'strict $[*]' DEFAULT 9 ON ERROR) → 9

##

###########_#######jsonb#####jsonb########################## ####### ########################### ###############ON############### #

##

#####*JSON_*##SQLnull#######_VALUE)#JSON##### ###JSON_###)# JSONnull######### null

9.16.4. JSON_TABLE

#####JSON_TABLE#JSON#############################SQL/JSON##### #### #SELECT#UPDATE#DELETE#FROM######### ###MERGE##################

#####JSON#########JSON_TABLE#JSON## ###### ###########SQL/JSON############################

####################JSON_TABLE##################COLUMNS####### ######### JSON############################## ########################

##########################JSON#####NESTED PATH############ #NESTED PATH###########################1################### ##########COLUMNS#### #COLUMNS############ NESTED COLUMNS#################COLUMNS############ ########################### #####NESTED PATH################################## ##### ##########NESTED PATH##################################UNION######### ###

#####JSON_TABLE############################JSON########################### ############

################

```
JSON_TABLE (
    context_item, path_expression [ AS json_path_name ] [ PASSING { value
 AS varname } [, ...] ]
    COLUMNS ( json_table_column [, ...] )
    [ { ERROR | EMPTY [ARRAY]} ON ERROR ]
)
```

where *json_table_column* is:

```
  name FOR ORDINALITY
  | name type
        [ FORMAT JSON [ENCODING UTF8]]
        [ PATH path_expression ]
        [ { WITHOUT | WITH { CONDITIONAL | [UNCONDITIONAL] } } [ ARRAY ]
 WRAPPER ]
        [ { KEEP | OMIT } QUOTES [ ON SCALAR STRING ] ]
        [ { ERROR | NULL | EMPTY { [ARRAY] | OBJECT } | DEFAULT expression
 } ON EMPTY ]
        [ { ERROR | NULL | EMPTY { [ARRAY] | OBJECT } | DEFAULT expression
 } ON ERROR ]
  | name type EXISTS [ PATH path_expression ]
        [ { ERROR | TRUE | FALSE | UNKNOWN } ON ERROR ]
  | NESTED [ PATH ] path_expression [ AS json_path_name ] COLUMNS
 ( json_table_column [, ...] )
```

#################################

context_item, *path_expression* [AS *json_path_name*] [PASSING { *value* AS *varname* } [, ...]]

############_#########################_#########SQL/JSON#####*JSON_##_#*##_######### ##########PASSING######_#################### ##

COLUMNS (*json_table_column* [, ...])

#######################COLUMNS## ####################JSON##############SQL/JSON######## *json_table_column*###############

name FOR ORDINALITY

######1######################### #NESTED PATH######################### ############

name type [FORMAT JSON [ENCODING UTF8]] [PATH *path_expression*]

##########*type*################*path_expression*########SQL/JSON#### ####

######FORMAT JSON###########json#################### FORMAT JSON# ######*type*#bpchar#bytea#character varying#name#json#jsonb#text# ############################

############WRAPPER#QUOTES######################### ############ #json#########QUOTES######FORMAT JSON##################

############ON EMPTY#ON ERROR######JSON#######################JSON ##################################### #########NULL###

##
#############JSON_VALUE###JSON_QUERY########## ############ ###########FORMAT JSON#WRAPPER####QUOTES###############

name type EXISTS [PATH *path_expression*]

##########*type*################*path_expression*##################

##########PATH################################

##########*type*#boolean###############

###########ON ERROR#####JSON##################SQL/JSON########### ################################## ##########FALSE#####

##

##############JSON_EXISTS############

NESTED [PATH] *path_expression* [AS *json_path_name*] COLUMNS (*json_table_column* [, ...])

######################SQL/JSON#####COLUMNS#######1################ ####SQL/JSON####### COLUMNS###*json_table_column*#####COLUMNS###### ######

######NESTED PATH######## ########NESTED PATH#################### ############## #####SQL#####JSON_TABLE#####################JSON## ####################

##

########*json_table_column*###########PATH############$.*name*#### #### ####*name*##########

AS *json_path_name*

###########*json_path_name*######*path_expression*########### ######## ###################

{ ERROR | EMPTY } ON ERROR

###########ON ERROR#####*path_expression*########################## # ##########ERROR##############0#############EMPTY###### ############### ################### ########ON ERROR################################

Examples

################ JSON ########

```
CREATE TABLE my_films ( js jsonb );

INSERT INTO my_films VALUES (
'{ "favorites" : [
   { "kind" : "comedy", "films" : [
     { "title" : "Bananas",
       "director" : "Woody Allen"},
     { "title" : "The Dinner Game",
       "director" : "Francis Veber" } ] },
   { "kind" : "horror", "films" : [
     { "title" : "Psycho",
       "director" : "Alfred Hitchcock" } ] },
   { "kind" : "thriller", "films" : [
     { "title" : "Vertigo",
       "director" : "Alfred Hitchcock" } ] },
   { "kind" : "drama", "films" : [
     { "title" : "Yojimbo",
       "director" : "Akira Kurosawa" } ] }
  ] }');
```

############JSON_TABLE######my_films######JSON#########JSON#####
##kind#title####director#####################

```
SELECT jt.* FROM
 my_films,
 JSON_TABLE (js, '$.favorites[*]' COLUMNS (
   id FOR ORDINALITY,
   kind text PATH '$.kind',
   title text PATH '$.films[*].title' WITH WRAPPER,
   director text PATH '$.films[*].director' WITH WRAPPER)) AS jt;
```

```
 id |   kind   |             title              |             director
----+----------+--------------------------------
+----------------------------------
  1 | comedy   | ["Bananas", "The Dinner Game"] | ["Woody Allen", "Francis
 Veber"]
  2 | horror   | ["Psycho"]                     | ["Alfred Hitchcock"]
  3 | thriller | ["Vertigo"]                    | ["Alfred Hitchcock"]
  4 | drama    | ["Yojimbo"]                    | ["Akira Kurosawa"]
(4 rows)
```

##############################JSON##############PASSING#####################
############

```
SELECT jt.* FROM
 my_films,
 JSON_TABLE (js, '$.favorites[*] ? (@.films[*].director == $filter)'
   PASSING 'Alfred Hitchcock' AS filter, 'Vertigo' AS filter2
     COLUMNS (
     id FOR ORDINALITY,
     kind text PATH '$.kind',
     title text FORMAT JSON PATH '$.films[*].title' OMIT QUOTES,
     director text PATH '$.films[*].director' KEEP QUOTES)) AS jt;
```

```
 id |   kind   |  title  |      director
----+----------+---------+--------------------
  1 | horror   | Psycho  | "Alfred Hitchcock"
  2 | thriller | Vertigo | "Alfred Hitchcock"
(2 rows)
```

##############################NESTED PATH####################id#kind##########
##########

```
SELECT jt.* FROM
 my_films,
 JSON_TABLE ( js, '$.favorites[*] ? (@.films[*].director == $filter)'
   PASSING 'Alfred Hitchcock' AS filter
   COLUMNS (
    id FOR ORDINALITY,
    kind text PATH '$.kind',
    NESTED PATH '$.films[*]' COLUMNS (
      title text FORMAT JSON PATH '$.title' OMIT QUOTES,
      director text PATH '$.director' KEEP QUOTES))) AS jt;
```

```
 id |   kind   |  title  |      director
----+----------+---------+--------------------
  1 | horror   | Psycho  | "Alfred Hitchcock"
  2 | thriller | Vertigo | "Alfred Hitchcock"
(2 rows)
```

```
###############################################
```

```
SELECT jt.* FROM
 my_films,
 JSON_TABLE ( js, '$.favorites[*]'
   COLUMNS (
    id FOR ORDINALITY,
    kind text PATH '$.kind',
    NESTED PATH '$.films[*]' COLUMNS (
      title text FORMAT JSON PATH '$.title' OMIT QUOTES,
      director text PATH '$.director' KEEP QUOTES))) AS jt;
```

```
 id |   kind   |      title      |      director
----+----------+-----------------+--------------------
  1 | comedy   | Bananas         | "Woody Allen"
  1 | comedy   | The Dinner Game | "Francis Veber"
  2 | horror   | Psycho          | "Alfred Hitchcock"
  3 | thriller | Vertigo         | "Alfred Hitchcock"
  4 | drama    | Yojimbo         | "Akira Kurosawa"
(5 rows)
```

```
############JSON#############################                    ####NESTED#
#$.movies[*]#$.books[*]###SIBLING   JOIN##NESTED#####FOR   ORDINALITY#
##movie_id#book_id####author_id############
```

```
SELECT * FROM JSON_TABLE (
'{"favorites":
    {"movies":
      [{"name": "One", "director": "John Doe"},
       {"name": "Two", "director": "Don Joe"}],
     "books":
      [{"name": "Mystery", "authors": [{"name": "Brown Dan"}]},
       {"name": "Wonder", "authors": [{"name": "Jun Murakami"},
 {"name":"Craig Doe"}]}]
}}'::json, '$.favorites[*]'
COLUMNS (
  user_id FOR ORDINALITY,
  NESTED '$.movies[*]'
    COLUMNS (
    movie_id FOR ORDINALITY,
    mname text PATH '$.name',
    director text),
  NESTED '$.books[*]'
    COLUMNS (
      book_id FOR ORDINALITY,
      bname text PATH '$.name',
      NESTED '$.authors[*]'
        COLUMNS (
          author_id FOR ORDINALITY,
          author_name text PATH '$.name'))));
```

```
 user_id | movie_id | mname | director | book_id |  bname  | author_id |
 author_name
---------+----------+-------+----------+---------+---------+-----------
+--------------
       1 |        1 | One   | John Doe |         |         |           |
       1 |        2 | Two   | Don Joe  |         |         |           |
       1 |          |       |          |       1 | Mystery |         1 |
 Brown Dan
       1 |          |       |          |       2 | Wonder  |         1 |
 Jun Murakami
```

```
      1 |          |       |          |       2 | Wonder  |         2 |
 Craig Doe
(5 rows)
```

9.17.

################################ ##
########################CREATE SEQUENCE######## ########################
################ # 9.53###
#########

#9.53

##
nextval(regclass) → bigint   ##################### ######################nextval##################   ################### ##############################nextval#####1####   ######### ################CREATE SEQUENCE##################   ################USAGE####UPDATE#######
setval(regclass, bigint [, boolean]) → bigint   ############################is_called####### 2################   ####last_value################is_called#####true##########   #nextval######################## currval################### 3####   ##########is_called#true####false####### true#2################   ##### false###########nextval###############nextval#############   ## #######currval##########################   SELECT setval('myseq', 42); ##nextval#43###   SELECT setval('myseq', 42, true); ##   SELECT setval('myseq', 42, false); ##nextval#42###   setval#######2############   ##########UPDATE#######
currval(regclass) → bigint   ######################nextval############## ########nextval######   ############## ##########################nextval##################   #######   ##########USAGE####SELECT#######
lastval() → bigint   ######################nextval############## ##############nextval#   ############################currval##### #######nextval########   #lastval###########   ##########USAGE####SELECT#######

##

##nextval#####################
############################ ######################################
################################## ########################## ##
#ON CONFLICT####INSERT#########################nextval###########
####ON CONFLICT############################ ####PostgreSQL###########
############################

####setval###

> nextval#setval##
> ##
> ### ## ##########
> ##########################nextval##########################

####################regclass#############pg_class#######################
OID### ##########OID##########regclass########################### ###8.19###
#####

9.18.

####PostgreSQL######SQL###############

> ###
>
> ##

> ##
>
> COALESCE#GREATEST#LEAST###########################VARIADIC#######
> #######

9.18.1. CASE

SQL#CASE##############if/else##############

```
CASE WHEN condition THEN result
     [WHEN ...]
     [ELSE result]
END
```

CASE#######################*#####condition*##boolean########## ##condition####
######CASE#####condition###*result*#########CASE############ ##condition########
#######WHEN############### WHEN#*condition*#1########CASE####ELSE##*result*##
ELSE########condition##########NULL###

##########

```
SELECT * FROM test;

 a
---
 1
 2
 3


SELECT a,
       CASE WHEN a=1 THEN 'one'
            WHEN a=2 THEN 'two'
            ELSE 'other'
       END
    FROM test;

 a | case
```

```
---+-------
 1 | one
 2 | two
 3 | other
```

###*result*########################### ###10.5#########

###################CASE###############

```
CASE expression
    WHEN value THEN result
    [WHEN ...]
    [ELSE result]
END
```

###*expression*#######################WHEN######*value*######## ############ ##ELSE##*result*#####NULL######## ###C###switch#######

######CASE###################

```
SELECT a,
       CASE a WHEN 1 THEN 'one'
              WHEN 2 THEN 'two'
              ELSE 'other'
       END
    FROM test;

 a | case
---+-------
 1 | one
 2 | two
 3 | other
```

CASE############################## #######0###############

```
SELECT ... WHERE CASE WHEN x <> 0 THEN y/x > 1.5 ELSE false END;
```

> **##**
>
> 4.2.14#######################################CASE#################### ######### ######1/0###############CASE################################ ########

9.18.2. COALESCE

```
COALESCE(value [, ...])
```

COALESCE####NULL################ ######NULL######NULL################### ##NULL####################################

```
SELECT COALESCE(description, short_description, '(none)') ...
```

###description#NULL############ #######NULL#####short_description# NULL############ #######short_description#NULL#####(none)######

#######################################10.5##########

CASE####COALESCE############################NULL######################## ## ##SQL##########################NVL###IFNULL############

9.18.3. NULLIF

```
NULLIF(value1, value2)
```

NULLIF####*value1*#*value2*######NULL###### ######*value1*##### ######## #COALESCE############

```
SELECT NULLIF(value, '(none)') ...
```

#####value#(none)###NULL##### #####value#####

2################### #########*value1 = value2*################=######### #######

########################## #########=####################2################## #### ####NULLIF(1, 2.2)#numeric#########integer = numeric######numeric = numeric#########

9.18.4. GREATEST###LEAST

```
GREATEST(value [, ...])
```

```
LEAST(value [, ...])
```

GREATEST#LEAST############################## ############################## ##########10.5##########

#######NULL######## #####NULL##############NULL###### ####SQL######### ################NULL####NULL###### ########################

9.19.

9.54################### ######## 9.1################### ################### ########B-tree## ######################## ################## 2###

#9.54 #####

#
anyarray @> anyarray → boolean   ######2##############2############################## ############   ##ARRAY[1]#ARRAY[1,1]########################   ARRAY[1,4,3] @> ARRAY[3,1,3] → t
anyarray <@ anyarray → boolean   ######2##########   ARRAY[2,2,7] <@ ARRAY[1,7,4,2,6] → t
anyarray && anyarray → boolean   ##############################   ARRAY[1,4,3] && ARRAY[2,1] → t
anycompatiblearray || anycompatiblearray → anycompatiblearray

2########## null## ##### ################################2########### ######################## ####10.5#### ARRAY[1,2,3] || ARRAY[4,5,6,7] → {1,2,3,4,5,6,7} ARRAY[1,2,3] || ARRAY[[4,5,6],[7,8,9.9]] → {{1,2,3},{4,5,6}, {7,8,9.9}}
anycompatible || anycompatiblearray → anycompatiblearray ################################## 3 || ARRAY[4,5,6] → {3,4,5,6}
anycompatiblearray || anycompatible → anycompatiblearray ################################## ARRAY[4,5,6] || 7 → {4,5,6,7}

#############8.15######### ####################################11.2#########

9.55################## ################8.15#########

#9.55

array_append (anycompatiblearray, anycompatible) → anycompatiblearray ################anycompatiblearray || anycompatible######### array_append(ARRAY[1,2], 3) → {1,2,3}
array_cat (anycompatiblearray, anycompatiblearray) → anycompatiblearray 2###########anycompatiblearray || anycompatiblearray######### array_cat(ARRAY[1,2,3], ARRAY[4,5]) → {1,2,3,4,5}
array_dims (anyarray) → text ################## array_dims(ARRAY[[1,2,3], [4,5,6]]) → [1:2][1:3]
array_fill (anyelement, integer[] [, integer[]]) → anyarray ################2######################### ######3#################### ########1#### array_fill(11, ARRAY[2,3]) → {{11,11,11},{11,11,11}} array_fill(7, ARRAY[3], ARRAY[2]) → [2:4]={7,7,7}
array_length (anyarray, integer) → integer ################### #####################0####NULL######## array_length(array[1,2,3], 1) → 3 array_length(array[]::int[], 1) → NULL array_length(array['text'], 2) → NULL
array_lower (anyarray, integer) → integer ###################

##
##
#

array_lower('[0:2]={1,2,3}'::integer[], 1) → 0

array_ndims (anyarray) → integer
###########
array_ndims(ARRAY[[1,2,3], [4,5,6]]) → 2

array_position (anycompatiblearray, anycompatible [, integer]) → integer
2##########################NULL###### 3#######################################NULL#########
###IS NOT DISTINCT FROM###########NULL#########
array_position(ARRAY['sun', 'mon', 'tue', 'wed', 'thu', 'fri', 'sat'], 'mon') → 2

array_positions (anycompatiblearray, anycompatible) → integer[]
2#################################NULL###### 3#######################
###IS NOT DISTINCT FROM###########NULL######
###NULL#####NULL######
array_positions(ARRAY['A','A','B','A'], 'A') → {1,2,4}

array_prepend (anycompatible, anycompatiblearray) → anycompatiblearray
###############anycompatible || anycompatiblearray#########
array_prepend(1, ARRAY[2,3]) → {1,2,3}

array_remove (anycompatiblearray, anycompatible) → anycompatiblearray
####################### ### ###IS NOT DISTINCT FROM###
########NULL############
array_remove(ARRAY[1,2,3,2], 2) → {1,3}

array_replace (anycompatiblearray, anycompatible, anycompatible) → anycompatiblearray
2###########3############
array_replace(ARRAY[1,2,5,4], 5, 3) → {1,2,3,4}

array_sample (*array* anyarray, *n* integer) → anyarray
array##########*n*############## *n*#*array*####################
array########item#####################
array_sample(ARRAY[1,2,3,4,5,6], 3) → {2,6,1}
array_sample(ARRAY[[1,2],[3,4],[5,6]], 2) → {{5,6},{1,2}}

array_shuffle (anyarray) → anyarray
###1#################
array_shuffle(ARRAY[[1,2],[3,4],[5,6]]) → {{5,6},{1,2},{3,4}}

array_to_string (*array* anyarray, *delimiter* text [, *null_string* text]) → text
##############*delimiter*############# NULL###*null_string*#####
##NULL############################ string_to_array#########
array_to_string(ARRAY[1, 2, 3, NULL, 5], ',', '*') → 1,2,3,*,5

array_upper (anyarray, integer) → integer ################## array_upper(ARRAY[1,8,3,7], 1) → 4
cardinality (anyarray) → integer ##################0###### cardinality(ARRAY[[1,2],[3,4]]) → 4
trim_array (*array* anyarray, *n* integer) → anyarray ###*n*############### ######################## trim_array(ARRAY[1,2,3,4,5,6], 2) → {1,2,3,4}
unnest (anyarray) → setof anyelement ############# ################ unnest(ARRAY[1,2]) → 1 2 unnest(ARRAY[['foo','bar'],['baz','quux']]) → foo bar baz quux
unnest (anyarray, anyarray [, ...]) → setof anyelement, anyelement [, ...] ################################ ###################NULL####### ## ######FROM#########7.2.1.4######### select * from unnest(ARRAY[1,2], ARRAY['foo','bar','baz']) as x(a,b) → a | b ---+----- 1 | foo 2 | bar | baz

##########array_agg#####9.21#########

9.20.

##########8.17#######

9.56#################### # 9.57######(multirange)################## ######## 9.1### ######################### ### ########## #####################

#9.56

```
###
      ##
      #
```

```
anyrange @> anyrange → boolean
      ######2#########
      int4range(2,4) @> int4range(2,3) → t
```

```
anyrange @> anyelement → boolean
      ##############
      '[2011-01-01,2011-03-01)'::tsrange @>
      '2011-01-10'::timestamp → t
```

```
anyrange <@ anyrange → boolean
      ######2########
      int4range(2,4) <@ int4range(1,7) → t
```

```
anyelement <@ anyrange → boolean
      #############
      42 <@ int4range(1,7) → f
```

```
anyrange && anyrange → boolean
      #########################
      int8range(3,7) && int8range(4,12) → t
```

```
anyrange << anyrange → boolean
      ########2######
      int8range(1,10) << int8range(100,110) → t
```

```
anyrange >> anyrange → boolean
      ########2######
      int8range(50,60) >> int8range(20,30) → t
```

```
anyrange &< anyrange → boolean
      ######2#############
      int8range(1,20) &< int8range(18,20) → t
```

```
anyrange &> anyrange → boolean
      ######2#############
      int8range(7,20) &> int8range(5,10) → t
```

```
anyrange -|- anyrange → boolean
      ###########
      numrange(1.1,2.2) -|- numrange(2.2,3.3) → t
```

```
anyrange + anyrange → anyrange
      ##########################################################
      #range_merge()##########
      numrange(5,15) + numrange(10,20) → [5,20)
```

```
anyrange * anyrange → anyrange
      #############
      int8range(5,15) * int8range(10,20) → [10,15)
```

```
anyrange - anyrange → anyrange
      ########### ################2########################
```

```
###
      ##
      #
```

```
      int8range(5,15) - int8range(10,20) → [5,10)
```

#9.57

```
###
      ##
      #
```

```
anymultirange @> anymultirange → boolean
      ########2##########
      '{[2,4)}'::int4multirange @> '{[2,3)}'::int4multirange → t
```

```
anymultirange @> anyrange → boolean
      ##############
      '{[2,4)}'::int4multirange @> int4range(2,3) → t
```

```
anymultirange @> anyelement → boolean
      ##############
      '{[2011-01-01,2011-03-01)}'::tsmultirange @>
      '2011-01-10'::timestamp → t
```

```
anyrange @> anymultirange → boolean
      ##############
      '[2,4)'::int4range @> '{[2,3)}'::int4multirange → t
```

```
anymultirange <@ anymultirange → boolean
      ########2##########
      '{[2,4)}'::int4multirange <@ '{[1,7)}'::int4multirange → t
```

```
anymultirange <@ anyrange → boolean
      ###############
      '{[2,4)}'::int4multirange <@ int4range(1,7) → t
```

```
anyrange <@ anymultirange → boolean
      ###############
      int4range(2,4) <@ '{[1,7)}'::int4multirange → t
```

```
anyelement <@ anymultirange → boolean
      ###############
      4 <@ '{[1,7)}'::int4multirange → t
```

```
anymultirange && anymultirange → boolean
      ############################
      '{[3,7)}'::int8multirange && '{[4,12)}'::int8multirange → t
```

```
anymultirange && anyrange → boolean
      ##############
      '{[3,7)}'::int8multirange && int8range(4,12) → t
```

```
anyrange && anymultirange → boolean
      ##############
      int8range(3,7) && '{[4,12)}'::int8multirange → t
```

```
anymultirange << anymultirange → boolean
```

```
###
      ##
      #
      ###########2######
      '{[1,10)}'::int8multirange << '{[100,110)}'::int8multirange
      → t
```

```
anymultirange << anyrange → boolean
      #############
      '{[1,10)}'::int8multirange << int8range(100,110) → t
```

```
anyrange << anymultirange → boolean
      #############
      int8range(1,10) << '{[100,110)}'::int8multirange → t
```

```
anymultirange >> anymultirange → boolean
      ###########2######
      '{[50,60)}'::int8multirange >> '{[20,30)}'::int8multirange
      → t
```

```
anymultirange >> anyrange → boolean
      #############
      '{[50,60)}'::int8multirange >> int8range(20,30) → t
```

```
anyrange >> anymultirange → boolean
      #############
      int8range(50,60) >> '{[20,30)}'::int8multirange → t
```

```
anymultirange &< anymultirange → boolean
      #######2#############
      '{[1,20)}'::int8multirange &< '{[18,20)}'::int8multirange →
      t
```

```
anymultirange &< anyrange → boolean
      ##################
      '{[1,20)}'::int8multirange &< int8range(18,20) → t
```

```
anyrange &< anymultirange → boolean
      ##################
      int8range(1,20) &< '{[18,20)}'::int8multirange → t
```

```
anymultirange &> anymultirange → boolean
      #######2#############
      '{[7,20)}'::int8multirange &> '{[5,10)}'::int8multirange → t
```

```
anymultirange &> anyrange → boolean
      ##################
      '{[7,20)}'::int8multirange &> int8range(5,10) → t
```

```
anyrange &> anymultirange → boolean
      ##################
      int8range(7,20) &> '{[5,10)}'::int8multirange → t
```

```
anymultirange -|- anymultirange → boolean
      ################
```

'{[1.1,2.2)}'::nummultirange -|- '{[2.2,3.3)}'::nummultirange → t
anymultirange -|- anyrange → boolean ############### '{[1.1,2.2)}'::nummultirange -|- numrange(2.2,3.3) → t
anyrange -|- anymultirange → boolean ############### numrange(1.1,2.2) -|- '{[2.2,3.3)}'::nummultirange → t
anymultirange + anymultirange → anymultirange ############ ############################## '{[5,10)}'::nummultirange + '{[15,20)}'::nummultirange → {[5,10), [15,20)}
anymultirange * anymultirange → anymultirange ################# '{[5,15)}'::int8multirange * '{[10,20)}'::int8multirange → {[10,15)}
anymultirange - anymultirange → anymultirange ############## '{[5,20)}'::int8multirange - '{[10,15)}'::int8multirange → {[5,10), [15,20)}

###

############################## ############# ################ ################
####### ################ ###########################

################2######################## ################## ################
############################## ##
################

9.58################# # 9.59###################

#9.58

lower (anyrange) → anyelement #########################NULL###### lower(numrange(1.1,2.2)) → 1.1
upper (anyrange) → anyelement #########################NULL###### upper(numrange(1.1,2.2)) → 2.2
isempty (anyrange) → boolean ###### isempty(numrange(1.1,2.2)) → f

```
##
      ##
      #
```

```
lower_inc ( anyrange ) → boolean
      #############
      lower_inc(numrange(1.1,2.2)) → t
```

```
upper_inc ( anyrange ) → boolean
      #############
      upper_inc(numrange(1.1,2.2)) → f
```

```
lower_inf ( anyrange ) → boolean
      #########? #-Infinity###########
      lower_inf('(,)'::daterange) → t
```

```
upper_inf ( anyrange ) → boolean
      #########? #Infinity###########
      upper_inf('(,)'::daterange) → t
```

```
range_merge ( anyrange, anyrange ) → anyrange
      ########################
      range_merge('[1,2)'::int4range, '[3,4)'::int4range) → [1,4)
```

#9.59

```
##
      ##
      #
```

```
lower ( anymultirange ) → anyelement
      ##############################NULL#######
      lower('{[1.1,2.2)}'::nummultirange) → 1.1
```

```
upper ( anymultirange ) → anyelement
      ##############################NULL#######
      upper('{[1.1,2.2)}'::nummultirange) → 2.2
```

```
isempty ( anymultirange ) → boolean
      ########
      isempty('{[1.1,2.2)}'::nummultirange) → f
```

```
lower_inc ( anymultirange ) → boolean
      ##############
      lower_inc('{[1.1,2.2)}'::nummultirange) → t
```

```
upper_inc ( anymultirange ) → boolean
      ##############
      upper_inc('{[1.1,2.2)}'::nummultirange) → f
```

```
lower_inf ( anymultirange ) → boolean
      ###########? #-Infinity###########
      lower_inf('{(,)}'::datemultirange) → t
```

```
upper_inf ( anymultirange ) → boolean
      ###########? #Infinity###########
      upper_inf('{(,)}'::datemultirange) → t
```

#
range_merge (anymultirange) → anyrange ########################## range_merge('{[1,2), [3,4)}'::int4multirange) → [1,4)
multirange (anyrange) → anymultirange ##################### multirange('[1,2)'::int4range) → {[1,2)}
unnest (anymultirange) → setof anyrange ####################### unnest('{[1,2), [3,4)}'::int4multirange) → [1,2) [3,4)

lower_inc#upper_inc#lower_inf#upper_inf###################

9.21.

######################## # 9.60################ 9.61############ # 9.62######### ############## 9.63############################## # 9.64########################## ## ################4.2.7######### ##############2.7########

##################################

###################ORDER BY########4.2.7##############################

#9.60

## ##	#####
any_value (anyelement) → *same as input type* NULL#################	#
array_agg (anynonarray ORDER BY input_sort_columns) → anyarray NULL########################	#
array_agg (anyarray ORDER BY input_sort_columns) → anyarray ###############1########## ######################NULL####### #####	#
avg (smallint) → numeric avg (integer) → numeric avg (bigint) → numeric avg (numeric) → numeric avg (real) → double precision avg (double precision) → double precision avg (interval) → interval #####NULL##################	#
bit_and (smallint) → smallint	#

## ##	#####
bit_and (integer) → integer bit_and (bigint) → bigint bit_and (bit) → bit ####NULL###############	
bit_or (smallint) → smallint bit_or (integer) → integer bit_or (bigint) → bigint bit_or (bit) → bit ####NULL###############	#
bit_xor (smallint) → smallint bit_xor (integer) → integer bit_xor (bigint) → bigint bit_xor (bit) → bit #####NULL#################### ############################## ##	#
bool_and (boolean) → boolean #########################	#
bool_or (boolean) → boolean ##########################	#
count (*) → bigint #########	#
count ("any") → bigint #NULL##########	#
every (boolean) → boolean ###SQL###bool_and######	#
json_agg (anyelement ORDER BY input_sort_columns) → json jsonb_agg (anyelement ORDER BY input_sort_columns) → jsonb NULL###############JSON######## ###to_json####to_jsonb# JSON#######	#
json_objectagg ([{ *key_expression* { VALUE | ':' } *value_expression* }] [{ NULL | ABSENT } ON NULL] [{ WITH | WITHOUT } UNIQUE [KEYS]] [RETURNING *data_type* [FORMAT JSON [ENCODING UTF8]]]) JSON_##############################1##*key_expression*#1# #*value_expression*########## SELECT json_objectagg(k:v) FROM (VALUES ('a'::text,current_date),('b',current_date + 1)) AS t(k,v) → { "a" : "2022-05-10", "b" : "2022-05-11" }	#
json_object_agg (*key* "any", *value* "any" ORDER BY input_sort_columns) → json jsonb_object_agg (*key* "any", *value* "any" ORDER BY input_sort_columns) → jsonb ##########JSON############ ####################to_json### #to_jsonb############ ##NULL###########NULL#######	#
json_object_agg_strict (*key* "any", *value* "any") → json	#

## ##	#####
jsonb_object_agg_strict (*key* "any", *value* "any") → jsonb ##########JSON########### ##################to_json### #to_jsonb########### *key*#NULL####### *value*#NULL######## #########	
json_object_agg_unique (*key* "any", *value* "any") → json jsonb_object_agg_unique (*key* "any", *value* "any") → jsonb ##########JSON########### ##################to_json### #to_jsonb########### ##NULL###########NULL####### ####### ############	#
json_arrayagg ([*value_expression*] [ORDER BY *sort_expression*] [{ NULL | ABSENT } ON NULL] [RETURNING *data_type* [FORMAT JSON [ENCODING UTF8]]]) JSON_###########################1##*value*_############## ABSENT ON NULL##########NULL######## ORDER BY########### #################### SELECT json_arrayagg(v) FROM (VALUES(2),(1)) t(v) → [2, 1]	#
json_object_agg_unique_strict (*key* "any", *value* "any") → json jsonb_object_agg_unique_strict (*key* "any", *value* "any") → jsonb ##########JSON########### ##################to_json### #to_jsonb########### *key*#NULL####### *value*#NULL######## ######### ###################	#
max (*see text*) → *same as input type* #NULL########### ############## #inet#interval#money#oid#pg_lsn#tid#xid8################ #######	#
min (*see text*) → *same as input type* #NULL########### ############## #inet#interval#money#oid#pg_lsn#tid#xid8################ #######	#
range_agg (*value* anyrange) → anymultirange range_agg (*value* anymultirange) → anymultirange #NULL##########	#
range_intersect_agg (*value* anyrange) → anyrange range_intersect_agg (*value* anymultirange) → anymultirange #NULL############	#
json_agg_strict (anyelement) → json jsonb_agg_strict (anyelement) → jsonb NULL###############JSON######## ##to_JSON###to_JSON#### jsonb#######	#
string_agg (*value* text, *delimiter* text) → text string_agg (*value* bytea, *delimiter* bytea ORDER BY input_sort_columns) → bytea #NULL############### ###########*delimiter*#######NULL#### ########	#

## ##	#####
sum(smallint) → bigint sum(integer) → bigint sum(bigint) → numeric sum(numeric) → numeric sum(real) → real sum(double precision) → double precision sum(interval) → interval sum(money) → money #NULL############	#
xmlagg(xml ORDER BY input_sort_columns) → xml #NULL#XML##########9.15.1.8####	#

#######count######1###########NULL############### #########sum############ ######NULL######array_agg####################NULL###### #######NULL##### ###########coalesce############

#array_agg#json_agg#jsonb_agg#json_agg_strict#jsonb_agg_strict#json_object_agg#js ###xmlagg## ################### ##4.2.7###################ORDER BY## ##############

```
SELECT xmlagg(x) FROM (SELECT x FROM test ORDER BY y DESC) AS tab;
```

#################

> ##
>
> bool_and#bool_or########SQL#####every#any###some###### PostgreSQL #every#########any#####some######### any#some#################
>
> ```
> SELECT b1 = ANY((SELECT b2 FROM t2 ...)) FROM t1 ...;
> ```
>
> ######################ANY### ## ###########################

> ##
>
> ##SQL#########################count##################################
>
> ```
> SELECT count(*) FROM sometable;
> ```
>
> ############################PostgreSQL################################## ##########

################### 9.61###### ## *numeric_type*############### ##smallint#integer#bigint#numeric#real#double precision############# #

#######N###########NULL############ ############N#0################NULL#####
###

#9.61

## ##	#####
corr(*Y* double precision, *X* double precision) → double precision ##########	#
covar_pop(*Y* double precision, *X* double precision) → double precision ##########	#
covar_samp(*Y* double precision, *X* double precision) → double precision ############	#
regr_avgx(*Y* double precision, *X* double precision) → double precision ############## sum(*X*)/*N*.	#
regr_avgy(*Y* double precision, *X* double precision) → double precision ############## sum(*Y*)/*N*.	#
regr_count(*Y* double precision, *X* double precision) → bigint ######NULL###########	#
regr_intercept(*Y* double precision, *X* double precision) → double precision (*X*, *Y*)#######################Y########	#
regr_r2(*Y* double precision, *X* double precision) → double precision #############	#
regr_slope(*Y* double precision, *X* double precision) → double precision (*X*, *Y*)################################	#
regr_sxx(*Y* double precision, *X* double precision) → double precision ##########sum(*X*^2) - sum(*X*)^2/*N#######	#
regr_sxy(*Y* double precision, *X* double precision) → double precision ###############sum(*X**Y) - sum(*X*) * sum(*Y*)/*N#######	#
regr_syy(*Y* double precision, *X* double precision) → double precision ##########sum(*Y*^2) - sum(*Y*)^2/*N#######	#
stddev(*numeric_type*) → ###double precision####real#### #double precision#####numeric ###stddev_samp#########	#
stddev_pop(*numeric_type*) → ###double precision####real#### #double precision#####numeric	#

## ##	#####
###############	
stddev_samp(*numeric_type*) → ###double precision####real#### #double precision#####numeric ################	#
variance(*numeric_type*) → ###double precision####real#### #double precision#####numeric ###var_samp#########	#
var_pop(*numeric_type*) → ###double precision####real#### #double precision#####numeric #######################	#
var_samp(*numeric_type*) → ###double precision####real#### #double precision#####numeric #########################	#

9.62#################### ############################ #########ORDER BY#### ########*direct argument*################## *fraction*################0#1########## ################## ###NULL#*fraction*####NULL##########

#9.62

## ##	#####
mode() WITHIN GROUP(ORDER BY anyelement) → anyelement ## ########### ##########	#
percentile_cont(*fraction* double precision) WITHIN GROUP(ORDER BY double precision) → double precision percentile_cont(*fraction* double precision) WITHIN GROUP(ORDER BY interval) → interval ########################*fraction*########### ############### ######	#
percentile_cont(*fractions* double precision[]) WITHIN GROUP(ORDER BY double precision) → double precision[] percentile_cont(*fractions* double precision[]) WITHIN GROUP(ORDER BY interval) → interval[] ############### ###*fractions*##################NULL######## ##################################	#
percentile_disc(*fraction* double precision) WITHIN GROUP(ORDER BY anyelement) → anyelement ##################################*fraction*############## ### ###################	#
percentile_disc(*fractions* double precision[]) WITHIN GROUP(ORDER BY anyelement) → anyarray ############## ###*fractions*##################NULL######### ############## #####################	#

9.63####################9.22########################## ###########*args###### ####################################*sorted_args*############################

#########*args*###############*sorted_args*############################ ####################STRICT############NULL############# NULL##ORDER BY##

#9.63

## ##	#####
rank (*args*) WITHIN GROUP (ORDER BY *sorted_args*) → bigint ##	#
dense_rank (*args*) WITHIN GROUP (ORDER BY *sorted_args*) → bigint ##	#
percent_rank (*args*) WITHIN GROUP (ORDER BY *sorted_args*) → double precision ####################(rank - 1) / (total rows - 1)### ###############0 ##1#####	#
cume_dist (*args*) WITHIN GROUP (ORDER BY *sorted_args*) → double precision ###################(##################) / (####)### #######1/*N*##1 ###	#

#9.64

##
GROUPING (*group_by_expression(s)*) → integer ##GROUP BY################################### ################### ############# ##0########1###

9.64##################7.2.4######################## GROUPING############### ############GROUP BY###################

```
=> SELECT * FROM items_sold;
 make  | model | sales
-------+-------+-------
 Foo   | GT    |  10
 Foo   | Tour  |  20
 Bar   | City  |  15
 Bar   | Sport |  5
(4 rows)

=> SELECT make, model, GROUPING(make,model), sum(sales) FROM items_sold
 GROUP BY ROLLUP(make,model);
 make  | model | grouping | sum
-------+-------+----------+-----
 Foo   | GT    |        0 | 10
 Foo   | Tour  |        0 | 20
 Bar   | City  |        0 | 15
 Bar   | Sport |        0 | 5
 Foo   |       |        1 | 30
 Bar   |       |        1 | 20
       |       |        3 | 50
(7 rows)
```

#######4########0################################ #1#model################
##########3#make#model###
###

9.22.

#########3.5########4.2.8#########

############ 9.65######### ####################################OVER#######

###(##################
#####)###########9.21######### #############OVER##############################
############################1########

#9.65 #########

row_number () → bigint
rank () → bigint ##################################row_number########
dense_rank () → bigint ######################################
percent_rank () → double precision ############ (rank - 1) / (########## - 1)###### ##############0##1######
cume_dist () → double precision ############ (################) / (##########)###### ########1/*N*##1###
ntile (*num_buckets* integer) → integer ####################1###############
lag (*value* anycompatible [, *offset* integer [, *default* anycompatible]]) → anycompatible #############*offset*###########*value*###### ################ #*default*(*value*################)###### *offset*#*default*############ #### #######*offset*######1####*default*#NULL######
lead (*value* anycompatible [, *offset* integer [, *default* anycompatible]]) → anycompatible #############*offset*###########*value*###### ################ #*default*(*value*################)###### *offset*#*default*############ #### #######*offset*######1####*default*#NULL######
first_value (*value* anyelement) → anyelement #######################*value*######
last_value (*value* anyelement) → anyelement #######################*value*######
nth_value (*value* anyelement, *n* integer) → anyelement ################*n*############*value*###############NULL######

9.65#######################ORDER BY################ ORDER BY############
############ 4########cume_dist##################################

first_value#last_value#nth_value################################## ##### ### ###last_value#####nth_value### ################## OVER##########(RANGE#GROUP#####ROWS)################## ###### #############4.2.8#########

ORDER BY############### ## ################### ###ORDER BY#####ROWS BETWEEN UNBOUNDED PRECEDING AND UNBOUNDED FOLLOWING#########

##
SQL####lead#lag#first_value#last_value####nth_value###RESPECT NULLS####IGNORE NULLS########### ###PostgreSQL###################### #########RESPECT NULLS### ########nth_value####FROM FIRST####FROM LAST###################FROM FIRST############ #ORDER BY########## ##FROM LAST##############

9.23. Merge Support Functions

######PostgreSQL#########################MERGE#####RETURNING############### ###1######## #### 9.66#########

#9.66 Merge Support Functions

##
merge_action() → text ################################### ###'INSERT'#'UPDATE'### #'DELETE'######

##

```
MERGE INTO products p
  USING stock s ON p.product_id = s.product_id
  WHEN MATCHED AND s.quantity > 0 THEN
    UPDATE SET in_stock = true, quantity = s.quantity
  WHEN MATCHED THEN
    UPDATE SET in_stock = false, quantity = 0
  WHEN NOT MATCHED THEN
    INSERT (product_id, in_stock, quantity)
      VALUES (s.product_id, true, s.quantity)
  RETURNING merge_action(), p.*;

 merge_action | product_id | in_stock | quantity
--------------+------------+----------+----------
 UPDATE       |       1001 | t        |       50
 UPDATE       |       1002 | f        |        0
 INSERT       |       1003 | t        |       10
```

##########MERGE#####RETURNING######################## ################### ##

9.24.

####PostgreSQL######SQL################## ######################/########

9.24.1. EXISTS

EXISTS (*subquery*)

EXISTS########SELECT############# ################################ ##1#####
########EXISTS#####true#######################EXISTS#####false##########

##

##################
##

#############
######EXISTS####EXISTS(SELECT 1 WHERE ...)############### #####
##INTERSECT#########################

########col2################tab2##############tab1#######################
##

```
SELECT col1
FROM tab1
WHERE EXISTS (SELECT 1 FROM tab2 WHERE col2 = tab1.col2);
```

9.24.2. IN

expression IN (*subquery*)

###################################### ########################## #############
######IN#####true####### ######################################false#######

#####NULL############################1#######NULL######IN###########NULL###
########### ####NULL##############SQL############

EXISTS################################

row_constructor IN (*subquery*)

IN##########4.2.13################## ##
######################IN
##true######### ####################################false#######

#########NULL##SQL################ ##################NULL##################
#############NULL############################## ####################NULL##
##############NULL###########NULL#####IN####NULL######

9.24.3. NOT IN

expression NOT IN (*subquery*)

##################################### ################################ ########
##############################NOT IN#####true####### #####1###########false#####
##

#####NULL#####################################NULL########NOT IN###########
NULL############## ####NULL##############SQL############

EXISTS#################################

row_constructor NOT IN (*subquery*)

NOT IN##########4.2.13################ ##
######## ################################### ################################
###########NOT IN#####true########## ####################false#######

#########NULL##SQL################ 2################NULL##################
##########NULL########################### ###################NULL#### #######
########NULL#########1##NULL#####NOT IN####NULL######

9.24.4. ANY/SOME

expression operator ANY (*subquery*)
expression operator SOME (*subquery*)

####################1################ ##############################*operator*###
##############*operator*##################### #############ANY#####true####### #
################################false#######

SOME#ANY####### IN#= ANY######

#######################NULL########ANY###########NULL############### ##
##NULL##############SQL############

EXISTS#################################

row_constructor operator ANY (*subquery*)
row_constructor operator SOME (*subquery*)

ANY##########4.2.13################### ##
######## ##############################*operator*##################### #######
######################ANY#####true####### ####################################
#################false####### ####################################NULL########
NULL######

####################9.25.5#########

9.24.5. ALL

expression operator ALL (*subquery*)

##################################### ##############################*operator*#
################*operator*##################### ############################
###ALL#####true####### 1################false####### ####################1####
NULL##########NULL######

NOT IN#<> ALL######

EXISTS#################################

```
row_constructor operator ALL (subquery)
```

ALL##########4.2.13################ ##
##############################*operator*####################
###ALL#####true########## ################
#################false########## ##################################NULL########
NULL######

######################9.25.5##########

9.24.6.

```
row_constructor operator (subquery)
```

####4.2.13################### ##
################### ################NULL######### ############################
######

#########################9.25.5##########

9.25.

##############################
################ ##########PostgreSQL############SQL##### ######################
##/########

9.25.1. IN

```
expression IN (value [, ...])
```

################## ###########################true########## ############

```
expression = value1
OR
expression = value2
OR
...
```

#####NULL############################1#######NULL######IN###########NULL###
############ ####NULL##############SQL############

9.25.2. NOT IN

```
expression NOT IN (value [, ...])
```

################## ################################## ############

```
expression <> value1
AND
expression <> value2
AND
...
```

#####NULL##################################1#######NULL#######NOT IN#######
##############NULL############## ####NULL##############SQL############

###
#########x NOT IN y#NOT (x IN y)###### ####IN#######NOT IN######## ##NULL################ #######################

9.25.3. ANY/SOME (##)

```
expression operator ANY (array expression)
expression operator SOME (array expression)
```

############################ ######################operator###############
##operator#################### ############ANY#####true####### ##############
#################false#######

####NULL#########ANY####NULL##### ####NULL######ANY######NULL#####
#STRICT########################## ########NULL#########################
##ANY########NULL##########STRICT########## ####NULL#####SQL############
###########

SOME#ANY#######

9.25.4. ALL (##)

```
expression operator ALL (array expression)
```

############################ #######################operator############
#####operator#################### ##############################ALL###
##true####### 1################false#######

####NULL#########ALL####NULL##### ####NULL######ALL######NULL##########
######################## ########NULL##########################ALL########
NULL######################## ####NULL#####SQL#######################

9.25.5.

```
row_constructor operator row_constructor
```

4.2.13######################## 2############################## ##############
################ ## ##
##########B-################B-#########=################ ################
##=#<>#<#<=#>#>=#######################################

=#<>################ 2################NULL################## #########NULL
#########2################ ##################NULL####

<#<=#>#>=############################NULL#################### ###########
##NULL################NULL################################ ##
##ROW(1,2,NULL) < ROW(1,3,0)#####################NULL##########

row_constructor IS DISTINCT FROM *row_constructor*

##########<>#############NULL######NULL############## ##########NULL###NULL ########DISTINCT FROM###############NULL#####NOT DISTINCT########### ######## ############NULL########

row_constructor IS NOT DISTINCT FROM *row_constructor*

##########=#############NULL######NULL######## #####NULL########NULL######## ##DISTINCT FROM#########NULL#######NOT DISTINCT######## ################ ##NULL###########

9.25.6.

record operator record

SQL########2##NULL#####NULL##NULL#########################NULL############# ## PostgreSQL##(9.25.5######)#####################################(9.24######)###### ################### ##########################NULL#################NULL##NULL ############ ######################################

#################### ########*operator*#=#<>#<#<=#>#>=######################### ### (#########B-tree###################B-tree#######=#########################) ### ##########################IS [NOT] DISTINCT FROM######9.25.5####

######B-tree### ## #*=#*<>#*<#*<=#*>#*>=### #######2################### 2#####################2## ################### ##################################### ##################### #############B-Tree#####(64.1.4.3##)##

9.26.

########################### ###################### 9.67##### 9.68################## ###### ###################################### #######################7.2.1.4###### ####

#9.67

##
generate_series (*start* integer, *stop* integer [, *step* integer]) → setof integer generate_series (*start* bigint, *stop* bigint [, *step* bigint]) → setof bigint generate_series (*start* numeric, *stop* numeric [, *step* numeric]) → setof numeric *start*##*stop*#####*step*############ *step*#######1###
generate_series (*start* timestamp, *stop* timestamp, *step* interval) → setof timestamp generate_series (*start* timestamp with time zone, *stop* timestamp with time zone, *step* interval [, *timezone* text]) → setof timestamp with time zone

##
start##stop#####step############ *#############timezone############* ##########TimeZone##################################

step#####start#stop############## *####step######start#stop##############* *#########NULL###########* *step#############* *#############*

```
SELECT * FROM generate_series(2,4);
 generate_series
-----------------
               2
               3
               4
(3 rows)

SELECT * FROM generate_series(5,1,-2);
 generate_series
-----------------
               5
               3
               1
(3 rows)

SELECT * FROM generate_series(4,3);
 generate_series
-----------------
(0 rows)

SELECT generate_series(1.1, 4, 1.3);
 generate_series
-----------------
             1.1
             2.4
             3.7
(3 rows)


-- ########################
SELECT current_date + s.a AS dates FROM generate_series(0,14,7) AS s(a);
   dates
------------
 2004-02-05
 2004-02-12
 2004-02-19
(3 rows)

SELECT * FROM generate_series('2008-03-01 00:00'::timestamp,
                              '2008-03-04 12:00', '10 hours');
   generate_series
---------------------
 2008-03-01 00:00:00
 2008-03-01 10:00:00
 2008-03-01 20:00:00
 2008-03-02 06:00:00
 2008-03-02 16:00:00
 2008-03-03 02:00:00
 2008-03-03 12:00:00
 2008-03-03 22:00:00
 2008-03-04 08:00:00
(9 rows)
```

```
-- this example assumes that TimeZone is set to UTC; note the DST
 transition:
SELECT * FROM generate_series('2001-10-22 00:00 -04:00'::timestamptz,
                              '2001-11-01 00:00 -05:00'::timestamptz,
                              '1 day'::interval, 'America/New_York');
    generate_series
------------------------
 2001-10-22 04:00:00+00
 2001-10-23 04:00:00+00
 2001-10-24 04:00:00+00
 2001-10-25 04:00:00+00
 2001-10-26 04:00:00+00
 2001-10-27 04:00:00+00
 2001-10-28 04:00:00+00
 2001-10-29 05:00:00+00
 2001-10-30 05:00:00+00
 2001-10-31 05:00:00+00
 2001-11-01 05:00:00+00
(11 rows)
```

#9.68

##
generate_subscripts(*array* anyarray, *dim* integer) → setof integer   #######*dim*#######################
generate_subscripts(*array* anyarray, *dim* integer, *reverse* boolean) → setof integer   ######*dim*####################### *reverse*#################

generate_subscripts### ############ ##########NULL##0###### ##############

```
-- basic usage:
SELECT generate_subscripts('{NULL,1,NULL,2}'::int[], 1) AS s;
 s
---
 1
 2
 3
 4
(4 rows)


-- #######################
-- ############
SELECT * FROM arrays;
         a
--------------------
 {-1,-2}
 {100,200,300}
(2 rows)

SELECT a AS array, s AS subscript, a[s] AS value
FROM (SELECT generate_subscripts(a, 1) AS s, a FROM arrays) foo;
     array     | subscript | value
---------------+-----------+-------
 {-1,-2}       |         1 |    -1
 {-1,-2}       |         2 |    -2
 {100,200,300} |         1 |   100
```

```
 {100,200,300} |         2 |   200
 {100,200,300} |         3 |   300
(5 rows)


-- 2##############
CREATE OR REPLACE FUNCTION unnest2(anyarray)
RETURNS SETOF anyelement AS $$
select $1[i][j]
   from generate_subscripts($1,1) g1(i),
        generate_subscripts($1,2) g2(j);
$$ LANGUAGE sql IMMUTABLE;
CREATE FUNCTION
SELECT * FROM unnest2(ARRAY[[1,2],[3,4]]);
 unnest2
---------
       1
       2
       3
       4
(4 rows)
```

FROM#######WITH ORDINALITY########1##############1#####bigint#############
###unnest()#######################

```
-- WITH ORDINALITY###########
SELECT * FROM pg_ls_dir('.') WITH ORDINALITY AS t(ls,n);
       ls        | n
-----------------+----
 pg_serial       |  1
 pg_twophase     |  2
 postmaster.opts |  3
 pg_notify       |  4
 postgresql.conf |  5
 pg_tblspc       |  6
 logfile         |  7
 base            |  8
 postmaster.pid  |  9
 pg_ident.conf   | 10
 global          | 11
 pg_xact         | 12
 pg_snapshots    | 13
 pg_multixact    | 14
 PG_VERSION      | 15
 pg_wal          | 16
 pg_hba.conf     | 17
 pg_stat_tmp     | 18
 pg_subtrans     | 19
(19 rows)
```

9.27.

###########PostgreSQL#################################

9.27.1.

9.69############################

27.2.26############

#9.69

current_catalog → name current_database () → name ################ #######SQL########################## #current_catalog######
current_query () → text ##
current_role → name current_user#####
current_schema → name current_schema () → name ################################NULL####### #################### ##################################
current_schemas (*include_implicit* boolean) → name[] ################################## ####search_path################ ################## ########pg_catalog########################## ##
current_user → name ####################
inet_client_addr () → inet #########IP#########UNIX##############NULL######
inet_client_port () → integer #########IP##########UNIX##############NULL######
inet_server_addr () → inet ################IP#########UNIX##############NULL######
inet_server_port () → integer ################IP##########UNIX##############NULL######
pg_backend_pid () → integer ###########################ID######
pg_blocking_pids (integer) → integer[] ##########ID###################ID################################# ################## ### ### ## ## #####ID#####pg_backend_pid########## ####################PID##### ####### ##ID############ ######## ### ##
pg_conf_load_time () → timestamp with time zone ############################ ## ## ########postmaster#### ####################

pg_current_logfile([text]) → text ############################## ####log_directory################ ###### #############NULL###### ############################ #pg_current_logfile#######stderr#csvlog#jsonlog########## ############# ##################NULL###### #################### ##########csvlog#jsonlog###stderr################### ####### #log_destination################NULL###### ###current_logfiles###### ####### ######################pg_monitor################################ EXECUTE#############
pg_my_temp_schema() → oid ###############OID##
pg_is_other_temp_schema(oid) → boolean ####OID###############OID########### ############################## ###########
pg_jit_available() → boolean JIT############(#30###)#jit########on#########
pg_listening_channels() → setof text #####################################
pg_notification_queue_usage() → double precision ##0##1######### ## #LISTEN#NOTIFY#######
pg_postmaster_start_time() → timestamp with time zone #############
pg_safe_snapshot_blocking_pids(integer) → integer[] ##########ID#####################################ID################# ############################## SERIALIZABLE####################SERIALIZABLE READ ONLY DEFERRABLE## ####### ##13.2.3########## ### ########
pg_trigger_depth() → integer PostgreSQL###################(############################0#####)#
session_user → name ###############
system_user → text ##ID(######)###### ############# #auth_method:identity##########################Trust############# #NULL#########
user → name current_user######

##
current_catalog#current_role#current_schema#current_user#session_user# ##user#SQL## PostgreSQL# #current_schema#########################

session_user##SET SESSION AUTHORIZATION################ current_user#################### ######## #########SET ROLE########### SECURITY DEFINER################## Unix# ##################################### current_role#user#current_user#### ### ###SQL##current_role#current_user#########PostgreSQL#########1####### #######################

9.27.2.

9.70## ##########5.8######## ## ######################OID(pg_authid.oid)######## ###public########PUBLIC ############### ###*user*################current_user############ ######### #####OID######## ########################### ################################ #########SELECT################################ ########################### ########WITH GRANT OPTION########### ##################################### ###############

```
SELECT has_table_privilege('myschema.mytable', 'select');
SELECT has_table_privilege('joe', 'mytable', 'INSERT, SELECT WITH GRANT
 OPTION');
```

#9.70 #########

##
has_any_column_privilege ([*user* name or oid,] *table* text or oid, *privilege* text) → boolean ############################## ## ## ######SELECT#INSERT#UPDATE#REFERENCES###
has_column_privilege ([*user* name or oid,] *table* text or oid, *column* text or smallint, *privilege* text) → boolean ################################## ########################## ########## ###########(pg_attribute.attnum)######### ###### #SELECT#INSERT#UPDATE#REFERENCES###
has_database_privilege ([*user* name or oid,] *database* text or oid, *privilege* text) → boolean #################### #######CREATE#CONNECT#TEMPORARY#TEMP (TEMPORARY#####)###
has_foreign_data_wrapper_privilege ([*user* name or oid,] *fdw* text or oid, *privilege* text) → boolean ########################## #######USAGE#####
has_function_privilege ([*user* name or oid,] *function* text or oid, *privilege* text) → boolean ################ #######EXECUTE##### ###OID##############regprocedure#####8.19############### #######

##
SELECT has_function_privilege('joeuser', 'myfunc(int, text)', 'execute');
has_language_privilege ([*user* name or oid,] *language* text or oid, *privilege* text) → boolean ################ #######USAGE#####
has_parameter_privilege ([*user* name or oid,] *parameter* text, *privilege* text) → boolean #####################? ####################### ##########SET###ALTER SYSTEM###
has_schema_privilege ([*user* name or oid,] *schema* text or oid, *privilege* text) → boolean ################## #######CREATE#USAGE###
has_sequence_privilege ([*user* name or oid,] *sequence* text or oid, *privilege* text) → boolean ################### #######USAGE#SELECT#UPDATE###
has_server_privilege ([*user* name or oid,] *server* text or oid, *privilege* text) → boolean ################### #######USAGE#####
has_table_privilege ([*user* name or oid,] *table* text or oid, *privilege* text) → boolean #####[74.489796]################## ###### #SELECT#INSERT#UPDATE#DELETE#TRUNCATE#REFERENCES#TRIGGER# ## ###########################? ######### #SELECT#INSERT#UPDATE#DELETE#TRUNCATE#REFERENCES#TRIGGER#MAINTAIN# ##
has_tablespace_privilege ([*user* name or oid,] *tablespace* text or oid, *privilege* text) → boolean #################### #######CREATE###
has_type_privilege ([*user* name or oid,] *type* text or oid, *privilege* text) → boolean ################## #######USAGE##### OID#################### #regtype################8.19####
pg_has_role ([*user* name or oid,] *role* text or oid, *privilege* text) → boolean ################# #######MEMBER#USAGE#SET### ################## ####MEMBER##################################SET ROLE######### SET#SET ROLE##################################USAGE##########SET ROLE########################## #####*user*#public################## PUBLIC##############################
row_security_active (*table* text or oid) → boolean ##

################aclitem########### 9.71###### ####################5.8########

#9.71 aclitem###

aclitem = aclitem → boolean aclitem###### #aclitem########################## #####aclitem###### ########## 'calvin=r*w/hobbes'::aclitem = 'calvin=r*w*/ hobbes'::aclitem → f aclitem[] @> aclitem → boolean ################ ####aclitem####################################### ################## '{calvin=r*w/hobbes,hobbes=r*w*/postgres}'::aclitem[] @> 'calvin=r*/hobbes'::aclitem → t aclitem[] ~ aclitem → boolean ###@>########## '{calvin=r*w/hobbes,hobbes=r*w*/postgres}'::aclitem[] ~ 'calvin=r*/hobbes'::aclitem → t

9.72#aclitem################

#9.72 aclitem##

##
acldefault (*type* "char", *ownerId* oid) → aclitem[] *ownerId*#OID##########*type*##################aclitem###### ### #########ACL###null##################### ############## ##5.8######## *type*##################### 'c'#COLUMN#'r'#TABLE# ################'s'#SEQUENCE#'d'#DATABASE#'f'#FUNCTION# ###PROCEDURE#'l'#LANGUAGE#'L'#LARGE OBJECT#'n'#SCHEMA#'p'#PARAMETER#'t'#TABLESPACE#'F'#FOREIGN DATA WRAPPER#'S'#FOREIGN SERVER#'T'#TYPE####DOMAIN######
aclexplode (aclitem[]) → setof record (*grantor* oid, *grantee* oid, *privilege_type* text, *is_grantable* boolean) #######aclitem####### ################PUBLIC###*grantee*# #*grantee*########## ##########SELECT#INSERT#########(####### 5.1## ######## ###############*privilege_type*##########################
makeaclitem (*grantee* oid, *grantor* oid, *privileges* text, *is_grantable* boolean) → aclitem #######aclitem###### *privileges*##########SELECT#INSERT######## ############ ## ##########

9.27.3.

9.73### ########################### ## ################################### ########

```
SELECT relname FROM pg_class WHERE pg_table_is_visible(oid);
```

#########
##################################

#9.73 ###########

##
pg_collation_is_visible(*collation* oid) → boolean ################
pg_conversion_is_visible(*conversion* oid) → boolean ##############
pg_function_is_visible(*function* oid) → boolean ############## ###################
pg_opclass_is_visible(*opclass* oid) → boolean ##################
pg_operator_is_visible(*operator* oid) → boolean ###############
pg_opfamily_is_visible(*opclass* oid) → boolean ################
pg_statistics_obj_is_visible(*stat* oid) → boolean ######################
pg_table_is_visible(*table* oid) → boolean ################ ## ########
pg_ts_config_is_visible(*config* oid) → boolean ###################
pg_ts_dict_is_visible(*dict* oid) → boolean ###################
pg_ts_parser_is_visible(*parser* oid) → boolean ###################
pg_ts_template_is_visible(*template* oid) → boolean ########################
pg_type_is_visible(*type* oid) → boolean ######################

##################################OID####### #################OID### #regclass#regtype#regprocedure#regoperator#regconfig### #regdictionary########### #############

```
SELECT pg_type_is_visible('myschema.widget'::regtype);
```

###
###

9.27.4.

9.74##########################

#9.74

##	##
format_type (*type* oid, *typemod* integer) → text	#OID##############SQL####### ###########NULL##############
pg_basetype (regtype) → regtype	############OID#########OID###### ###########OID############### ### ####OID#################### ############################ ######CREATE DOMAIN mytext AS text###### pg_basetype('mytext'::regtype) → text
pg_char_to_encoding (*encoding* name) → integer	### ################ #-1######
pg_encoding_to_char (*encoding* integer) → name	## ########### #########################
pg_get_catalog_foreign_keys () → setof record (*fktable* regclass, *fkcols* text[], *pktable* regclass, *pkcols* text[], *is_array* boolean, *is_opt* boolean)	PostgreSQL#################################### *fktable*############## ##*fkcols*############### ####*pktable*################*pkcols*######## ####### *is_array*################################ *is_opt*########### #########################
pg_get_constraintdef (*constraint* oid [, *pretty* boolean]) → text	################## #####################################
pg_get_expr (*expr* pg_node_tree, *relation* oid [, *pretty* boolean]) → text	### ##############2############# #########OID####################################
pg_get_functiondef (*func* oid) → text	########################## ################################### ##### #CREATE OR REPLACE FUNCTION####CREATE OR REPLACE PROCEDURE## ##
pg_get_function_arguments (*func* oid) → text	CREATE FUNCTION################################ ##############
pg_get_function_identity_arguments (*func* oid) → text	ALTER FUNCTION### ########### #######
pg_get_function_result (*func* oid) → text	CREATE FUNCTION#########RETURNS######## #########NULL######
pg_get_indexdef (*index* oid [, *column* integer, *pretty* boolean]) → text	###################### ################################### *column*## ##############################
pg_get_keywords () → setof record (*word* text, *catcode* "char", *barelabel* boolean, *catdesc* text, *baredesc* text)	#######SQL###################### *word*############## *catcode##### ###########U#########C##############T##################R######## ###### *barelabel*#########bare########SELECT#########true##AS###

##
#######false### *catdesc*### *baredesc*##
pg_get_partkeydef (*table* oid) → text ########################CREATE TABLE#PARTITION BY###########(## #####################################)
pg_get_ruledef (*rule* oid [, *pretty* boolean]) → text ################### ###################################
pg_get_serial_sequence (*table* text, *column* text) → text ################### ###################NULL###### ##IDENTITY ##############IDENTITY######################## SERIAL# #serial#smallserial#bigserial####################SERIAL###### ############### ############ALTER SEQUENCE OWNED BY########### ###### #########pg_get_owned_sequence######################### #serial####bigserial##################### ######################### ######## ############## ## ## ################### ## #### #############9.17########################## #######IDENTITY####SERIAL##########################   SELECT currval(pg_get_serial_sequence('sometable', 'id'));
pg_get_statisticsobjdef (*statobj* oid) → text ########################### ###################################
pg_get_triggerdef (*trigger* oid [, *pretty* boolean]) → text ################### ###################################
pg_get_userbyid (*role* oid) → name OID###############
pg_get_viewdef (*view* oid [, *pretty* boolean]) → text #######################SELECT########### ########################## ##########
pg_get_viewdef (*view* oid, *wrap_column* integer) → text #######################SELECT########### ########################## ######### ##
pg_get_viewdef (*view* text [, *pretty* boolean]) → text #######################SELECT###############OID################### ###########OID###############
pg_index_column_has_property (*index* regclass, *column* integer, *property* text) → boolean ############################### ################ 9.75######### ####### ## ############################ ##OID#########################NULL######
pg_index_has_property (*index* regclass, *property* text) → boolean ############################### ############### 9.76######### ####### ## ############################ ##OID##################NULL######
pg_indexam_has_property (*am* oid, *property* text) → boolean

##
################################### ############## 9.77######### ## ############################OID##################NULL######
pg_options_to_table(*options_array* text[]) → setof record(*option_name* text, *option_value* text) pg_class.reloptions####pg_attribute.attoptions################ ########
pg_settings_get_flags(*guc* text) → text[] #####GUC################## #######NULL###### GUC################ ############## # 9.78##########################
pg_tablespace_databases(*tablespace* oid) → setof oid ##############################OID######## ############ ######################### ############################## ##pg_tablespace_databases################pg_class############## ##
pg_tablespace_location(*tablespace* oid) → text ##############################
pg_typeof("any") → regtype ##########OID###### #################SQL################ ######OID# ######regtype##############8.19##############OID################## pg_typeof(33) → integer
COLLATION FOR("any") → text ################## ####################### ################NULL#### ## ######################## collation for ('foo'::text) → "default" collation for ('foo' COLLATE "de_DE") → "de_DE"
to_regclass(text) → regclass ##############OID###### ###########regclass################### #8.19#### ################################NULL######
to_regcollation(text) → regcollation ###########OID###### ###########regcollation################### #8.19#### ################################NULL######
to_regnamespace(text) → regnamespace ############OID###### ###########regnamespace################### #8.19#### ################################NULL######
to_regoper(text) → regoper ###########OID###### ###########regoper#####################8.19# ### ######################################NULL######
to_regoperator(text) → regoperator ####################OID###### ###########regoperator############ ########8.19#### ################################NULL######
to_regproc(text) → regproc ####################OID###### ###########regproc################## ###8.19#### ######################################NULL######
to_regprocedure(text) → regprocedure

##
##########################OID####### ###########regprocedure###### ###############8.19#### ################################NULL######
to_regrole(text) → regrole ###########OID####### ###########regrole####################8.19### ################################NULL######
to_regtype(text) → regtype ##############################OID####### ########################### ##################NULL###### ####regtype#8.19#################### ########################
to_regtypemod(text) → integer ## ################################NULL###### ################-1### ######to_regtypemod#to_regtype#######format_type################## ### ######################## format_type(to_regtype('varchar(32)'), to_regtypemod('varchar(32)')) → character varying(32)

#####################################*pretty*#######true#################### # ################################## ############################PostgreSQL##### ############### ########################### *pretty*###false################# ########

#9.75

##	##
asc	##################
desc	##################
nulls_first	########NULL###########
nulls_last	########NULL###########
orderable	##################
distance_orderable	###########################ORDER BY col <-> ####
returnable	###########################
search_array	##col = ANY(array)################ ##
search_nulls	##IS NULL###IS NOT NULL########### #

#9.76

##	##
clusterable	#######CLUSTER##############
index_scan	##################################
bitmap_scan	##########################
backward_scan	##################################### ###### FETCH BACKWARD##########

#9.77

##	##
can_order	########CREATE INDEX####ASC#DESC## ##################
can_unique	#########################
can_multi_col	##############################
can_exclude	#####################
can_include	########CREATE INDEX#INCLUDE##### #####

#9.78 GUC Flags

###	##
EXPLAIN	#############EXPLAIN (SETTINGS)### #######
NO_SHOW_ALL	#############SHOW ALL############
NO_RESET	#############RESET#############
NO_RESET_ALL	#############RESET ALL############
NOT_IN_SAMPLE	################## #postgresql.conf#######
RUNTIME_COMPUTED	########################

9.27.5.

9.79###############################

#9.79

##
pg_describe_object (*classid* oid, *objid* oid, *objsubid* integer) → text ####OID######OID############OID##############################0### ############################## ##################################### ############ ###pg_depend########################## ############# #####NULL######
pg_identify_object (*classid* oid, *objid* oid, *objsubid* integer) → record (*type* text, *schema* text, *name* text, *identity* text) ####OID######OID############ID################################### ####### ################################ *type*##################### ## *schema*##NULL###### *name*# ### #######NULL##### *identity*### ## ###################################### #########NULL########
pg_identify_object_as_address (*classid* oid, *objid* oid, *objsubid* integer) → record (*type* text, *object_names* text[], *object_args* text[]) ####OID######OID############ID################################### ######## ################### ################################### ####### *type*######################## *object_names*#*object_args*# ################################### ####3########################## #pg_get_object_address##########

##
pg_get_object_address (*type* text, *object_names* text[], *object_args* text[]) → record (*classid* oid, *objid* oid, *objsubid* integer) ## ########### #######pg_depend################## ##pg_identify_object#pg_describe_object################### *classid*##################OID### *objid*#########OID### *objsubid*### #######ID#####0### #####pg_identify_object_as_address###### ##### #####NULL########

9.27.6.

9.80########COMMENT##########################
##NULL#######

#9.80 #######

##
col_description (*table* oid, *column* integer) → text #####OID######################### #obj_description################ ##OID###########
obj_description (*object* oid, *catalog* name) → text OID## ### #obj_description(123456, 'pg_class')#OID 123456###############
obj_description (*object* oid) → text OID############################## ########################OID######## ###################################
shobj_description (*object* oid, *catalog* name) → text OID## ######################## ###########################obj_description###### ################# ##

9.27.7.

9.81##################################

#9.81 ##########

#
pg_input_is_valid (*string* text, *type* text) → boolean ## ### ########### ###################################### pg_input_is_valid('42', 'integer') → t pg_input_is_valid('42000000000', 'integer') → f pg_input_is_valid('1234.567', 'numeric(7,4)') → f
pg_input_error_info (*string* text, *type* text) → record (*message* text, *detail* text, *hint* text, *sql_error_code* text)

###########NULL###### ###pg_input_is_valid###### ### ####################################### SELECT * FROM pg_input_error_info('42000000000', 'integer') → message | detail | hint | sql_error_code ----------+-------- +------+---------- value "42000000000" is out of range for type integer | | | 22003

9.27.8. ########ID##############

9.82#################################### ##################################

#9.82 ########ID#############

##
pg_current_xact_id () → xid8 ###########ID###### ##########ID#####(##################)####ID#### ###### ###66.1######### ####################-##########ID######## ## #66.3#########
pg_current_xact_id_if_assigned () → xid8 ##########ID####################NULL###### #################### XID######################## ####################-##########ID###### ##
pg_xact_status (xid8) → text ########################## ## #################in progress#committed####aborted######### ##### ##NULL###### COMMIT### ### ####### ##############in progress################ID################ #########################pg_prepared_xacts####################
pg_current_snapshot () → pg_snapshot #########ID#in-progress##################*snapshot*###### ################ #######ID######## #########ID###########66.3#######
pg_snapshot_xip (pg_snapshot) → setof xid8 ############in-progress########ID########
pg_snapshot_xmax (pg_snapshot) → xid8 ########xmax######
pg_snapshot_xmin (pg_snapshot) → xid8 ########xmin######
pg_visible_in_snapshot (xid8, pg_snapshot) → boolean

##
##########################ID################################ ######## ######ID#subxid########################### ###66.3##########

##########ID##xid##32#######40################## ######## 9.82############ ##################xid8##64##################xid############## ###66.1####### ### ################pg_snapshot####################ID################ # ##### 9.83######### pg_snapshot#######*xmin*:*xmax*:*xip_list*### ### #10:20:10,14,15#xmin=10, xmax=20, xip_list=10, 14, 15###########

#9.83

##	##
xmin	##################ID# xmin########## ##################################### ####
xmax	################ID######ID# xmax##### ################ID################### ########
xip_list	######################## xmin <= *X* <xmax#################ID############ ################################## ## ####################ID#subxids########

PostgreSQL#####13#####xid8################64####XID########bigint############ ########txid_snapshot######### #########txid########### ################## ############################## # 9.84#########

#9.84 ##############ID##############

##
age (xid) → integer ############ID################################
mxid_age (xid) → integer ################ID################################ID########
txid_current () → bigint pg_current_xact_id()###
txid_current_if_assigned () → bigint pg_current_xact_id_if_assigned()###
txid_current_snapshot () → txid_snapshot pg_current_snapshot()###
txid_snapshot_xip (txid_snapshot) → setof bigint pg_snapshot_xip()###
txid_snapshot_xmax (txid_snapshot) → bigint pg_snapshot_xmax()###
txid_snapshot_xmin (txid_snapshot) → bigint pg_snapshot_xmin()###
txid_visible_in_snapshot (bigint, txid_snapshot) → boolean

##	##
pg_visible_in_snapshot()	###
txid_status (bigint) → text	pg_xact_status() ###

9.27.9.

9.85################################## track_commit_timestamp################ ##

#9.85 ######################

##	##
pg_xact_commit_timestamp (xid) → timestamp with time zone	########################
pg_xact_commit_timestamp_origin (xid) → record (*timestamp* timestamp with time zone, *roident* oid)	##################################
pg_last_committed_xact () → record (*xid* xid, *timestamp* timestamp with time zone, *roident* oid)	######################ID##########################

9.27.10.

9.86#####################initdb################## ######################### ############ ## #######pg_controldata##

#9.86 #######

##	##
pg_control_checkpoint () → record	# 9.87##############################
pg_control_system () → record	# 9.88##############################
pg_control_init () → record	# 9.89##########################
pg_control_recovery () → record	# 9.90#######################

#9.87 pg_control_checkpoint####

##	####
checkpoint_lsn	pg_lsn
redo_lsn	pg_lsn
redo_wal_file	text
timeline_id	integer

##	####
prev_timeline_id	integer
full_page_writes	boolean
next_xid	text
next_oid	oid
next_multixact_id	xid
next_multi_offset	xid
oldest_xid	xid
oldest_xid_dbid	oid
oldest_active_xid	xid
oldest_multi_xid	xid
oldest_multi_dbid	oid
oldest_commit_ts_xid	xid
newest_commit_ts_xid	xid
checkpoint_time	timestamp with time zone

#9.88 pg_control_system####

##	####
pg_control_version	integer
catalog_version_no	integer
system_identifier	bigint
pg_control_last_modified	timestamp with time zone

#9.89 pg_control_init Output Columns

##	####
max_data_alignment	integer
database_block_size	integer
blocks_per_segment	integer
wal_block_size	integer
bytes_per_wal_segment	integer
max_identifier_length	integer
max_index_columns	integer
max_toast_chunk_size	integer
large_object_chunk_size	integer
float8_pass_by_value	boolean
data_page_checksum_version	integer

#9.90 pg_control_recovery####

##	####
min_recovery_end_lsn	pg_lsn
min_recovery_end_timeline	integer
backup_start_lsn	pg_lsn

##	####
backup_end_lsn	pg_lsn
end_of_backup_record_required	boolean

9.27.11. Version Information Functions

####### 9.91####################

#9.91 Version Information Functions

##
##
version () → text PostgreSQL###################### ###server_version######## ##### #########server_version_num######## ######################## ##server_version_num (8.2######)#PQserverVersion#######
unicode_version () → text ######PostgreSQL##########################
icu_unicode_version () → text ######### ICU ############## ICU ##### Unicode ###################### ###### NULL

9.27.12. WAL Summarization Information Functions

####### 9.92#######WAL################# summarize_wal##########

#9.92 WAL Summarization Information Functions

##
##
pg_available_wal_summaries () → setof record (*tli* bigint, *start_lsn* pg_lsn, *end_lsn* pg_lsn) ######pg_wal/summaries##############WAL################## #WAL ########1##1##WAL######### ###########LSN########TLI####WAL##### ## #######LSN###WAL######### #################
pg_wal_summary_contents (*tli* bigint, *start_lsn* pg_lsn, *end_lsn* pg_lsn) → setof record (*relfilenode* oid, *reltablespace* oid, *reldatabase* oid, *relforknumber* smallint, *relblocknumber* bigint, *is_limit_block* boolean) ######TLI########WAL################### ##LSN###LSN####### is_limit_block### ###1##WAL################### is_limit_block########(a) ##fork#WAL ############relblocknumber#################(b) ##fork#WAL######### ################### #####relblocknumber#0######
pg_get_wal_summarizer_state () → record (*summarized_tli* bigint, *summarized_lsn* pg_lsn, *pending_lsn* pg_lsn, *summarizer_pid* int) ######WAL################### #############WAL################ ##summarized_tli#summarized_lsn#####0#0/0###### ################ ######WAL######TLI###LSN###### WAL################pending_lsn### ###########LSN###### #####summarized_lsn############## WAL####### ###########summarized_lsn###### summarizer_pid#WAL############# ######PID### ##########null#PID###

##
##############wal_##=minimal######WAL########WAL######WAL##### ########### ### ######## ## wal_############### ########WAL################################

9.28.

##########PostgreSQL##########################

9.28.1.

9.93##################################

#9.93 ######

#
current_setting (*setting_name* text [, *missing_ok* boolean]) → text   ###*setting_name*######### ############*missing_ok*#######true####   #current_setting################NULL###### #####SQL#####SHOW##   #####   current_setting('datestyle') → ISO, MDY
set_config (*setting_name* text, *new_value* text, *is_local* boolean) → text   *setting_name*#####*new_value*############ *is_local*#######true###   ######################### ############################false#######   ######SQL#####SET######   set_config('log_statement_stats', 'off', false) → off

9.28.2.

9.94############################## ## ####GRANT#####################

#######################true##################false######

#9.94 ###########

##
pg_cancel_backend (*pid* integer) → boolean   #######ID################################## ########################   ###########pg_signal_backend################### ################   ##################
pg_log_backend_memory_contexts (*pid* integer) → boolean   ######ID############################## ##############################   ###### ##########LOG############### ###########19.8##################   ##client_min_messages##################
pg_reload_conf () → boolean

##	##
	PostgreSQL############################## ####SIGHUP##### postmaster################postmaster####SIGHUP######### ####### ##pg_file_settings#pg_hba_file_rules#pg_ident_file_mappings### ################################
pg_rotate_logfile () → boolean	###################################### ############################## ######################
pg_terminate_backend (*pid* integer, *timeout* bigint DEFAULT 0) → boolean	###############ID############### ############################## ##pg_signal_backend################### ########################## ######## *timeout*## ####true###### *timeout*## #### ###############true###### ##################false######

pg_cancel_backend#pg_terminate_backend######SIGINT###SIGTERM######## ######################## ##################pg_stat_activity####pid####### ###Unix##ps#Windows##Task Manager########postgres################### ####### #######pg_stat_activity#usename##############

############################pg_log_backend_memory_contexts####### #######

```
postgres=# SELECT pg_log_backend_memory_contexts(pg_backend_pid());
 pg_log_backend_memory_contexts
--------------------------------
 t
(1 row)
```

#######################################

```
LOG:  logging memory contexts of PID 10377
STATEMENT:  SELECT pg_log_backend_memory_contexts(pg_backend_pid());
LOG:  level: 0; TopMemoryContext: 80800 total in 6 blocks; 14432 free (5
 chunks); 66368 used
LOG:  level: 1; pgstat TabStatusArray lookup hash table: 8192 total in 1
 blocks; 1408 free (0 chunks); 6784 used
LOG:  level: 1; TopTransactionContext: 8192 total in 1 blocks; 7720 free
 (1 chunks); 472 used
LOG:  level: 1; RowDescriptionContext: 8192 total in 1 blocks; 6880 free
 (0 chunks); 1312 used
LOG:  level: 1; MessageContext: 16384 total in 2 blocks; 5152 free (0
 chunks); 11232 used
LOG:  level: 1; Operator class cache: 8192 total in 1 blocks; 512 free (0
 chunks); 7680 used
LOG:  level: 1; smgr relation table: 16384 total in 2 blocks; 4544 free (3
 chunks); 11840 used
LOG:  level: 1; TransactionAbortContext: 32768 total in 1 blocks; 32504
 free (0 chunks); 264 used
...
LOG:  level: 1; ErrorContext: 8192 total in 1 blocks; 7928 free (3
 chunks); 264 used
LOG:  Grand total: 1651920 bytes in 201 blocks; 622360 free (88 chunks);
 1029560 used
```

####100###################100#################################### ############ ################################### ##########################

9.28.3.

9.95################################ ####################(pg_backup_start#pg_backup_stop#pg_wal_lsn_diff#####)#

###################25.3#########

#9.95 ##########

##
pg_create_restore_point (*name* text) → pg_lsn ## # ###########################recovery_target_name########### ############## ## ##EXECUTE##############
pg_current_wal_flush_lsn () → pg_lsn ###
pg_current_wal_insert_lsn () → pg_lsn ######################################
pg_current_wal_lsn () → pg_lsn ######################################
pg_backup_start (*label* text [, *fast* boolean]) → pg_lsn ########################### ################################ #### ################################### ######2########true######## ##pg_backup_start##################### #####I/O################### ############################## ##EXECUTE##############
pg_backup_stop ([*wait_for_archive* boolean]) → record (*lsn* pg_lsn, *labelfile* text, *spcmapfile* text) #################### ## ############################ ###################################### ####################PostgreSQL########### ######boolean########### false###################WAL############### ######## #####WAL################################### ############ ##################WAL########################### ################ true##############pg_backup_stop#WAL################ ############ ##archive_mode = always################ ########################## ##################pg_switch_wal################ ### ####### #pg_backup_start## ######### ## ######## ## ################ *lsn*################################### 2########### #############3########################### ########################### #################### ##EXECUTE##############
pg_switch_wal () → pg_lsn ## ####### ############## ##1### ##### #####################################pg_switch_wal################## ############### ##EXECUTE##############

##	##
pg_walfile_name (*lsn* pg_lsn) → text	##########################WAL#############
pg_walfile_name_offset (*lsn* pg_lsn) → record (*file_name* text, *file_offset* integer)	##
pg_split_walfile_name (*file_name* text) → record (*segment_number* numeric, *timeline_id* bigint)	WAL####################ID#######
pg_wal_lsn_diff (*lsn1* pg_lsn, *lsn2* pg_lsn) → numeric	2#####################(*lsn1* - *lsn2*)####### ## #pg_stat_replication## 9.95##################################

pg_current_wal_lsn## ## #pg_current_wal_insert_lsn#################### ##pg_current_wal_flush_lsn########################### ##### ############## ############ ## ##### ### ### ################################ #####################################

pg_walfile_name_offset######pg_lsn#######################################
##########

```
postgres=# SELECT * FROM pg_walfile_name_offset((pg_backup_stop()).lsn);
        file_name         | file_offset
--------------------------+-------------
 000000010000000000000000D |     4039624
(1 row)
```

##########pg_walfile_name#####################

pg_split_walfile_name###########WAL######LSN##################

```
postgres=# \set file_name '000000010000000100C000AB'
postgres=# \set offset 256
postgres=# SELECT '0/0'::pg_lsn + pd.segment_number * ps.setting::int
 + :offset AS lsn
  FROM pg_split_walfile_name(:'file_name') pd,
       pg_show_all_settings() ps
  WHERE ps.name = 'wal_segment_size';
      lsn
---------------
 C001/AB000100
(1 row)
```

9.28.4.

9.96################################

#9.96 ########

##	##
pg_is_in_recovery () → boolean	

##
###################
pg_last_wal_receive_lsn () → pg_lsn ## ## ################################WAL###### ############ ######################################NULL######
pg_last_wal_replay_lsn () → pg_lsn ############################## ############################## ###### ####################WAL############### ########################### #NULL######
pg_last_xact_replay_timestamp () → timestamp with time zone ################################## ############################## ################WAL############## ############################## #NULL###### ############################## ########################## ############################NULL######
pg_get_wal_resource_managers () → setof record (*rm_id* integer, *rm_name* text, *rm_builtin* boolean) ##############WAL############## #*rm_builtin*#################### ##############################

9.97########################

#9.97

##
pg_is_wal_replay_paused () → boolean ######################
pg_get_wal_replay_pause_state () → text ############## ##################not paused### ################### #########pause requested### ##################paused###
pg_promote (*wait* boolean DEFAULT true, *wait_seconds* integer DEFAULT 60) → boolean ###################### *wait*#true######################## ##*wait_seconds*#################true######false###### *wait*#false###################postmaster#SIGUSR1#########true###### ##EXECUTE##############
pg_wal_replay_pause () → void ############# ############################## ################### #### pg_get_wal_replay_pause_state()######################## pg_is_wal_replay_paused()########################## ############ ################## ## ################################### ##EXECUTE##############
pg_wal_replay_resume () → void ############### ##EXECUTE##############

pg_wal_replay_pause#pg_wal_replay_resume############### ############### ###################

####################WAL##########
######WAL##

9.28.5.

PostgreSQL## ######################
########################## ###########2##################################
##2##################################3###############2##########START
TRANSACTION###

##############PostgreSQL###
##################################
##################(###############)##
##########(###############)#################### ##############################
#################

########## 9.98###pg_export_snapshot##############SET TRANSACTION#######
#########

#9.98 ############

##
pg_export_snapshot () → text ##############################text######## ###################### ########################## ## ######## ################################### ###READ COMMITTED############## ################# REPEATABLE READ################################ ########### ############################PREPARE TRANSACTION##### ################
pg_log_standby_snapshot () → pg_lsn bgwriter#checkpointer#######################################WAL####### ### ##########

9.28.6.

9.99## ##################26.2.5#26.2.6##48
########## ##GRANT############
############ ################################REPLICATION################

53.4#########

9.28.3#9.28.4#9.28.5##########################

#9.99 ############

##
pg_create_physical_replication_slot (*slot_name* name [, *immediately_reserve* boolean, *temporary* boolean]) → record (*slot_name* name, *lsn* pg_lsn) *slot_name*############################## 2###############true####### ##########LSN################# #######LSN######################### ############### ##53.4### ####### 3#######*temporary*########true################################

##
################################### ######################### #### ######################CREATE_REPLICATION_SLOT ... PHYSICAL#### ######
pg_drop_replication_slot (*slot_name* name) → void   *slot_name*############################## ############### #DROP_REPLICATION_SLOT###### ################################### ################
pg_create_logical_replication_slot (*slot_name* name, *plugin* name [, *temporary* boolean, *twophase* boolean, *failover* boolean]) → record (*slot_name* name, *lsn* pg_lsn)   #####[68.834081]#######*plugin*####*slot_name*####################### ############ 3###########*temporary*#true####################### ############################## ######################## ######4#### #*twophase*#true################################## ################# #########CREATE_REPLICATION_SLOT ... LOGICAL######### ###### ######*plugin*######*slot_name*############################## ###### 3#####*temporary*################################## ############## ########## ##################### ######4#######*twophase*########## ## ######5#######*failover*### ## ###### ###################CREATE_REPLICATION_SLOT ... LOGICAL####### ###
pg_copy_physical_replication_slot (*src_slot_name* name, *dst_slot_name* name [, *temporary* boolean]) → record (*slot_name* name, *lsn* pg_lsn)   *src_slot_name*#####################*dst_slot_name############## ############ ######################LSN##WAL######### *temporary*### ###### *temporary*########################
pg_copy_logical_replication_slot (*src_slot_name* name, *dst_slot_name* name [, *temporary* boolean [, *plugin* name]]) → record (*slot_name* name, *lsn* pg_lsn)   *src_slot_name*#####################*dst_slot_name############### ################################## ######################LSN###### ## *temporary*#*plugin*############ ########################
pg_logical_slot_get_changes (*slot_name* name, *upto_lsn* pg_lsn, *upto_nchanges* integer, VARIADIC *options* text[]) → setof record (*lsn* pg_lsn, *xid* xid, *data* text)   ################################_############   *upto_lsn*#*upto_nchanges*#null###############WAL##########   *upto_lsn*#null################LSN####include###################   *upto_nchanges*#null## ####### ## ############ ################################synchronized_##### _slots################WAL###############
pg_logical_slot_peek_changes (*slot_name* name, *upto_lsn* pg_lsn, *upto_nchanges* integer, VARIADIC *options* text[]) → setof record (*lsn* pg_lsn, *xid* xid, *data* text)   ##################pg_logical_slot_get_changes()################ #######################

##
pg_logical_slot_get_binary_changes (*slot_name* name, *upto_lsn* pg_lsn, *upto_nchanges* integer, VARIADIC *options* text[]) → setof record (*lsn* pg_lsn, *xid* xid, *data* bytea) ###bytea###############pg_logical_slot_get_changes()########## ####
pg_logical_slot_peek_binary_changes (*slot_name* name, *upto_lsn* pg_lsn, *upto_nchanges* integer, VARIADIC *options* text[]) → setof record (*lsn* pg_lsn, *xid* xid, *data* bytea) ###bytea############pg_logical_slot_peek_changes()############ ##
pg_replication_slot_advance (*slot_name* name, *upto_lsn* pg_lsn) → record (*slot_name* name, *end_lsn* pg_lsn) ########################_############## ########################### ######## ######################### ############################### ######### ### ############### ##################synchronized_#####_slots###############WAL## #############
pg_replication_origin_create (*node_name* text) → oid ################################ID######
pg_replication_origin_drop (*node_name* text) → void ##
pg_replication_origin_oid (*node_name* text) → oid ####################ID###### ######################NULL######
pg_replication_origin_session_setup (*node_name* text) → void ## ######################## ## ######pg_replication_origin_session_reset#######
pg_replication_origin_session_reset () → void pg_replication_origin_session_setup()##########
pg_replication_origin_session_is_setup () → boolean ##################################
pg_replication_origin_session_progress (*flush* boolean) → pg_lsn ################################## #####*flush*######################### ############################
pg_replication_origin_xact_setup (*origin_lsn* pg_lsn, *origin_timestamp* timestamp with time zone) → void ###############LSN##################################### ########## ###pg_replication_origin_session_setup#####################
pg_replication_origin_xact_reset () → void pg_replication_origin_xact_setup()##########
pg_replication_origin_advance (*node_name* text, *lsn* pg_lsn) → void ################################ ################################### ##
pg_replication_origin_progress (*node_name* text, *flush* boolean) → pg_lsn

##
######################### #####*flush*################################### ###################
pg_logical_emit_message (*transactional* boolean, *prefix* text, *content* text [, *flush* boolean DEFAULT false]) → pg_lsn pg_logical_emit_message (*transactional* boolean, *prefix* text, *content* bytea [, *flush* boolean DEFAULT false]) → pg_lsn #####[58.887677]################## ############WAL################## ########### #####*transactional*################################### ## *prefix*#################### ## *content*################### ############### ########### ########### ###WAL################### ################## *transactional*################################### ## *prefix*############# ## *content*############# ######################### *flush*#######false###########WAL######### ######## *flush*#*transactional*########## ######WAL################ ###########
pg_sync_replication_slots () → void ########### ######### ###################### ########### ############## ########### ## ###47.2.3##### ##### sync_replication_slots################################## #########################

##
####################hot_standby_feedback###### ##primary_slot_name## VACUUM##############################

9.28.7.

9.100## bigint######## ######## ################OID######NULL######

#9.100

##
pg_column_size ("any") → integer ############################## ###################################
pg_column_compression ("any") → text ######################### #############NULL######
pg_column_toast_chunk_id ("any") → oid ###########TOAST#####chunk_id####### ##TOAST################## TOAST##########NULL###### 65.2#########
pg_database_size (name) → bigint pg_database_size (oid) → bigint

##	##
	######OID################################## ################### #CONNECT#######################pg_read_all_stats############## ######
pg_indexes_size (regclass) → bigint	###
pg_relation_size (*relation* regclass [, *fork* text]) → bigint	##############fork###################### ########################### ###pg_total_relation_size####pg_table_size########## ##1######## ################### 2############################## • main######################### • fsm#######################(65.3###)######### • vm#######################(65.4###)######### • init####################################
pg_size_bytes (text) → bigint	#pg_size_pretty############################## #### ##bytes#B#kB#MB#GB#TB#PB###
pg_size_pretty (bigint) → text pg_size_pretty (numeric) → text	############(####kB#MB#GB#TB#PB########)###################### ## #10########2######################1kB#1024#####1MB#1024^2 = 1048576## #########
pg_table_size (regclass) → bigint	##TOAST################## ########
pg_tablespace_size (name) → bigint pg_tablespace_size (oid) → bigint	######OID################################## ######################## ######################CREATE#########pg_read_all_stats########## #########
pg_total_relation_size (regclass) → bigint	###################TOAST##################### ###pg_table_size + pg_indexes_size######

####################regclass##############################pg_class######## #############OID### ####regclass###########################OID############ ###8.19########

9.101

#9.101

##	##
pg_relation_filenode (*relation* regclass) → oid	##################################### ############################## ####(####65.1########) ###################pg_class.relfilenode##### #######################relfilenode################################## ###################### ####################################NULL### ###

pg_relation_filepath (*relation* regclass) → text #####################################PGDATA############
pg_filenode_relation (*tablespace* oid, *filenode* oid) → regclass ###########OID######################OID###### ###### #pg_relation_filepath######### ################################## #0######## ################################NULL######

9.102#######################

#9.102

pg_collation_actual_version (oid) → text ## ## #pg_collation.collversion###################################### ## ALTER COLLATION#########
pg_database_collation_actual_version (oid) → text ## ## #pg_database.datcollversion####################################### ### ALTER DATABASE#########
pg_import_system_collations (*schema* regnamespace) → integer #####################################pg_collation############ ## #initdb###################23.2.2######### ########################## ## pg_collation####### ################### ## ############ *schema*########pg_catalog############################ ########################## ############################## ############ ##############

9.103#################################

#9.103

pg_partition_tree (regclass) → setof record (*relid* regclass, *parentrelid* regclass, *isleaf* boolean, *level* integer) 1#1### ############ ###############OID#######OID########################## ############# ##0#####1 ###############2######################## ##############################
pg_partition_ancestors (regclass) → setof regclass ###################################### ############################ ###########################
pg_partition_root (regclass) → regclass ##################################### ############################## #################NULL######

###############measurement##################################

```
SELECT pg_size_pretty(sum(pg_relation_size(relid))) AS total_size
  FROM pg_partition_tree('measurement');
```

9.28.8.

9.104########################## ## ############## ###################

#9.104 #########

##
brin_summarize_new_values (*index* regclass) → integer #####BRIN## ######### ################################### ############################
brin_summarize_range (*index* regclass, *blockNumber* bigint) → integer ################################## ###brin_summarize_new_values## ##
brin_desummarize_range (*index* regclass, *blockNumber* bigint) → void ######################################BRIN###############
gin_clean_pending_list (*index* regclass) → bigint ##GIN## ################ ###### fastupdate#############GIN################################ ################### #######fastupdate############64.4.4.1#64.4.5#### ####

9.28.9.

9.105## ### #pg_read_server_files##############################log_directory######## ############ ######################################log_directory##########

#######EXECUTE###pg_read_file()## ## ################ ####################pg_authid## ############# ######################################

psql#\df##########################

###############*missing_ok*## true###### #NULL###################### false############# ######false### ############ ################################ ######false###

#9.105 ############

##
pg_ls_dir (*dirname* text [, *missing_ok* boolean, *include_dot_dirs* boolean]) → setof text ## *include_dot_dirs*##.###..#################### #####(false)########## ##########*missing_ok*#true##

##
##########EXECUTE##########
pg_ls_logdir () → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) ##########mtime####### ########## ########## ##########pg_monitor########## EXECUTE##########
pg_ls_waldir () → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) ########WAL##########mtime####### ########## ########## ##########pg_monitor########## EXECUTE##########
pg_ls_logicalmapdir () → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) ####pg_logical/mappings##########mtime##### ## ########## ##########pg_monitor########## EXECUTE##########
pg_ls_logicalsnapdir () → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) ####pg_logical/snapshots##########mtime#### ### ########## ##########pg_monitor########## EXECUTE##########
pg_ls_replslotdir (*slot_name* text) → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) ####pg_replslot/slot_name##########mtime###### *slot_name*########## ########## ########## ##########pg_monitor##########EXECUTE## ##########
pg_ls_archive_statusdir () → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) WAL##########(pg_wal/archive_status)########## ##mtime####### ########## ##########pg_monitor########## EXECUTE##########
pg_ls_tmpdir ([*tablespace* oid]) → setof record (*name* text, *size* bigint, *modification* timestamp with time zone) #####*tablespace*##########mtime###### *tablespace*########pg_default########## ########## ########## ##########pg_monitor########## EXECUTE##########
pg_read_file (*filename* text [, *offset* bigint, *length* bigint] [, *missing_ok* boolean]) → text #####*offset*#######*length*########## ########## *offset*########## *offset*#*length*######

##	##
	############## ## ############ ########################### ##EXECUTE##############
pg_read_binary_file (*filename* text [, *offset* bigint, *length* bigint] [, *missing_ok* boolean]) → bytea	################## ###text#####bytea########################### ##pg_read_file###### ################## ##EXECUTE############## convert_from## ###### SELECT convert_from(pg_read_binary_file('file_in_utf8.txt'), 'UTF8');
pg_stat_file (*filename* text [, *missing_ok* boolean]) → record (*size* bigint, *access* timestamp with time zone, *modification* timestamp with time zone, *change* timestamp with time zone, *creation* timestamp with time zone, *isdir* boolean)	##Unix################### #Windows########################### ##EXECUTE##############

9.28.10.

9.106################## #######################13.3.5##########

############64#######2##32###false########## ##3##################################3############ #####

#9.106

##	##
pg_advisory_lock (*key* bigint) → void pg_advisory_lock (*key1* integer, *key2* integer) → void	################################
pg_advisory_lock_shared (*key* bigint) → void pg_advisory_lock_shared (*key1* integer, *key2* integer) → void	################################
pg_advisory_unlock (*key* bigint) → boolean pg_advisory_unlock (*key1* integer, *key2* integer) → boolean	############################## #############true###### ############ ##false#######SQL##############
pg_advisory_unlock_all () → void	##################################### ############################## ###################

##
pg_advisory_unlock_shared (*key* bigint) → boolean pg_advisory_unlock_shared (*key1* integer, *key2* integer) → boolean ############################## #############true###### ############ ##false#######SQL##############
pg_advisory_xact_lock (*key* bigint) → void pg_advisory_xact_lock (*key1* integer, *key2* integer) → void ##################################
pg_advisory_xact_lock_shared (*key* bigint) → void pg_advisory_xact_lock_shared (*key1* integer, *key2* integer) → void ###################################
pg_try_advisory_lock (*key* bigint) → boolean pg_try_advisory_lock (*key1* integer, *key2* integer) → boolean ########################### #############true####################### #false######
pg_try_advisory_lock_shared (*key* bigint) → boolean pg_try_advisory_lock_shared (*key1* integer, *key2* integer) → boolean ########################### #############true####################### #false######
pg_try_advisory_xact_lock (*key* bigint) → boolean pg_try_advisory_xact_lock (*key1* integer, *key2* integer) → boolean ############################## #############true#################### ####false######
pg_try_advisory_xact_lock_shared (*key* bigint) → boolean pg_try_advisory_xact_lock_shared (*key1* integer, *key2* integer) → boolean ############################## #############true#################### ####false######

9.29.

##############################PostgreSQL#################################### ##### 9.107########## ## ################ ####################

#############CREATE TRIGGER#######

#9.107

##
suppress_redundant_updates_trigger () → trigger do-nothing########## ############## CREATE TRIGGER ... suppress_redundant_updates_trigger()
tsvector_update_trigger () → trigger #################tsvector####### ######################## ###12.4.3# #######

##
###
CREATE TRIGGER ... tsvector_update_trigger(tsvcol, 'pg_catalog.swedish', title, body)
tsvector_update_trigger_column() → trigger ################tsvector####### ################regconfig###### ### ###12.4.3######## CREATE TRIGGER ... tsvector_update_trigger_column(tsvcol, tsconfigcol, title, body)

####BEFORE UPDATE#####suppress_redundant_updates_trigger############# ######################### ## #### ######################################

#################################### ################################### ######################## ## ######## ############################ ## ###### ##suppress_redundant_updates_trigger############ ###################### # ######################## ################################## ################### ##################################

suppress_redundant_updates_trigger####################

```
CREATE TRIGGER z_min_update
BEFORE UPDATE ON tablename
FOR EACH ROW EXECUTE FUNCTION suppress_redundant_updates_trigger();
```

########z###########

9.30.

PostgreSQL#######################################

################38#########

9.30.1.

pg_event_trigger_ddl_commands () → setof record

pg_event_trigger_ddl_commands#ddl_command_end######################### ###############DDL############# ############################ pg_event_trigger_ddl_commands#######################1####### 1##SQL###### ################################## ###############

##	#	##
classid	oid	###############OID
objid	oid	#############OID
objsubid	integer	#########ID###########
command_tag	text	#######
object_type	text	########

##	#	##
schema_name	text	###################### ## ##NULL# #########
object_identity	text	######################### ####### ################# ########
in_extension	boolean	##################
command	pg_ddl_command	################### ##### ######################### ###################

9.30.2. DDL##################

pg_event_trigger_dropped_objects () → setof record

##pg_event_trigger_dropped_objects########sql_drop################### ############## ###################### ##############

##	#	##
classid	oid	###############OID
objid	oid	################OID
objsubid	integer	########ID##########
original	boolean	###################
normal	boolean	######################### #######
is_temporary	boolean	#####################
object_type	text	#######
schema_name	text	###################### ### ####NULL# #########
object_name	text	######################### ######################### #########NULL# ####### ###################
object_identity	text	######################### ## ###################### #####
address_names	text[]	object_type## #address_args### #pg_get_object_address()# ######################### ######################### ######
address_args	text[]	address_names####

##pg_event_trigger_dropped_objects######################

```
CREATE FUNCTION test_event_trigger_for_drops()
        RETURNS event_trigger LANGUAGE plpgsql AS $$
DECLARE
    obj record;
```

```
BEGIN
    FOR obj IN SELECT * FROM pg_event_trigger_dropped_objects()
    LOOP
        RAISE NOTICE '% dropped object: % %.% %',
                     tg_tag,
                     obj.object_type,
                     obj.schema_name,
                     obj.object_name,
                     obj.object_identity;
    END LOOP;
END;
$$;
CREATE EVENT TRIGGER test_event_trigger_for_drops
   ON sql_drop
   EXECUTE FUNCTION test_event_trigger_for_drops();
```

9.30.3.

9.108#######table_rewrite################################ #################

#9.108 ############

##
pg_event_trigger_table_rewrite_oid () → oid ################OID######
pg_event_trigger_table_rewrite_reason () → integer ###################### ###################

##########################

```
CREATE FUNCTION test_event_trigger_table_rewrite_oid()
 RETURNS event_trigger
 LANGUAGE plpgsql AS
$$
BEGIN
  RAISE NOTICE 'rewriting table % for reason %',
                pg_event_trigger_table_rewrite_oid()::regclass,
                pg_event_trigger_table_rewrite_reason();
END;
$$;

CREATE EVENT TRIGGER test_table_rewrite_oid
                  ON table_rewrite
   EXECUTE FUNCTION test_event_trigger_table_rewrite_oid();
```

9.31.

PostgreSQL#CREATE STATISTICS##############################

9.31.1. MCV######

pg_mcv_list_items (pg_mcv_list) → setof record

pg_mcv_list_items####MCV################### #############

##	#	##
index	integer	MCV#############
values	text[]	MCV########
nulls	boolean[]	NULL########
frequency	double precision	##MCV#####
base_frequency	double precision	##MCV#######

pg_mcv_list_items##################

```
SELECT m.* FROM pg_statistic_ext join pg_statistic_ext_data on (oid =
 stxoid),
                pg_mcv_list_items(stxdmcv) m WHERE stxname = 'stts';
```

pg_mcv_list####pg_statistic_ext_data.stxdmcv##########

#10#

#############SQL#######1########################## PostgreSQL############## #################

############################## ####PostgreSQL################################## # ######################################

#####PostgreSQL#################### ################################8###9###### ##########

10.1.

SQL############# ## PostgreSQL#####SQL ################################## ######PostgreSQL############################ ######################### ##############################

PostgreSQL#####/################################5############### ########## ################## SQL## PostgreSQL######## ########################

```
SELECT text 'Origin' AS "label", point '(0,0)' AS "value";

 label  | value
--------+-------
 Origin | (0,0)
(1 row)
```

########text#point###2################## ################################ ########################unknown#########

PostgreSQL##################4######SQL#########

######

PostgreSQL############################ ################## PostgreSQL###### ## ################################### ######

###

PostgreSQL######2################1################### ################### ###########################

####

SQL#INSERT#UPDATE################# ##################################### ###

UNION#CASE#########

UNION#####SELECT#############1#######################SELECT############# ################ ####CASE#########################CASE################## ##### ARRAY[]#############GREATEST###LEAST#########################

#CREATE CAST################### ############################ ##############

####
##boolean#numeric#string#bitstring#datetime#timespan#geometric#network#
################################ ##### 51.65#####################################
####### ################################1#################### ####################
###

#######################################

- #######################################
- ### ####################
  ##
  ######
- ##
  ######################################

10.2.

################################## ##
################################## ###4.1.6##########

###########

1. pg_operator############################## ################################
   ## #5.10.3########### ###########
   ####################################

   - (Optional) ## #
     ##

2. ############################## ##################################1########
   ############ ##(########
   #####)######## [1] ##################### ######################################

   a. (Optional) ######1#####unknown################################## 2#
      #unknown#######unknown##

   b. (Optional) ######1#####unknown#######1##################################
      ############

3. ###############

   a. ## unknown#
      ########################### 1#################### ##################

   b. ## ######################
      ##################################

   c. ############################## ######################### 1################
      ##### ##################

   d. ### ##################
      ############### 1#################### ##################

   e. #####unknown## ##########
      #string######################### #unknown ############################
      ########## ## #############

[1] ##

###################################### ################################
##########################1################################### #######
#################### 1#################### #################

f. ##unknown###unknown###########
#####unknown########################### ###1######################
#############

#########

#10.1

#########double precision################1###########|/######## ########
###########integer########

```
SELECT |/ 40 AS "square root of 40";
 square root of 40
-------------------
 6.324555320336759
(1 row)
```

################################

```
SELECT |/ CAST(40 AS double precision) AS "square root of 40";
```

#10.2

################################### #########################

###############1######

```
SELECT text 'abc' || 'def' AS "text and unknown";

 text and unknown
------------------
 abcdef
(1 row)
```

############text#################### ###################text################
##

#########2########

```
SELECT 'abc' || 'def' AS "unspecified";

 unspecified
-------------
 abcdef
(1 row)
```

##########################
########################### ##
######text#######################

#10.3

PostgreSQL################@############ ############################## ##1#####
#############float8######## ######PostgreSQL##unknown##################

```
SELECT @ '-4.5' AS "abs";
 abs
-----
 4.5
(1 row)
```

#######################unknown#######float8########### ######float8########
##################

```
SELECT @ '-4.5e500' AS "abs";

ERROR:  "-4.5e500" is out of range for type double precision
```

########~######################float8########### #####~####################
######

```
SELECT ~ '20' AS "negation";

ERROR:  operator is not unique: ~ "unknown"
HINT:  Could not choose a best candidate operator. You might need to add
explicit type casts.
```

#############~####################################### ##########################
##

```
SELECT ~ CAST('20' AS int8) AS "negation";

 negation
----------
      -21
(1 row)
```

#10.4

##

```
SELECT array[1,2] <@ '{1,2,3}' as "is subset";

 is subset
-----------
 t
(1 row)
```

PostgreSQL##########<@########################## ##########################
(anyarray <@ anyarray)#####(anyelement <@ anyrange)#2###### #############
(8.21###)## ######## 3.f###############
########################### #####2##################################(###########
##)

#10.5

################################## ##
#################################### ################

```
CREATE DOMAIN mytext AS text CHECK(...);
CREATE FUNCTION mytext_eq_text (mytext, text) RETURNS boolean AS ...;
CREATE OPERATOR = (procedure=mytext_eq_text, leftarg=mytext,
 rightarg=text);
CREATE TABLE mytable (val mytext);
```

```
SELECT * FROM mytable WHERE val = 'foo';
```

```
#################### ######mytext = mytext####### 2.a####################
#####text#####text = text####### 2.b##################unknown##text######
##text = text######### ##################################
```

```
SELECT * FROM mytable WHERE val = text 'foo';
```

```
##################mytext = text############# ##############################
###### ######################################################################
##################################################
```

10.3.

```
####################################
```

```
##########
```

1. pg_proc########################## ##################################### ################################ #5.10.3########## ######################### ####################

   a. (Optional) ## ## ####################################

   b. (Optional) #####VARIADIC######################VARIADIC############### ## ##################### ##2# #########################

      ### [2], ################ ##### ####################################SQL######### VARIADIC###### ################################### VARIADIC "any"################## ########VARIADIC################# ################################ ######################

   c. (Optional) ######################################0############# ################### ###############2############################ ## ## #########2## ##ambiguous function call################

      ##[2]#################### ### ## ######## ############# ##

2. ############################ ################################1############# ####### ##[2]###### ############### ##################################### #unknown########### ##################

3. ## ############1########### ################################ ############unknown#################### ## #### ##################CAST############## [3]

[2] ##
[3] ## ############################## ################### #########CREATE CAST#########

4. ###########

   a. ## unknown## ######################### 1#################### #################

   b. ## ##################### ################################

   c. ############################ ######################### 1################ ##### #################

   d. ## ############### ################ 1#################### #################

   e. #####unknown## ###### #string######################### #unknown ########################### ######### ## ############# ##################################### ################################ ##########################1################################## ######## ################### 1#################### #################

   f. ##unknown###unknown########### ## #####unknown########################### ###1#################### ############

#################################### #########

#10.6

2#######round###1######## #1#####numeric###2#####integer####### ############## #####integer###1###numeric#######

```
SELECT round(4, 4);

 round
--------
 4.0000
(1 row)
```

```
###########################
```

```
SELECT round(CAST (4 AS numeric), 4);
```

```
#############numeric#################################################
```

```
SELECT round(4.0, 4);
```

#10.7

```
CREATE FUNCTION public.variadic_example(VARIADIC numeric[]) RETURNS int
  LANGUAGE sql AS 'SELECT 1';
CREATE FUNCTION
```

```
###############VARIADIC############# ####################
```

```
SELECT public.variadic_example(0),
       public.variadic_example(0.0),
       public.variadic_example(VARIADIC array[0.0]);
```

```
 variadic_example | variadic_example | variadic_example
------------------+------------------+------------------
                 1 |                1 |                1
(1 row)
```

#######1###2################################

```
CREATE FUNCTION public.variadic_example(numeric) RETURNS int
  LANGUAGE sql AS 'SELECT 2';
CREATE FUNCTION

CREATE FUNCTION public.variadic_example(int) RETURNS int
  LANGUAGE sql AS 'SELECT 3';
CREATE FUNCTION

SELECT public.variadic_example(0),
       public.variadic_example(0.0),
       public.variadic_example(VARIADIC array[0.0]);
 variadic_example | variadic_example | variadic_example
------------------+------------------+------------------
                 3 |                2 |                1
(1 row)
```

###########################1###2################# #####2###3################### ###### ###########VARIADIC###########3#############

#10.8

substr########## ##1##text#integer###### ################################ #string#####text#####################

```
SELECT substr('1234', 3);

 substr
--------
     34
(1 row)
```

####varchar##text################

```
SELECT substr(varchar '1234', 3);

 substr
--------
     34
(1 row)
```

#########################

```
SELECT substr(CAST (varchar '1234' AS text), 3);
```

##
####pg_cast#####text#varchar### ################### ################################

###integer##########################text###########

```
SELECT substr(1234, 3);
ERROR:  function substr(integer, integer) does not exist
HINT:  No function matches the given name and argument types. You might
 need
to add explicit type casts.
```

integer#text########################## ##########################

```
SELECT substr(CAST (1234 AS text), 3);

 substr
--------
     34
(1 row)
```

10.4.

##

###########

1. ####################

2. ################### ##2############pg_cast#####CREATE CAST############# ############### #########unknown##

3. ############################## ############################## pg_cast#####1 ############################## ##################integer#######1######### ##############attypmod####### #atttypmod############################## ######### ######################boolean############### ################# ##############################

#10.9 character#########

character(20)##

```
CREATE TABLE vv (v character(20));
INSERT INTO vv SELECT 'abc' || 'def';
SELECT v, octet_length(v) FROM vv;

          v           | octet_length
----------------------+--------------
 abcdef               |           20
(1 row)
```

##################||####text##############2##unknown#####text############# #######text################bpchar############, character############### ## ###text##bpchar## ####bpchar(bpchar, integer, boolean)## ############# #####################

10.5. UNION#CASE##########

UNION SQL################1########################## #########1##union ################## INTERSECT###EXCEPT####UNION######################## ## CASE#ARRAY#VALUES##########GREATEST#LEAST############################ #############

UNION#CASE#############

1. ###############unknown###################
2. ##[4]
3. ########unknown#######text######################### ########unknown### ###############
4. ##unknown##########################
5. ###unknown######################## #######unknown#########################[5] ### ################## ## ####################################
6. ##################### ####################################

#########

#10.10 Union##############

```
SELECT text 'a' AS "text" UNION SELECT 'b';

 text
------
 a
 b
(2 rows)
```

####unknown######'b'#text########

#10.11 ###Union#######

```
SELECT 1.2 AS "numeric" UNION SELECT 1;

 numeric
---------
       1
     1.2
(2 rows)
```

numeric######1.2#integer###1######numeric######### ################

#10.12 #####UNION#######

```
SELECT 1 AS "real" UNION SELECT CAST('2.2' AS REAL);

 real
------
    1
  2.2
(2 rows)
```

####real#####integer#################integer######real############## ##UNION########real##########

[4] ##UNION############## ### ##################

[5] #########CASE#(#####)##ELSE##############THEN########### ####################################

#10.13 ####UNION#######

```
SELECT NULL UNION SELECT NULL UNION SELECT 1;

ERROR:  UNION types text and integer cannot be matched
```

######PostgreSQL####UNION###

```
(SELECT NULL UNION SELECT NULL) UNION SELECT 1;
```

###UNION##############text############## #######UNION##text#integer#######
######################## ####UNION##############1############################

INTERSECT#EXCEPT################### ##############################1########

10.6. SELECT###

#############SELECT#####################################SQL######unknown#####
##################### #######

```
SELECT 'Hello World';
```

################################ ######PostgreSQL########text##############

SELECT#UNION(###INTERSECT####EXCEPT)##########INSERT ... SELECT#######
################################### #################1#######UNION#######2####
###################

RETURNING############SELECT##############

##
PostgreSQL 10###############SELECT#######################unknown##### ## ############################

#11#

################################## ##
#################### ##
########

11.1.

##################

```
CREATE TABLE test1 (
    id integer,
    content varchar
);
```

############################

```
SELECT content FROM test1 WHERE id = constant;
```

####################################test1######1################# test1####
##########################0##1########################### ###########id#####
##############################

################################## ##
##
##

######id################################

```
CREATE INDEX test1_id_index ON test1 (id);
```

test1_id_index##
###

#############DROP INDEX########## ###########################

################################## ##
################################
#ANALYZE############################## ######################################
######################14##########

########UPDATE#DELETE################# ########################## ##########
##

####PostgreSQL##############1####WHERE####JOIN#######################

indexed-column indexable-operator comparison-value

####indexed-column################################ indexable-operator###
######################## ######## ###comparison-value#######################
###############

####################SQL########################## ##################

comparison-value operator indexed-column

##operator###

################################## ######PostgreSQL#################### #SELECT###############INSERT#UPDATE#DELETE######################## ####### ############## ## ##### ##################

###############

11.2.

PostgreSQL###B-tree#Hash#GiST#SP-GiST#GIN#BRIN############################## ## ## ######CREATE INDEX#### ##B-tree########################### ################USING################### ####### ##############################

```
CREATE INDEX name ON table USING HASH (column);
```

11.2.1. B-Tree

B-tree## ######PostgreSQL####### ##################################B-tree###############

```
<   <=   =   >=   >
```

```
###BETWEEN#IN#####################B-tree################### ###############IS
NULL#IS NOT NULL##B-tree##################
```

```
##################LIKE#~########################################B-tree#######
########## ####col   LIKE   'foo%'###col   ~   '^foo'#########col   LIKE
'%bar'######### ##########C################################################
#################################### ###11.10######### ###ILIKE#~*##B-tree###################
##########################################
```

B-tree#################################### ################################## #############

11.2.2. Hash

################################32################## ########################### ######### ##

```
=
```

11.2.3. GiST

GiST## #########GiST####### ################################## ####PostgreSQL#####################GiST##### ##

```
<<   &<   &>   >>   <<|   &<|   |&>   |>>   @>   <@   ~=   &&
```

```
###############9.11########## #########GiST####### 64.1######### #####GiST###
####contrib################### ###64.2#########
```

GiST################################

```
SELECT * FROM places ORDER BY location <-> point '(101,456)' LIMIT 10;
```

############################## ##################################### ########### #### 64.1####################

11.2.4. SP-GiST

SP-GiST########GiST################################ SP-GiST#################### ################,kd################ ####PostgreSQL############################## ###########2######SP-GiST################

```
<<    >>    ~=    <@    <<|    |>>
```

################9.11########### ##########SP-GiST######## 64.2########## ### 64.3### #######

GiST#####SP-GiST################ ##############SP-GiST################ ## 64.2####################

11.2.5. GIN

GIN## ################################ ################################

GiST#SP-GiST###GIN##############################GIN########################### ##### ####PostgreSQL############GIN## ########

```
<@    @>    =    &&
```

################9.19########### ##########GIN######## 64.3########## #####GIN##### ##contrib################### ###64.4##########

11.2.6. BRIN

BRIN######(#############(Block Range INdex)########)########################### ############ ## GiST#SP-GiST#GIN##### ##BRIN######################BRIN#################################### ######## ### ########################### ############

```
<    <=    =    >=    >
```

##########BRIN######## 64.4########## ###64.5##########

11.3.

#############2################## #####################

```
CREATE TABLE test2 (
  major int,
  minor int,
  name varchar
);
```

#####/dev##

```
SELECT name FROM test2 WHERE major = constant AND minor = constant;
```

########major###minor#######1#######################

```
CREATE INDEX test2_mm_idx ON test2 (major, minor);
```

###B-tree#GiST#GIN###BRIN################################ ################## ##INCLUDE######################## #######INCLUDE#######32############ ##### ##PostgreSQL############## pg_config_manual.h##############

#######B-tree## ##################### ######################## ### ################ ################################### ######################### #################### ####(a, b, c)#############WHERE a = 5 AND b >= 42 AND c < 77################## a = 5##b = 42#########a = 5######################### ###### c >= 77################################### ############ a##########b### #c######################## ### #################

###GiST## ######################## ####### ## ########################### ###########GiST################

###GIN## B-tree#GiST############### ####################################

###BRIN## GIN#######B-tree#GiST### ## ##############BRIN######### ########BRIN##################pages_per_range################

#################

######################### ## ####### ##################4################################### ######################### ####11.5###11.9#########

11.4. #######ORDER BY

########ORDER BY############ PostgreSQL#####################B-tree############ ######

######ORDER BY###I/O ##################################### ################################ ORDER BY#LIMIT *n*################### ###*n*################################### ##### #ORDER BY#####################################*n*##################

########B-tree#################NULL######### #####TID########################### ####x#####################ORDER BY x########ORDER BY x ASC NULLS LAST################### ################################ORDER BY x DESC#### #####ORDER BY x DESC NULLS FIRST#NULLS FIRST#ORDER BY DESC######### ###############

###################ASC#DESC#NULLS FIRST#NULLS LAST#####################B-tree#####################

```
CREATE INDEX test2_info_nulls_low ON test2 (info NULLS FIRST);
CREATE INDEX test3_desc_index ON test3 (id DESC NULLS LAST);
```

####NULL################################ORDER BY x ASC NULLS FIRST## #ORDER BY x DESC NULLS LAST###########

4#######################################2##########ORDER BY################# ####### ### (x, y)###2##### ############## ##############ORDER BY x, y################ORDER BY x DESC, y DESC###### ####ORDER BY x ASC, y DESC################################## ########################### ##########(x ASC, y DESC)###(x DESC, y ASC)# #####################

##

11.5.

###AND############### ####(a, b)#########WHERE a = 5 AND b = 6######################## ####WHERE a = 5 OR b = 6##########################

#####PostgreSQL## ######### ######################AND#####OR######### #### WHERE x = 42 OR x = 47 OR x = 53 OR x = 99################1######x##############4############# ###### ################OR################# ######x#y#################### WHERE x = 5 AND y = 6###AND##### ############

###
################# ############################AND###OR####### ############# ########### ##################### ######################## ##################### ######### #############ORDER BY######################### ##################### ##

#######x########y############################### ###x#y#######2## # ##(x, y)######################### ## #### ####11.3########y######################## ######################### ####### ###y############################## x################################## ######## ###########x ################### ########3#################### ################ ################3############################ ########1######################## #############2######################

11.6.

###

```
CREATE UNIQUE INDEX name ON table (column [, ...]) [ NULLS [ NOT ]
 DISTINCT ];
```

####################B-tree###########

############NULL############# #####NULL######## NULLS NOT DISTINCT###################NULL############## ##

PostgreSQL## ######################
##

##
############################ ############################

11.7.

###########
########################

################################lower###############

```
SELECT * FROM test1 WHERE lower(col1) = 'value';
```

lower(col1)##

```
CREATE INDEX test1_lower_col1_idx ON test1 (lower(col1));
```

########UNIQUE#########col1##############col1######################### ##
###

################################

```
SELECT * FROM people WHERE (first_name || ' ' || last_name) = 'John
 Smith';
```

##############################

```
CREATE INDEX people_names ON people ((first_name || ' ' || last_name));
```

##########CREATE INDEX############################## ######################
################

#####[95.040000]############HOT## ##
######################
#WHERE indexedcolumn = 'constant'################################### ##
##

11.8.

################################## ################################## ########
################################## ################ ######################

##############################
###############
################################## ##
11.1#################

#11.1

############################### ############IP################### ###############
############################### ############IP#######################IP######
######################

###################

```
CREATE TABLE access_log (
    url varchar,
    client_ip inet,
    ...
);
```

#####################################

```
CREATE INDEX access_log_client_ip_ix ON access_log (client_ip)
WHERE NOT (client_ip > inet '192.168.100.0' AND
           client_ip < inet '192.168.100.255');
```

####################################

```
SELECT *
FROM access_log
WHERE url = '/index.html' AND client_ip = inet '212.78.10.32';
```

########IP####################### #########################IP##################
#############

```
SELECT *
FROM access_log
WHERE url = '/index.html' AND client_ip = inet '192.168.100.23';
```

######################
########### ##

11.2##########
############### ##
##

#11.2

################################### ##
########### ## ###################
##########

```
CREATE INDEX orders_unbilled_index ON orders (order_nr)
    WHERE billed is not true;
```

####################################

```
SELECT * FROM orders WHERE billed is not true AND order_nr < 10000;
```

##############order_nr########################### ##########

```
SELECT * FROM orders WHERE billed is not true AND amount > 5000.00;
```

####################################amount############################## #######
###

#####################################

```
SELECT * FROM orders WHERE order_nr = 3501;
```

####3501##########################

11.2## PostgreSQL##########
##################################
################ ##WHERE##########
################## PostgreSQL##
####### ## #####
######x < 1###x < 2############################## #######################WHERE#
################## ############################## ############################
####### ##################################### ######x < ?#########################
##################x < 2########

################################
11.3#########

#11.3

######################### ########subject#######target#########################1###
######### ###################### ##################


```
CREATE TABLE tests (
    subject text,
    target text,
    success boolean,
    ...
);

CREATE UNIQUE INDEX tests_success_constraint ON tests (subject, target)
    WHERE success;
```

###################################### ###IS NULL##############################
#####NULL#############

################################
####################### ## #
##PostgreSQL###
##################################

###
##############PostgreSQL##################################
############################## # 11.4###################

#11.4

#######################################


```
CREATE INDEX mytable_cat_1 ON mytable (data) WHERE category = 1;
CREATE INDEX mytable_cat_2 ON mytable (data) WHERE category = 2;
CREATE INDEX mytable_cat_3 ON mytable (data) WHERE category = 3;
...
CREATE INDEX mytable_cat_N ON mytable (data) WHERE category = N;
```

###


```
CREATE INDEX mytable_cat_data ON mytable (category, data);
```

#11.3################category######## ################################2,3########
#########################1## #############
###

##5.12###########
##

################[ston89b]#[olson93]####[seshadri95]#########

11.9.

PostgreSQL##PostgreSQL#########
######################## ##
############ ################WHERE##
################################### ##
#11.5###########################
##

##################PostgreSQL##################### #########################
################ ## ######
#######2##############

1. ## B-tree##############################
   ##### GiST#SP-GiST## ###
   ################################### ##
   ## #######GIN#####################
   ##

2. ################################### ########x#y######################z###
   ##############################

   ```
   SELECT x, y FROM tab WHERE x = 'key';
   SELECT x FROM tab WHERE x = 'key' AND y < 42;
   ```

   ##################################

   ```
   SELECT x, z FROM tab WHERE x = 'key';
   SELECT x FROM tab WHERE x = 'key' AND z < 42;
   ```

   ##

##2###
####PostgreSQL############################## #####13######################
#MVCC## ##############################
########### ## ######################
############ ##################################### PostgreSQL##################
############
############### ##
################
#######
##################################### ##################4#####################I/
O########### ######################################

#####2###
################## ##

 ## ############################
##################### ##################################PostgreSQL#############
############################### ###########INCLUDE############### #############
###############

```
SELECT y FROM tab WHERE x = 'key';
```

#######################x################## ###################

```
CREATE INDEX tab_x_y ON tab(x) INCLUDE (y);
```

y##

y## ############################
################## #######################

```
CREATE UNIQUE INDEX tab_x_y ON tab(x) INCLUDE (y);
```

x################ x#y############# #INCLUDE#######################
#UNIQUE#PRIMARY KEY################

############################
########## ## ############
##
################ ## ####
##B-tree#GiST#SP-GiST############
#######

INCLUDE########PostgreSQL##

```
CREATE INDEX tab_x_y ON tab(x, y);
```

####y#WHERE########################### ########################## ############
####11.3###################### ######################################

###########B-tree################# ##################################### #
################################B-tree## ##
##INCLUDE### ###############
################################

####x########f(x)####################
###########################

```
SELECT f(x) FROM tab WHERE f(x) < 1;
```

######f()######################## ###PostgreSQL################################
##
######f(x)#######x###
#####################################x############### #######

```
CREATE INDEX tab_f_x ON tab (f(x)) INCLUDE (x);
```

###f(x)#####################################WHERE####f(x)###################
############ ### #######
PostgreSQL#######################

11.3#################

```
CREATE UNIQUE INDEX tests_success_constraint ON tests (subject, target)
    WHERE success;
```

##

```
SELECT target FROM tests WHERE subject = 'some-subject' AND success;
```

####WHERE######success############################## ######################
###########WHERE##success
= true############################## PostgreSQL######9.6#######################
####################################

11.10.

#######################################

CREATE INDEX *name* ON *table* (*column opclass* [(*opclass_options*)] [*sort options*] [, ...]);

#################################### ####int4#####B-tree########int4_ops#####
##########int4################# ###############################
############################
################### ############2###
###########COLLATE#ASC/DESC#NULLS FIRST/NULLS
LAST##############

##

- text_pattern_ops#varchar_pattern_ops#bpchar_pattern_ops##########
  ##text#varchar#char###B-tree############## ################################
  ################# ##############C##############################LIKE#POSIX#
  ########################## #########varchar##############

  CREATE INDEX test_index ON test_table (col varchar_pattern_ops);

  ######<#<=#>####>=###
  ######### ##########*xxx*_pattern_ops#####################################
  ################## ## C##########
  ##*xxx*_pattern_ops############## C#######################################
  #############

############################

SELECT am.amname AS index_method,
       opc.opcname AS opclass_name,
       opc.opcintype::regtype AS indexed_type,
       opc.opcdefault AS is_default
    FROM pg_am am, pg_opclass opc
    WHERE opc.opcmethod = am.oid
    ORDER BY index_method, opclass_name;

####################################
############################ ### ######
####################1#########################

##

SELECT am.amname AS index_method,
       opc.opcname AS opclass_name,
       opf.opfname AS opfamily_name,
       opc.opcintype::regtype AS indexed_type,
       opc.opcdefault AS is_default
    FROM pg_am am, pg_opclass opc, pg_opfamily opf
    WHERE opc.opcmethod = am.oid AND
          opc.opcfamily = opf.oid

```
    ORDER BY index_method, opclass_name;
```

##

```
SELECT am.amname AS index_method,
       opf.opfname AS opfamily_name,
       amop.amopopr::regoperator AS opfamily_operator
    FROM pg_am am, pg_opfamily opf, pg_amop amop
    WHERE opf.opfmethod = am.oid AND
          amop.amopfamily = opf.oid
    ORDER BY index_method, opfamily_name, opfamily_operator;
```

> ###
>
> psql##\dAc#\dAf#\dAo#################################

11.11.

##
######

###########

```
CREATE TABLE test1c (
    id integer,
    content varchar COLLATE "x"
);
```

```
CREATE INDEX test1c_content_index ON test1c (content);
```

##################################

```
SELECT * FROM test1c WHERE content > constant;
```

###################
########################### ####

```
SELECT * FROM test1c WHERE content > constant COLLATE "y";
```

################################"y"################################

```
CREATE INDEX test1c_content_y_index ON test1c (content COLLATE "y");
```

11.12.

PostgreSQL###
#####################EXPLAIN########### ##########14.1###### ###27.2####
##

####
########## ############################

- #####ANALYZE############# ################################## ############
  ##################### ## #####

######ANALYZE######### ## ########24.1.3#24.1.6##########

- ######################## ## ######

  ## 100,000###1,000########################## ######100###1########################### #####100######1################1######## ######################

  ## ############ ##

- ## ########################## ######### #19.7.1########## ##########################enable_seqscan###### #########enable_nestloop################################## ################ ### # ################################### ### ###

- ##2############# ################# ### ######################### ########################## #####EXPLAIN ANALYZE#########

- #######################2############# ## ####### ################################### #19.7.2########## ############# ################### ## #ALTER TABLE###########

  ## #####PostgreSQL################## ########

#12#

12.1.

####(##########)## ######### ### ############################### ## ##

################################PostgreSQL############~,~*, LIKE,ILIKE####### ######################################

- ##################### #######satisfies####satisfy####################### ## satisfy#############satisfies############################## OR####### ##
- #########(####)##
- ##

###

  ############## ## ######### ## PostgreSQL##################### ##

  #######(lexemes)####### ################################### ########## ######## ####################(###s###es####)############### ######################### ##################### ## (## ##) PostgreSQL########## ########## ##

  ###################### ##################################### ############### ## ###########

######################### #######################

- ####################
- Ispell####################
- ####(thesaurus)###################
- Ispell#######################
- Snowball##########################

###################tsvector############################tsquery######### (8.11)# ############################(9.13)##########12.1.2#######@@###### ####### ####(12.9)###########

12.1.1. ######?

###################### ##################### ######################(#####)##### ######################### ################################

PostgreSQL###(##)#### ## ########################## ##

```
SELECT title || ' ' ||  author || ' ' ||  abstract || ' ' || body AS
 document
FROM messages
WHERE mid = 12;

SELECT m.title || ' ' || m.author || ' ' || m.abstract || ' ' || d.body AS
 document
FROM messages m, docs d
WHERE m.mid = d.did AND m.mid = 12;
```

> **##**
>
> ##################coalesce#######NULL##################NULL#########
> ######

##################################
########################
##PostgreSQL######################### #
###

##########################tsvector################## ##############
#tsvector######### — ## #########
#tsvector######################tsvector###################

12.1.2.

PostgreSQL#########tsvector(##)##tsquery(#####)######true#######@@#########
###################

```
SELECT 'a fat cat sat on a mat and ate a fat rat'::tsvector @@ 'cat &
 rat'::tsquery;
 ?column?
----------
 t

SELECT 'fat & cow'::tsquery @@ 'a fat cat sat on a mat and ate a fat
 rat'::tsvector;
 ?column?
----------
 f
```

###########tsquery##tsvector################### tsquery################
###AND, OR, NOT, FOLLOWED BY######################### (#
##8.11.2########) ####################################tsquery##########
#to_tsquery#plainto_tsquery#phraseto_tsquery###### ####################
#to_tsvector######## ############################

```
SELECT to_tsvector('fat cats ate fat rats') @@ to_tsquery('fat & rat');
 ?column?
----------
 t
```

##################################

```
SELECT 'fat cats ate fat rats'::tsvector @@ to_tsquery('fat & rat');
 ?column?
----------
```

```
 f
```

#######rats################# tsvector##############################
##rats#rat#######

###@@#####text##################################tsvector###tsquery########
####################

```
tsvector @@ tsquery
tsquery  @@ tsvector
text @@ tsquery
text @@ text
```

###2###############text @@ tsquery######to_tsvector(x) @@ y######text @@
text######to_tsvector(x) @@plainto_tsquery(y)######

tsquery######## & (AND) ## ######
| (OR) ########################## ! (NOT) ###############################
##fat & ! rat#########fat####rat#################

######tsquery### <-> (FOLLOWED BY)########## ############################
################## #######

```
SELECT to_tsvector('fatal error') @@ to_tsquery('fatal <-> error');
 ?column?
----------
 t

SELECT to_tsvector('error is not fatal') @@ to_tsquery('fatal <-> error');
 ?column?
----------
 f
```

FOLLOWED BY#######################<*N*>########## ###*N*####################
<1>#<->######<2>##############1#########################
phraseto_tsquery####################################tsquery######### #####
##

```
SELECT phraseto_tsquery('cats ate rats');
       phraseto_tsquery
-------------------------------
 'cat' <-> 'ate' <-> 'rat'

SELECT phraseto_tsquery('the cats ate the rats');
       phraseto_tsquery
-------------------------------
 'cat' <-> 'ate' <-> 'rat'
```

###################<0>#2##################################

tsquery############################## #######|##########&#####<->##!#######
###

FOLLOWED BY####################AND/OR/NOT#####FOLLOWED BY############
##################################### ######!x#x#################### ####!x
<-> y##x###########y################x############### #######x & y###x#y#
########################(x & y) <-> z##x#y######z################# ######
#######x <-> z & y <-> z#################2########x z#y z##########
(x#y###
#######################)

12.1.3.

###################### ################################## ##################
#(#######)####(synonym)###################################### #################
PostgreSQL## (psql#\dF##########
#########)

####################default_text_search_config#postgresql.conf######### ######
############postgresql.conf########### ################################
##ALTER DATABASE ... SET###### #############
#default_text_search_config#######

######################regconfig#######################
##default_text_search_config#####################

##PostgreSQL###########4#######
###################

- ################################(##############)#
- ######################################
- ################################(##############################)#
- ##

######################C#######################C###########################
################## (PostgreSQL#####contrib/#####################) ###########
#################
#########

12.2.

###

12.2.1.

#########################body#######friend#########title##################
#####

```
SELECT title
FROM pgweb
WHERE to_tsvector('english', body) @@ to_tsquery('english', 'friend');
```

########friends#friendly##

########english###

```
SELECT title
FROM pgweb
WHERE to_tsvector(body) @@ to_tsquery('friend');
```

########default_text_search_config##############

###########create#table#title###body############10##############

```
SELECT title
FROM pgweb
WHERE to_tsvector(title || ' ' || body) @@ to_tsquery('create & table')
ORDER BY last_mod_date DESC
```

```
LIMIT 10;
```

##########################NULL############coalesce###############

################
###########

12.2.2.

###############GIN######(12.9)##########

```
CREATE INDEX pgweb_idx ON pgweb USING GIN (to_tsvector('english', body));
```

2########to_tsvector################ ##########################(11.7)######
#############default_text_search_config############ ##########################
#tsvector###
################################

###########2########to_tsvector###############2########to_tsvector########
################## #####WHERE to_tsvector('english', body) @@ 'a & b'#
############WHERE to_tsvector(body) @@ 'a & b'###### ##################
####################################

##

```
CREATE INDEX pgweb_idx ON pgweb USING GIN (to_tsvector(config_name,
 body));
```

####config_name#pgweb######## ##
################# ############################## ##########################
##############################WHERE to_tsvector(config_name, body) @@
'a & b'#

#######################

```
CREATE INDEX pgweb_idx ON pgweb USING GIN (to_tsvector('english', title ||
 ' ' || body));
```

########to_tsvector#########tsvector########### ##########################
############# #####title#body####coalesce#############NULL##############
###################

```
ALTER TABLE pgweb
    ADD COLUMN textsearchable_index_col tsvector
                GENERATED ALWAYS AS (to_tsvector('english', coalesce(title,
 '') || ' ' || coalesce(body, ''))) STORED;
```

####GIN###################

```
CREATE INDEX textsearch_idx ON pgweb USING GIN (textsearchable_index_col);
```

#######################

```
SELECT title
FROM pgweb
WHERE textsearchable_index_col @@ to_tsquery('create & table')
ORDER BY last_mod_date DESC
```

```
LIMIT 10;
```

###########
#####default_text_search_config####### ######################
#to_tsvector###################### (####GIN##############GiST############
####12.9###) ############################tsvector###########################
########

12.3.

#################tsvector###############tsquery################### #######
##################
PostgreSQL##################

12.3.1.

PostgreSQL#####tsvector########to_tsvector##########

```
to_tsvector([ config regconfig, ] document text) returns tsvector
```

to_tsvector###tsvector##
###

```
SELECT to_tsvector('english', 'a fat  cat sat on a mat - it ate a fat
 rats');
                  to_tsvector
-----------------------------------------------------
 'ate':9 'cat':3 'fat':2,11 'mat':7 'rat':12 'sat':4
```

##########tsvector##a#on#it#############rats#####rat#############-######
##########

to_tsvector## ##############
(12.6)#################################### ################################
########### ########rats#rat##################rats#rat##### ##########
#(12.6.1)## ######a#on###it###### #
#################################### #########-####################(Space
symbols)## ###################
####################(12.7)########## ######################################
######## ##################english#########

##setweight####tsvector##################################A, B, C, D#########
############
###########

to_tsvector(NULL)#NULL#####NULL##############coalesce##################
####tsvector##################

```
UPDATE tt SET ti =
    setweight(to_tsvector(coalesce(title,'')), 'A')    ||
    setweight(to_tsvector(coalesce(keyword,'')), 'B')  ||
    setweight(to_tsvector(coalesce(abstract,'')), 'C') ||
    setweight(to_tsvector(coalesce(body,'')), 'D');
```

#########tsvector################setweight###########tsvector#####||###
##########tsvector#########(###12.4.1#########)

12.3.2.

PostgreSQL#######tsquery#####
#to_tsquery#plainto_tsquery#phraseto_tsquery#websearch_to_tsquery###
to_tsquery##plainto_tsquery#phraseto_tsquery#################
################### websearch_to_tsquery##web#######################
#to_tsquery##########

```
to_tsquery([ config regconfig, ] querytext text) returns tsquery
```

to_tsquery##*querytext*##tsquery########### *querytext*##tsquery###& (AND), | (OR)#! (NOT)#<-> (FOLLOWED BY)########################## ######### ########### #######to_tsquery#####8.11.2#######tsquery#################### ## #######tsquery#######################to_tsquery###

```
SELECT to_tsquery('english', 'The & Fat & Rats');
  to_tsquery
---------------
 'fat' & 'rat'
```

####tsquery################################tsvector#######################
######

```
SELECT to_tsquery('english', 'Fat | Rats:AB');
    to_tsquery
------------------
 'fat' | 'rat':AB
```

################*################

```
SELECT to_tsquery('supern:*A & star:A*B');
        to_tsquery
--------------------------
 'supern':*A & 'star':*AB
```

#####################tsvector#################

to_tsquery###
##############supernovae stars : sn#############

```
SELECT to_tsquery('''supernovae stars'' & !crab');
  to_tsquery
---------------
 'sn' & !'crab'
```

##########to_tsquery##AND#OR####FOLLOWED BY#############################
####

```
plainto_tsquery([ config regconfig, ] querytext text) returns tsquery
```

plainto_tsquery############*querytext*##tsquery######## #########
##to_tsvector############### ####tsquery###& (AND) ################

##

```
SELECT plainto_tsquery('english', 'The Fat Rats');
 plainto_tsquery
-----------------
 'fat' & 'rat'
```

plainto_tsquery#####tsquery#####################################

```
SELECT plainto_tsquery('english', 'The Fat & Rats:C');
   plainto_tsquery
---------------------
 'fat' & 'rat' & 'c'
```

#########################

phraseto_tsquery([*config* regconfig,] *querytext* text) returns tsquery

phraseto_tsquery#plainto_tsquery##################& (AND) #######<-> (FOLLOWED BY) ################ ####################<->#######<*N*>######### ############# FOLLOWED BY## ########################

#######

```
SELECT phraseto_tsquery('english', 'The Fat Rats');
 phraseto_tsquery
------------------
 'fat' <-> 'rat'
```

plainto_tsquery#####phraseto_tsquery########tsquery################### ######

```
SELECT phraseto_tsquery('english', 'The Fat & Rats:C');
      phraseto_tsquery
-----------------------------
 'fat' <-> 'rat' <-> 'c'
```

websearch_to_tsquery([*config* regconfig,] *querytext* text) returns tsquery

websearch_to_tsquery##*querytext*# #tsquery####### plainto_tsquery###phraseto_tsquery################## ### ## ##############

- ##########:##################plainto_tsquery########&##########
- "#########":##############phraseto_tsquery########<->###############
- OR:###or##|##########
- -:#####!##########

############## #####plainto_tsquery#phraseto_tsquery## ##websearch_to_tsquery###tsquery##############################

#######

```
SELECT websearch_to_tsquery('english', 'The fat rats');
 websearch_to_tsquery
----------------------
```

```
 'fat' & 'rat'
(1 row)

SELECT websearch_to_tsquery('english', '"supernovae stars" -crab');
       websearch_to_tsquery
----------------------------------
 'supernova' <-> 'star' & !'crab'
(1 row)

SELECT websearch_to_tsquery('english', '"sad cat" or "fat rat"');
       websearch_to_tsquery
-----------------------------------
 'sad' <-> 'cat' | 'fat' <-> 'rat'
(1 row)

SELECT websearch_to_tsquery('english', 'signal -"segmentation fault"');
         websearch_to_tsquery
---------------------------------------
 'signal' & !( 'segment' <-> 'fault' )
(1 row)

SELECT websearch_to_tsquery('english', '""" )( dummy \\ query <->');
 websearch_to_tsquery
----------------------
 'dummi' & 'queri'
(1 row)
```

12.3.3.

##
############ PostgreSQL##2###
##
######################### ## #
################### ##

#######################

ts_rank([*weights* float4[],] *vector* tsvector, *query* tsquery [, *normalization* integer]) returns float4

#########################

ts_rank_cd([*weights* float4[],] *vector* tsvector, *query* tsquery [, *normalization* integer]) returns float4

######1999##"Information Processing and Management"##########Clarke, Cormack, Tudhope#"Relevance Ranking for One to Three Term Queries"#########################*########(cover density)*######### ##############################ts_rank###### #######

############################### #####tsvector################## ######### ######## ###0### (strip###tsvector##################12.4.1#########)

##*weights*###################### ##

{D-weight, C-weight, B-weight, A-weight}

weights#######################

{0.1, 0.2, 0.4, 1.0}

##
##################################

###5########100####
#####5#######1000################## ##
######################################
######|#######(##2|4)#

- 0(#####):###########
- 1:####(1 + log(#####))#####
- 2:##############
- 4:#######################(###ts_rank_cd###########)
- 8: ####################
- 16: ####log(###########)+1 #####
- 32: ########+1 #####

2#####################################

##1%##100%################ ########
32(rank/(rank+1))##################0##1#####################################
##########################

##########10#############

```
SELECT title, ts_rank_cd(textsearch, query) AS rank
FROM apod, to_tsquery('neutrino|(dark & matter)') query
WHERE query @@ textsearch
ORDER BY rank DESC
LIMIT 10;
                     title                     |   rank
-----------------------------------------------+----------
 Neutrinos in the Sun                          |      3.1
 The Sudbury Neutrino Detector                 |      2.4
 A MACHO View of Galactic Dark Matter          |  2.01317
 Hot Gas and Dark Matter                       |  1.91171
 The Virgo Cluster: Hot Plasma and Dark Matter |  1.90953
 Rafting for Solar Neutrinos                   |      1.9
 NGC 4650A: Strange Galaxy and Dark Matter     |  1.85774
 Hot Gas and Dark Matter                       |   1.6123
 Ice Fishing for Cosmic Neutrinos              |      1.6
 Weak Lensing Distorts the Universe            | 0.818218
```

#######################

```
SELECT title, ts_rank_cd(textsearch, query, 32 /* rank/(rank+1) */ ) AS
 rank
FROM apod, to_tsquery('neutrino|(dark & matter)') query
WHERE  query @@ textsearch
ORDER BY rank DESC
LIMIT 10;
                     title                     |        rank
-----------------------------------------------+-------------------
 Neutrinos in the Sun                          | 0.756097569485493
 The Sudbury Neutrino Detector                 | 0.705882361190954
 A MACHO View of Galactic Dark Matter          | 0.668123210574724
 Hot Gas and Dark Matter                       |  0.65655958650282
 The Virgo Cluster: Hot Plasma and Dark Matter | 0.656301290640973
 Rafting for Solar Neutrinos                   | 0.655172410958162
 NGC 4650A: Strange Galaxy and Dark Matter     | 0.650072921219637
```

```
 Hot Gas and Dark Matter                        | 0.617195790024749
 Ice Fishing for Cosmic Neutrinos               | 0.615384618911517
 Weak Lensing Distorts the Universe             | 0.450010798361481
```

#######I/O##########################tsvector############################ #####
##

12.3.4.

###PostgreSQL########
#ts_headline##########

```
ts_headline([ config regconfig, ] document text, query tsquery
 [, options text ]) returns text
```

ts_headline## ####################
###############
##*config*#######*config*#########default_text_search_config########

options################*option=value*#######################

- MaxWords, MinWords (##): ############################ ######35#15###
- ShortWord (##): ## #######3######### ##########
- HighlightAll (###): true##################3############## ######false###
- MaxFragments (##): ################ #######0######################## 0####################(####)#
- StartSel, StopSel (###): ## ###### ##########HTML##########
- FragmentDelimiter (###): ############################## ####### ... ####

############################ ################################

#################ts_headline######*query*################################
############### #################ts_headline####################MaxWords#
##
#############
#########################MinWords####################

#######

```
SELECT ts_headline('english',
  'The most common type of search
is to find all documents containing given query terms
and return them in order of their similarity to the
query.',
  to_tsquery('english', 'query & similarity'));
                        ts_headline
------------------------------------------------------------
 containing given <b>query</b> terms                         +
 and return them in order of their <b>similarity</b> to the+
 <b>query</b>.

SELECT ts_headline('english',
  'Search terms may occur
many times in a document,
requiring ranking of the search matches to decide which
occurrences to display in the result.',
```

```
  to_tsquery('english', 'search & term'),
  'MaxFragments=10, MaxWords=7, MinWords=3, StartSel=<<, StopSel=>>');
                        ts_headline
-------------------------------------------------------------
 <<Search>> <<terms>> may occur                              +
 many times ... ranking of the <<search>> matches to decide
```

ts_headline##tsvector##

12.4.

################################

12.4.1.

12.3.1####################tsvector##################PostgreSQL##tsvector### ##########################

```
tsvector || tsvector
```

tsvector#######2############################ ###################### #### ##2############## #to_tsvector################ (##)

to_tsvector## ###############setweight## #########################setweight############

```
setweight(vector tsvector, weight "char") returns tsvector
```

setweight##A, B, C, D##########*weight*############################ (D##### ##########################) ## ############

#################################### ####################setweight###### ##

```
length(vector tsvector) returns integer
```

#######################

```
strip(vector tsvector) returns tsvector
```

###tsquery###<-> (FOLLOWED BY)###################### ##

tsvector############### 9.43######

12.4.2.

12.3.2#############tsquery##################PostgreSQL##tsquery############# ########################

```
tsquery && tsquery
```

2########AND############

tsquery || tsquery

2########OR############

!! tsquery

##################

tsquery <-> tsquery

1##################2##########################tsquery###<-> (FOLLOWED BY) ######## #######

```
SELECT to_tsquery('fat') <-> to_tsquery('cat | rat');
          ?column?
----------------------------
 'fat' <-> ( 'cat' | 'rat' )
```

tsquery_phrase(*query1* tsquery, *query2* tsquery [, *distance* integer]) returns tsquery

1#####################*distance*########2#######################
##tsquery###<*N*>######## #######

```
SELECT tsquery_phrase(to_tsquery('fat'), to_tsquery('cat'), 10);
  tsquery_phrase
------------------
 'fat' <10> 'cat'
```

numnode(*query* tsquery) returns integer

tsquery#####(#######)###########################(### > 0)################(### 0)####################

```
SELECT numnode(plainto_tsquery('the any'));
NOTICE:  query contains only stopword(s) or doesn't contain lexeme(s),
 ignored
 numnode
---------
       0

SELECT numnode('foo & bar'::tsquery);
 numnode
---------
       3
```

querytree(*query* tsquery) returns text

################tsquery###
###########################

```
SELECT querytree(to_tsquery('defined'));
 querytree
-----------
 'defin'

SELECT querytree(to_tsquery('!defined'));
 querytree
-----------
 T
```

12.4.2.1.

ts_rewrite#############tsquery#################################### ######
##############tsquery#### ### ########
####################### ########(####new york, big apple, nyc, gotham)########
###########(12.6.4)#####
########### ##
###############

ts_rewrite (*query* tsquery, *target* tsquery, *substitute* tsquery) returns tsquery

ts_rewrite ####################*query*#####*target*##*substitute*#####
##############

```
SELECT ts_rewrite('a & b'::tsquery, 'a'::tsquery, 'c'::tsquery);
 ts_rewrite
------------
 'b' & 'c'
```

ts_rewrite (*query* tsquery, *select* text) returns tsquery

#####ts_rewrite####*query*##############SQL#*select*###########
select##tsquery##2################ ###*query*###*select*################
(#####)##2#######(#####)########### #######

```
CREATE TABLE aliases (t tsquery PRIMARY KEY, s tsquery);
INSERT INTO aliases VALUES('a', 'c');

SELECT ts_rewrite('a & b'::tsquery, 'SELECT t,s FROM aliases');
 ts_rewrite
------------
 'b' & 'c'
```

###ORDER BY#############
########

####################################supernovae######

```
CREATE TABLE aliases (t tsquery primary key, s tsquery);
INSERT INTO aliases VALUES(to_tsquery('supernovae'),
 to_tsquery('supernovae|sn'));

SELECT ts_rewrite(to_tsquery('supernovae & crab'), 'SELECT * FROM
 aliases');
           ts_rewrite
--------------------------------
 'crab' & ( 'supernova' | 'sn' )
```

################################

```
UPDATE aliases
SET s = to_tsquery('supernovae|sn & !nebulae')
WHERE t = to_tsquery('supernovae');

SELECT ts_rewrite(to_tsquery('supernovae & crab'), 'SELECT * FROM
 aliases');
                 ts_rewrite
---------------------------------------------
 'crab' & ( 'supernova' | 'sn' & !'nebula' )
```

#######
#############tsquery################ ##################################

```
SELECT ts_rewrite('a & b'::tsquery,
                  'SELECT t,s FROM aliases WHERE ''a & b''::tsquery @>
 t');
 ts_rewrite
------------
 'b' & 'c'
```

12.4.3.

##
###########12.2.2##############################

tsvector##tsvector################## ##################2################################

```
tsvector_update_trigger(tsvector_column_name,
 config_name, text_column_name [, ... ])
tsvector_update_trigger_column(tsvector_column_name,
 config_column_name, text_column_name [, ... ])
```

##########1###########CREATE TRIGGER####################tsvector####### ##############

```
CREATE TABLE messages (
    title       text,
    body        text,
    tsv         tsvector
);

CREATE TRIGGER tsvectorupdate BEFORE INSERT OR UPDATE
ON messages FOR EACH ROW EXECUTE FUNCTION
tsvector_update_trigger(tsv, 'pg_catalog.english', title, body);

INSERT INTO messages VALUES('title here', 'the body text is here');

SELECT * FROM messages;
   title    |         body          |            tsv
------------+-----------------------+----------------------------
 title here | the body text is here | 'bodi':4 'text':5 'titl':1

SELECT title, body FROM messages WHERE tsv @@ to_tsquery('title & body');
   title    |         body
------------+-----------------------
 title here | the body text is here
```

################ title ###body#########################tsv#######

##############tsvector############## 2################################ tsvector_update_trigger##########2############### ######################## ####search_path######################## tsvector_update_trigger_column# ##2##########################regconfig########## ########################## ##########(text, varchar, char#####)####### ################### NULL######### (######################)#

—
####### —################## ########PL/pgSQL###########

```
CREATE FUNCTION messages_trigger() RETURNS trigger AS $$
begin
  new.tsv :=
     setweight(to_tsvector('pg_catalog.english', coalesce(new.title,'')),
 'A') ||
     setweight(to_tsvector('pg_catalog.english', coalesce(new.body,'')),
 'D');
  return new;
end
$$ LANGUAGE plpgsql;

CREATE TRIGGER tsvectorupdate BEFORE INSERT OR UPDATE
    ON messages FOR EACH ROW EXECUTE FUNCTION messages_trigger();
```

tsvector##
##default_text_search_config##
#######################

12.4.4.

ts_stat##################################

```
ts_stat(sqlquery text, [ weights text, ]
        OUT word text, OUT ndoc integer,
        OUT nentry integer) returns setof record
```

sqlquery####tsvector####SQL#############ts_stat##########tsvector######
#####(##)##########################

- *word* text — #####
- *ndoc* integer — ########(tsvector)##
- *nentry* integer — #######

weights##############################

######################10###############

```
SELECT * FROM ts_stat('SELECT vector FROM apod')
ORDER BY nentry DESC, ndoc DESC, word
LIMIT 10;
```

########A#B#######################

```
SELECT * FROM ts_stat('SELECT vector FROM apod', 'ab')
ORDER BY nentry DESC, ndoc DESC, word
LIMIT 10;
```

12.5.

##
— #######################
############################## ######PostgreSQL##############################
##################

#########pg_catalog.default######## # 12.1###23############

#12.1

##	##	#
asciiword	#######ASCII##	elephant
word	#########	mañana
numword	########	beta1
asciihword	################ ASCII	up-to-date
hword	##################	lógico-matemática
numhword	#####################	postgresql-beta1
hword_asciipart	################### ASCII	postgresql-beta1#postgresql
hword_part	#####################	lógico-matemática#lógico## #matemática
hword_numpart	##############+#####	postgresql-beta1#beta1
email	#########	foo@example.com
protocol	########	http://
url	URL	example.com/stuff/index.html
host	####	example.com
url_path	URL#####	URL##/stuff/index.html
file	##########	URL####/usr/local/foo.txt
sfloat	######	-1.234e56
float	10###	-1.234
int	######	-1234
uint	######	1234
version	#######	8.3.0
tag	XML##	<a href="dictionaries.html">>
entity	XML######	&
blank	####	(##################)

> ##
>
> ##############################lc_ctype######### ####ASCII############## ##################################### ################word # asciiword## ##########
>
> email#RFC 5322[1]#################################### ################### ####################################

##

```
SELECT alias, description, token FROM ts_debug('foo-bar-beta1');
      alias      |               description                |     token
-----------------+------------------------------------------
+---------------
```

[1] https://datatracker.ietf.org/doc/html/rfc5322

```
 numhword        | Hyphenated word, letters and digits      | foo-bar-
beta1
 hword_asciipart | Hyphenated word part, all ASCII          | foo
 blank           | Space symbols                            | -
 hword_asciipart | Hyphenated word part, all ASCII          | bar
 blank           | Space symbols                            | -
 hword_numpart   | Hyphenated word part, letters and digits | beta1
```

##

```
SELECT alias, description, token FROM ts_debug('http://example.com/stuff/
index.html');
  alias   |  description  |            token
----------+---------------+------------------------------
 protocol | Protocol head | http://
 url      | URL           | example.com/stuff/index.html
 host     | Host          | example.com
 url_path | URL path      | /stuff/index.html
```

12.6.

################(#######)############## ##
############ ################## ##################################tsvector###
###

###########

- #### — Ispell##
- ######URL####URL####################
  - http://www.pgsql.ru/db/mw/index.html
  - http://www.pgsql.ru/db/mw/
  - http://www.pgsql.ru/db/../db/mw/index.html
- ######16########### red, green, blue, magenta -> FF0000, 00FF00, 0000FF, FF00FF
- ##2########
  ##*3.14*159265359#*3.14*15926#*3.14*###########

##########################

- ####################(######################################)
- ##TSL_FILTER###############(#######
  ########################)
- ####################################
- ##################NULL

PostgreSQL###################### ## #
############################ ####################################PostgreSQL####
#contrib/#######

###
############################
################################ ####NULL####################################
##

###Snowball#####simple#
############################## #############(astro_en##)###asciiword (ASCII#
#)#########################Snowball###################

```
ALTER TEXT SEARCH CONFIGURATION astro_en
```

```
    ADD MAPPING FOR asciiword WITH astrosyn, english_ispell, english_stem;
```

##############################(####################) ##############################
################## ##############unaccent#####################################
################

12.6.1.

###################
################a#the#####################################
#tsvector##############################

```
SELECT to_tsvector('english', 'in the list of stop words');
        to_tsvector
----------------------------
 'list':3 'stop':5 'word':6
```

##1, 2, 4################### ##

```
SELECT ts_rank_cd (to_tsvector('english', 'in the list of stop words'),
 to_tsquery('list & stop'));
 ts_rank_cd
------------
       0.05

SELECT ts_rank_cd (to_tsvector('english', 'list stop words'),
 to_tsquery('list & stop'));
 ts_rank_cd
------------
        0.1
```

################################ ####ispell#################################
##Snowball######################### ##############################

12.6.2. simple##

simple###
###################### ################################## ##################
###

simple####################

```
CREATE TEXT SEARCH DICTIONARY public.simple_dict (
    TEMPLATE = pg_catalog.simple,
    STOPWORDS = english
);
```

####english####################### ###########$SHAREDIR/tsearch_data/
english.stop###$SHAREDIR##PostgreSQL##########################/usr/
local/share/postgresql#######(#########pg_config --sharedir#######)#
#########1############## ###

##############

```
SELECT ts_lexize('public.simple_dict', 'YeS');
 ts_lexize
-----------
 {yes}
```

```
SELECT ts_lexize('public.simple_dict', 'The');
 ts_lexize
-----------
 {}
```

##NULL############### #######
#Accept#####false############## #########

```
ALTER TEXT SEARCH DICTIONARY public.simple_dict ( Accept = false );

SELECT ts_lexize('public.simple_dict', 'YeS');
 ts_lexize
-----------


SELECT ts_lexize('public.simple_dict', 'The');
 ts_lexize
-----------
 {}
```

#######Accept = true###simple##
#########Accept = false##########################

> ##
>
> ##UTF-8#################
> ##########UTF-8##

> ##
>
> ##
> ##################ALTER TEXT SEARCH DICTIONARY##################
> #########################

12.6.3.

#################################### ##############(##############(12.6.4)#######
#)# ##########################Paris#######pari############### Paris paris#####
#########english_stem################### ########

```
SELECT * FROM ts_debug('english', 'Paris');
   alias   |   description   | token |  dictionaries  |  dictionary  |
 lexemes
-----------+-----------------+-------+----------------+--------------
+---------
 asciiword | Word, all ASCII | Paris | {english_stem} | english_stem |
 {pari}

CREATE TEXT SEARCH DICTIONARY my_synonym (
    TEMPLATE = synonym,
    SYNONYMS = my_synonyms
);

ALTER TEXT SEARCH CONFIGURATION english
    ALTER MAPPING FOR asciiword
    WITH my_synonym, english_stem;
```

```
SELECT * FROM ts_debug('english', 'Paris');
   alias   |   description   | token |       dictionaries        |
 dictionary | lexemes
-----------+-----------------+-------+---------------------------
+------------+---------
 asciiword | Word, all ASCII | Paris | {my_synonym,english_stem} |
 my_synonym | {paris}
```

synonym###############SYNONYMS################### — #####my_synonyms### ###########$SHAREDIR/tsearch_data/my_synonyms.syn #####(## #$SHAREDIR##PostgreSQL#######################)# ###############1#####1####### ################## ###############

synonym#########CaseSensitive#########################false### CaseSensitive#false################################## true################ ##############

######(*)########################### #################### ########### #to_tsvector()#################to_tsquery()############################## ##(####12.3.2#######) ####$SHAREDIR/tsearch_data/synonym_sample.syn###### ##############

```
postgres        pgsql
postgresql      pgsql
postgre pgsql
gogle   googl
indices index*
```

######################

```
mydb=# CREATE TEXT SEARCH DICTIONARY syn (template=synonym,
 synonyms='synonym_sample');
mydb=# SELECT ts_lexize('syn', 'indices');
 ts_lexize
-----------
 {index}
(1 row)

mydb=# CREATE TEXT SEARCH CONFIGURATION tst (copy=simple);
mydb=# ALTER TEXT SEARCH CONFIGURATION tst ALTER MAPPING FOR asciiword
 WITH syn;
mydb=# SELECT to_tsvector('tst', 'indices');
 to_tsvector
-------------
 'index':1
(1 row)

mydb=# SELECT to_tsquery('tst', 'indices');
 to_tsquery
------------
 'index':*
(1 row)

mydb=# SELECT 'indexes are very useful'::tsvector;
            tsvector
---------------------------------
 'are' 'indexes' 'useful' 'very'
(1 row)

mydb=# SELECT 'indexes are very useful'::tsvector @@ to_tsquery('tst',
 'indices');
```

```
 ?column?
----------
 t
(1 row)
```

12.6.4.

####(TZ###########)##########################(BT)#####(NT)################
###

PostgreSQL##############
######################### ################################

```
# this is a comment
sample word(s) : indexed word(s)
more sample word(s) : more indexed word(s)
...
```

#######(:)###################

#########(##########)##################################### ############### #######
################### ###################################### ######(*)###########
###

###

###?######## #####a#the####
##################

```
? one ? two : swsw
```

##a one the two#the one a two##############swsw#########

################################
############## ######################## #########asciiword#################
##one 7#################uint######################

##
###################################

12.6.4.1.

##############thesaurus##################

```
CREATE TEXT SEARCH DICTIONARY thesaurus_simple (
    TEMPLATE = thesaurus,
    DictFile = mythesaurus,
    Dictionary = pg_catalog.english_stem
);
```

####

- thesaurus_simple###########
- mythesaurus################## (######$SHAREDIR/tsearch_data/mythesaurus.ths#########$SHAREDIR######################)

- #########pg_catalog.english_stem######(#####Snowball######)# ###########(## ########)#############################

#######thesaurus_simple#######################################

```
ALTER TEXT SEARCH CONFIGURATION russian
    ALTER MAPPING FOR asciiword, asciihword, hword_asciipart
    WITH thesaurus_simple;
```

12.6.4.2.

#####################thesaurus_astro#######

```
supernovae stars : sn
crab nebulae : crab
```

####################################

```
CREATE TEXT SEARCH DICTIONARY thesaurus_astro (
    TEMPLATE = thesaurus,
    DictFile = thesaurus_astro,
    Dictionary = english_stem
);

ALTER TEXT SEARCH CONFIGURATION russian
    ALTER MAPPING FOR asciiword, asciihword, hword_asciipart
    WITH thesaurus_astro, english_stem;
```

##################ts_lexize###
#######plainto_tsquery#to_tsvector######################

```
SELECT plainto_tsquery('supernova star');
 plainto_tsquery
-----------------
 'sn'

SELECT to_tsvector('supernova star');
 to_tsvector
-------------
 'sn':1
```

###############to_tsquery######

```
SELECT to_tsquery('''supernova star''');
 to_tsquery
------------
 'sn'
```

english_stem####################supernova star#thesaurus_astro#
#supernovae stars################# #####e#s########

###

```
supernovae stars : sn supernovae stars

SELECT plainto_tsquery('supernova star');
       plainto_tsquery
----------------------------
 'sn' & 'supernova' & 'star'
```

12.6.5. Ispell##

Ispell## #######Ispell## #####bank##############banking, banked, banks, banks', bank's######

PostgreSQL########Ispell################ #########Ispell[2]######## ################ ########## — MySpell[3](OO < 2.0.1)#Hunspell[4](OO >= 2.0.2)# ######### OpenOffice Wiki[5]# #######

Ispell###################

- ################### OpenOffice##########.oxt###### .aff#####.dic########### #.affix#.dict############ ######################UTF-8#################### #####################

```
iconv -f ISO_8859-1 -t UTF-8 -o nn_no.affix nn_NO.aff
iconv -f ISO_8859-1 -t UTF-8 -o nn_no.dict nn_NO.dic
```

- #####$SHAREDIR/tsearch_data#############
- ############PostgreSQL########

```
CREATE TEXT SEARCH DICTIONARY english_hunspell (
    TEMPLATE = ispell,
    DictFile = en_us,
    AffFile = en_us,
    Stopwords = english);
```

####DictFile, AffFile, ###StopWords################################## #### ##############simple######### ##

Ispell##################################Snowball############################

Ispell#.affix###################

```
prefixes
flag *A:
    .           >   RE      # As in enter > reenter
suffixes
flag T:
    E           >   ST      # As in late > latest
    [^AEIOU]Y   >   -Y,IEST # As in dirty > dirtiest
    [AEIOU]Y    >   EST     # As in gray > grayest
    [^EY]       >   EST     # As in small > smallest
```

####.dict###################

```
lapse/ADGRS
lard/DGRS
large/PRTY
lark/MRS
```

.dict##################

```
basic_form/affix_class_name
```

[2] https://www.cs.hmc.edu/~geoff/ispell.html
[3] https://en.wikipedia.org/wiki/MySpell
[4] https://hunspell.github.io/
[5] https://wiki.openoffice.org/wiki/Dictionaries

.affix##########affix(##)#####################

```
condition > [-stripping_letters,] adding_affix
```

####condition(##)##################### [...]###[^...]################ ##
#[AEIOU]Y##########"y"########"a"#"e"#"i"#"o"#"u"################ [^EY]#
######"e"##"y"############

Ispell################################ ##############################
#compoundwords controlled################################

```
compoundwords  controlled z
```

#################

```
SELECT ts_lexize('norwegian_ispell',
 'overbuljongterningpakkmesterassistent');
   {over,buljong,terning,pakk,mester,assistent}
SELECT ts_lexize('norwegian_ispell', 'sjokoladefabrikk');
   {sjokoladefabrikk,sjokolade,fabrikk}
```

MySpell########Hunspell######## Hunspell#.affix####################

```
PFX A Y 1
PFX A   0     re         .
SFX T N 4
SFX T   0     st         e
SFX T   y     iest       [^aeiou]y
SFX T   0     est        [aeiou]y
SFX T   0     est        [^ey]
```

##(affix)####1######## ########################

- #######PFX###SFX#
- ############
- ################################
- #######
- ################

.dict#####Ispell#.dict##############

```
larder/M
lardy/RT
large/RSPMYT
largehearted
```

> ##
>
> MySpell################ Hunspell#################### #######PostgreSQL#
> Hunspell######################

12.6.6. Snowball##

Snowball####################################Martin Porter################ Snowball
##########################(###Snowball####[6]#########)# ########################

[6] https://snowballstem.org/

##################################### Snowball#################################
(PostgreSQL##################
Snowball project###########) #######################

```
CREATE TEXT SEARCH DICTIONARY english_stem (
    TEMPLATE = snowball,
    Language = english,
    StopWords = english
);
```

################################

Snowball## #################
##Snowball#######################

12.7.

############tsvector###
######### to_tsvector###to_tsquery################################ ######
##default_text_search_config##
postgresql.conf###############SET##############

###########################SQL######
##################################psql################(12.10)#

#########english#############pg#######

```
CREATE TEXT SEARCH CONFIGURATION public.pg ( COPY = pg_catalog.english );
```

PostgreSQL###############$SHAREDIR/tsearch_data/pg_dict.syn#############
###########

```
postgres    pg
pgsql       pg
postgresql  pg
```

################

```
CREATE TEXT SEARCH DICTIONARY pg_dict (
    TEMPLATE = synonym,
    SYNONYMS = pg_dict
);
```

###Ispell###english_ispell#######################

```
CREATE TEXT SEARCH DICTIONARY english_ispell (
    TEMPLATE = ispell,
    DictFile = english,
    AffFile = english,
    StopWords = english
);
```

####pg##################

```
ALTER TEXT SEARCH CONFIGURATION pg
    ALTER MAPPING FOR asciiword, asciihword, hword_asciipart,
                      word, hword, hword_part
```

```
    WITH pg_dict, english_ispell, english_stem;
```

##

```
ALTER TEXT SEARCH CONFIGURATION pg
    DROP MAPPING FOR email, url, url_path, sfloat, float;
```

######################

```
SELECT * FROM ts_debug('public.pg', '
PostgreSQL, the highly scalable, SQL compliant, open source object-
relational
database management system, is now undergoing beta testing of the next
version of our software.
');
```

################################public##############

```
=> \dF
   List of text search configurations
 Schema  | Name | Description
---------+------+-------------
 public  | pg   |

SET default_text_search_config = 'public.pg';
SET

SHOW default_text_search_config;
 default_text_search_config
----------------------------
 public.pg
```

12.8.

####################################
####################################

12.8.1.

ts_debug##########################

```
ts_debug([ config regconfig, ] document text,
         OUT alias text,
         OUT description text,
         OUT token text,
         OUT dictionaries regdictionary[],
         OUT dictionary regdictionary,
         OUT lexemes text[])
         returns setof record
```

ts_debug#####################*document*#####################*config*########
#############default_text_search_config#######

ts_debug####################1########### ############

- *alias* text — ########
- *description* text — #######

- *token* text — #######
- *dictionaries* regdictionary[] — ####################
- *dictionary* regdictionary — ####################### NULL
- *lexemes* text[] — ################################NULL#####({})############ ###############

##########

```
SELECT * FROM ts_debug('english', 'a fat  cat sat on a mat - it ate a fat
 rats');
   alias   |   description   | token |  dictionaries  |  dictionary  |
 lexemes
-----------+-----------------+-------+----------------+--------------
+---------
 asciiword | Word, all ASCII | a     | {english_stem} | english_stem | {}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | fat   | {english_stem} | english_stem |
 {fat}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | cat   | {english_stem} | english_stem |
 {cat}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | sat   | {english_stem} | english_stem |
 {sat}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | on    | {english_stem} | english_stem | {}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | a     | {english_stem} | english_stem | {}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | mat   | {english_stem} | english_stem |
 {mat}
 blank     | Space symbols   |       | {}             |              |
 blank     | Space symbols   | -     | {}             |              |
 asciiword | Word, all ASCII | it    | {english_stem} | english_stem | {}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | ate   | {english_stem} | english_stem |
 {ate}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | a     | {english_stem} | english_stem | {}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | fat   | {english_stem} | english_stem |
 {fat}
 blank     | Space symbols   |       | {}             |              |
 asciiword | Word, all ASCII | rats  | {english_stem} | english_stem |
 {rat}
```

#########################public.english####Ispell########

```
CREATE TEXT SEARCH CONFIGURATION public.english ( COPY =
 pg_catalog.english );

CREATE TEXT SEARCH DICTIONARY english_ispell (
    TEMPLATE = ispell,
    DictFile = english,
    AffFile = english,
    StopWords = english
);

ALTER TEXT SEARCH CONFIGURATION public.english
   ALTER MAPPING FOR asciiword WITH english_ispell, english_stem;
```

```
SELECT * FROM ts_debug('public.english', 'The Brightest supernovaes');
   alias   |   description   |    token    |         dictionaries
 |   dictionary   |   lexemes
-----------+-----------------+-------------
+-------------------------------+----------------+-------------
 asciiword | Word, all ASCII | The         | {english_ispell,english_stem}
 | english_ispell | {}
 blank     | Space symbols   |             | {}
 |                |
 asciiword | Word, all ASCII | Brightest   | {english_ispell,english_stem}
 | english_ispell | {bright}
 blank     | Space symbols   |             | {}
 |                |
 asciiword | Word, all ASCII | supernovaes | {english_ispell,english_stem}
 | english_stem   | {supernova}
```

#######Brightest##ASCII word (###asciiword)########### ####
############english_ispell #english_stem#######english_ispell######
#bright########## ##supernovaes#english_ispell##########################
(#####english_stem#Snowball#######################################)#

##The##english_ispell############(12.6.1)##################################
##########################

##################################

```
SELECT alias, token, dictionary, lexemes
FROM ts_debug('public.english', 'The Brightest supernovaes');
   alias   |    token    |   dictionary   |   lexemes
-----------+-------------+----------------+-------------
 asciiword | The         | english_ispell | {}
 blank     |             |                |
 asciiword | Brightest   | english_ispell | {bright}
 blank     |             |                |
 asciiword | supernovaes | english_stem   | {supernova}
```

12.8.2.

##################################

```
ts_parse(parser_name text, document text,
         OUT tokid integer, OUT token text) returns setof record
ts_parse(parser_oid oid, document text,
         OUT tokid integer, OUT token text) returns setof record
```

ts_parse#####*document*##################1##1##############################
##########tokid#############token####### #######

```
SELECT * FROM ts_parse('default', '123 - a number');
 tokid | token
-------+--------
    22 | 123
    12 |
    12 | -
     1 | a
    12 |
     1 | number
```

```
ts_token_type(parser_name text, OUT tokid integer,
              OUT alias text, OUT description text) returns setof record
ts_token_type(parser_oid oid, OUT tokid integer,
              OUT alias text, OUT description text) returns setof record
```

ts_token_type###
#######tokid##################alias########description####### #######

```
SELECT * FROM ts_token_type('default');
 tokid |      alias      |               description
-------+-----------------+-----------------------------------------
     1 | asciiword       | Word, all ASCII
     2 | word            | Word, all letters
     3 | numword         | Word, letters and digits
     4 | email           | Email address
     5 | url             | URL
     6 | host            | Host
     7 | sfloat          | Scientific notation
     8 | version         | Version number
     9 | hword_numpart   | Hyphenated word part, letters and digits
    10 | hword_part      | Hyphenated word part, all letters
    11 | hword_asciipart | Hyphenated word part, all ASCII
    12 | blank           | Space symbols
    13 | tag             | XML tag
    14 | protocol        | Protocol head
    15 | numhword        | Hyphenated word, letters and digits
    16 | asciihword      | Hyphenated word, all ASCII
    17 | hword           | Hyphenated word, all letters
    18 | url_path        | URL path
    19 | file            | File or path name
    20 | float           | Decimal notation
    21 | int             | Signed integer
    22 | uint            | Unsigned integer
    23 | entity          | XML entity
```

12.8.3.

ts_lexize################

```
ts_lexize(dict regdictionary, token text) returns text[]
```

ts_lexize####*token*###
######NULL######

#:

```
SELECT ts_lexize('english_stem', 'stars');
 ts_lexize
-----------
 {star}

SELECT ts_lexize('english_stem', 'a');
 ts_lexize
-----------
 {}
```

> ##
>
> ts_lexize###

```
SELECT ts_lexize('thesaurus_astro', 'supernovae stars') is null;
 ?column?
----------
 t

####thesaurus_astro###supernovae         stars#######ts_lexize##
#####################################   ############plainto_tsquery##
#to_tsvector###############


SELECT plainto_tsquery('supernovae stars');
 plainto_tsquery
-----------------
 'sn'
```

12.9.

#############2############## GIN#GiST### ################################## ################

##############################

```
CREATE INDEX name ON table USING GIN (column);
```

GIN (Generalized Inverted Index)########### *column*#tsvector###########

```
CREATE INDEX name ON table USING GIST (column [ { DEFAULT | tsvector_ops } (siglen = number) ] );
```

GiST (Generalized Search Tree)###########*column*# tsvector ###tsquery #### #### #########siglen################(##############)#

GIN############################## ##################################(###)#### ############# ### GIN#### ###tsvector####(###)######################### ############################## ######

GiST## (PostgreSQL#####################) GiST#################################### ######### ##########################siglen######## (siglen########)######## ###124#############2024###### ################################n########OR##n ######################## 2############################## ####################(# ##################)#################################### ######################## ######(##################################)#############

GiST#####################INCLUDE####### ####GiST################INCLUDE##### ####### ###################

####################GiST######### ######### ###

GIN###########maintenance_work_mem#####################################GiST## ###########################

#################GIN#GiST## ## ######################################(1)###########(2)######################## ######################## 2################################

12.10. psql####

psql###

```
\dF{d,p,t}[+] [PATTERN]
```

```
######+#################
```

###########PATTERN######################### *#######################* *PATTERN#*
############################ PATTERN###
########### ################

```
=> \dF *fulltext*
       List of text search configurations
 Schema |  Name        | Description
--------+--------------+-------------
 public | fulltext_cfg |
```

```
=> \dF *.fulltext*
       List of text search configurations
 Schema   |  Name        | Description
----------+----------------------------
 fulltext | fulltext_cfg |
 public   | fulltext_cfg |
```

```
###############
```

\dF[+] [PATTERN]

   ##############(+#######)#

```
=> \dF russian
            List of text search configurations
   Schema   |  Name   |            Description
------------+---------+-------------------------------------
 pg_catalog | russian | configuration for russian language

=> \dF+ russian
Text search configuration "pg_catalog.russian"
Parser: "pg_catalog.default"
      Token      | Dictionaries
-----------------+--------------
 asciihword      | english_stem
 asciiword       | english_stem
 email           | simple
 file            | simple
 float           | simple
 host            | simple
 hword           | russian_stem
 hword_asciipart | english_stem
 hword_numpart   | simple
 hword_part      | russian_stem
 int             | simple
 numhword        | simple
 numword         | simple
 sfloat          | simple
 uint            | simple
 url             | simple
 url_path        | simple
 version         | simple
```

```
 word            | russian_stem
```

\dFd[+] [PATTERN]

```
##############(+#######)#


=> \dFd
                                List of text search dictionaries
   Schema   |      Name       |                        Description
------------+-----------------
+-----------------------------------------------------------
 pg_catalog | arabic_stem     | snowball stemmer for arabic language
 pg_catalog | armenian_stem   | snowball stemmer for armenian language
 pg_catalog | basque_stem     | snowball stemmer for basque language
 pg_catalog | catalan_stem    | snowball stemmer for catalan language
 pg_catalog | danish_stem     | snowball stemmer for danish language
 pg_catalog | dutch_stem      | snowball stemmer for dutch language
 pg_catalog | english_stem    | snowball stemmer for english language
 pg_catalog | finnish_stem    | snowball stemmer for finnish language
 pg_catalog | french_stem     | snowball stemmer for french language
 pg_catalog | german_stem     | snowball stemmer for german language
 pg_catalog | greek_stem      | snowball stemmer for greek language
 pg_catalog | hindi_stem      | snowball stemmer for hindi language
 pg_catalog | hungarian_stem  | snowball stemmer for hungarian language
 pg_catalog | indonesian_stem | snowball stemmer for indonesian
 language
 pg_catalog | irish_stem      | snowball stemmer for irish language
 pg_catalog | italian_stem    | snowball stemmer for italian language
 pg_catalog | lithuanian_stem | snowball stemmer for lithuanian
 language
 pg_catalog | nepali_stem     | snowball stemmer for nepali language
 pg_catalog | norwegian_stem  | snowball stemmer for norwegian language
 pg_catalog | portuguese_stem | snowball stemmer for portuguese
 language
 pg_catalog | romanian_stem   | snowball stemmer for romanian language
 pg_catalog | russian_stem    | snowball stemmer for russian language
 pg_catalog | serbian_stem    | snowball stemmer for serbian language
 pg_catalog | simple          | simple dictionary: just lower case and
 check for stopword
 pg_catalog | spanish_stem    | snowball stemmer for spanish language
 pg_catalog | swedish_stem    | snowball stemmer for swedish language
 pg_catalog | tamil_stem      | snowball stemmer for tamil language
 pg_catalog | turkish_stem    | snowball stemmer for turkish language
 pg_catalog | yiddish_stem    | snowball stemmer for yiddish language
```

\dFp[+] [PATTERN]

```
###############(+#######)#


=> \dFp
        List of text search parsers
   Schema   |  Name   |     Description
------------+---------+---------------------
 pg_catalog | default | default word parser
=> \dFp+
    Text search parser "pg_catalog.default"
     Method      |    Function    | Description
-----------------+----------------+-------------
 Start parse     | prsd_start     |
 Get next token  | prsd_nexttoken |
 End parse       | prsd_end       |
 Get headline    | prsd_headline  |
 Get token types | prsd_lextype   |
```

```
          Token types for parser "pg_catalog.default"
     Token name    |               Description
-----------------+-----------------------------------------
 asciihword      | Hyphenated word, all ASCII
 asciiword       | Word, all ASCII
 blank           | Space symbols
 email           | Email address
 entity          | XML entity
 file            | File or path name
 float           | Decimal notation
 host            | Host
 hword           | Hyphenated word, all letters
 hword_asciipart | Hyphenated word part, all ASCII
 hword_numpart   | Hyphenated word part, letters and digits
 hword_part      | Hyphenated word part, all letters
 int             | Signed integer
 numhword        | Hyphenated word, letters and digits
 numword         | Word, letters and digits
 protocol        | Protocol head
 sfloat          | Scientific notation
 tag             | XML tag
 uint            | Unsigned integer
 url             | URL
 url_path        | URL path
 version         | Version number
 word            | Word, all letters
(23 rows)
```

\dFt[+] [PATTERN]

##################(+#######)#

```
=> \dFt
                           List of text search templates
   Schema   |   Name    |                        Description
------------+-----------
+-------------------------------------------------------------
 pg_catalog | ispell    | ispell dictionary
 pg_catalog | simple    | simple dictionary: just lower case and check
 for stopword
 pg_catalog | snowball  | snowball stemmer
 pg_catalog | synonym   | synonym dictionary: replace word by its
 synonym
 pg_catalog | thesaurus | thesaurus dictionary: phrase by phrase
 substitution
```

12.11.

PostgreSQL###################

- #########2################
- tsvector### (### + ##)#1M##############
- ######2^{64}###########
- tsvector######0#######16,383###########
- tsquery###<*N*>##########16,384##############
- ########256###########
- tsquery######(### + ###)#32,768###########

##############PostgreSQL8.1 ########10,441##################335,420#######
##postgresql##655#####6,127########

— PostgreSQL###############910,989############461,020########57,491,343######
#####

#13#

#######2#############################PostgreSQL############################ ###### ### ######################### ########################

13.1.

PostgreSQL### ########################## MVCC############### ################################SQL#################### ########################## ## MVCC#################### ##

#############MVCC#############MVCC#######################################
(SSI)#####################################PostgreSQL############

###PostgreSQL############ ################ #####MVCC######################### ########################### ##############################

13.2.

SQL######################4############## ################################## ## 1###

#####	#######	########	########	#####
#########	##	#####	#####	#####
#########	##	##	##### ##PostgreSQL### ####	#####
########	##	##	##	##

PostgreSQL###4################################ ########3################### ## ##PostgreSQL################################### ####PostgreSQL################## ##############################

##########PostgreSQL## SQL################ ############################## ############## ##############################

############################SET TRANSACTION##############

> **##**
>
> #####PostgreSQL################################## ################## ##serial## ######### 9.17###8.1.4#########

13.2.1.

PostgreSQL######################### ######################SELECT######FOR UPDATE/SHARE## ## ######SELECT################### ################################ ###SELECT################################### ################ ##################SELECT##2###############SELECT########2## #SELECT##SELECT######################## #########

UPDATE#DELETE#SELECT FOR UPDATE####SELECT FOR SHARE######## ######SELECT############### ## #### ## ####### ## ################### #########2######################### ################2######################### ############## ## #########WHERE####### ################################## ##############2############################ SELECT FOR UPDATE###SELECT FOR SHARE################################### #########

ON CONFLICT DO UPDATE####INSERT########### ############################## ###### ###################2#######1######### #######INSERT##################### ##UPDATE##############

ON CONFLICT DO NOTHING####INSERT###INSERT############################## ############################## ##############################

#####[80.904523]#MERGE###########INSERT#UPDATE###DELETE################### ## INSERT###UPDATE##########MERGE######ON CONFLICT DO UPDATE## ####INSERT#######INSERT###UPDATE######################### MERGE#UPDATE## #DELETE###MERGE#UPDATE## #DELETE## ####MERGE################### ## ################### ################## ##############################MERGE####NOT MATCHED####

######################### MERGE#INSERT#################################
############MERGE#MATCHED############################## ##################
#INSERT############################## INSERT#######################ON###
###DO######INSERT#######INSERT############################### ##########
###
#################### ##
###
##################NOT MATCHED BY####NOT MATCHED[BY#####]#############
#################### ####################NOT MATCHED[BY#####]##########
################# ####INSERT## ##
###MATCHED##################################

##############################
###########################
############################## ############################### ###############
####################

BEGIN;
UPDATE accounts SET balance = balance + 100.00 WHERE acctnum = 12345;
UPDATE accounts SET balance = balance - 100.00 WHERE acctnum = 7534;
COMMIT;

2#######################12345################################2################
################# ##
##########

#############DELETE############
################# ####website#2#######website.hits####9#10#######

BEGIN;
UPDATE website SET hits = hits + 1;

-- ######### DELETE FROM website WHERE hits = 10; ######
COMMIT;

UPDATE######website.hits = 10###########DELETE######
##################9#########UPDATE#####DELETE##################10###
#11###########################

###
##########################
#################

##########################
############## ##
##########

13.2.2.

##########################
############################### ##
######### ###SQL#################################### 13.1################# ####
###

###
####################SELECT##########
###

###

UPDATE#DELETE#MERGE#SELECT FOR UPDATE####SELECT FOR SHARE######SELECT# ############## ## ############# ## ################### ## #################### ## ################################## ##

```
ERROR:  could not serialize access due to concurrent update
```

###
#####

2### ##

################################## ################################

##1# ############# ## ################################## ## # ### ######################### ##################################### 2####################SQL################## ######################### ###########[berenson95]##########

> ##
>
> PostgreSQL version 9.1## ##### ##

13.2.3.

##################################### ## ## ######################### ## ############################## ## ######## ## ##########################

```
##############mytab##############

 class | value
-------+-------
     1 |    10
     1 |    20
     2 |   100
     2 |   200
```

##################A########

```
SELECT SUM(value) FROM mytab WHERE class = 1;
```

```
####value######30###class = 2################ ##################B########
```

```
SELECT SUM(value) FROM mytab WHERE class = 2;
```

####300###########class = 1################ ######################### ####### ## ##################################### ##

```
ERROR:  could not serialize access due to read/write dependencies among
 transactions
```

########A#B###############B####300####330#########################A########### #########

######################################

############PostgreSQL############# ### ## PostgreSQL##################### ######################### ### ############### ## ## ##############SELECT FOR UPDATE####SELECT FOR SHARE############

PostgreSQL## ### ##pg_locks#######mode#SIReadLock########### ############################ ################ ## ### ###################### #########READ ONLY##################SIRead################ ###READ ONLY# ## SERIALIZABLE READ ONLY DEFERRABLE## ######################### ## ####SIRead########################### ##

#################################### ##################################### ## ## ############## ######################SQLSTATE##'40001'###################### ####### ###################/## ############## ####/## ######## #############SELECT FOR UPDATE###SELECT FOR SHARE####################### ##

PostgreSQL## ##################################### ##################################### ### ######################### ## ####################select## ##1########################### ######## ## ################################

###

- ##############READ ONLY###########

- ##################################### ##################################### ############################

- ##################1#######################
- ## ##########################
  #idle_in_transaction_session_timeout##########
- ##SELECT FOR UPDATE###SELECT FOR SHARE##########
- ##
  ################## ##
  ##max_pred_locks_per_transaction#max_pred_locks_per_relation#max_pred_locks_per_page##
  ###########################
- ###################################### ###########################
  random_page_cost##########cpu_tuple_cost#######################################
  ## ##

##
##############################
############### ######[ports12]##########

13.3.

PostgreSQL## ########MVCC######
############################### ########PostgreSQL##############################
#####TRUNCATE###############
##ACCESS EXCLUSIVE##########

###############################pg_locks################ ####################
####27##########

13.3.1.

#######################PostgreSQL################## ###LOCK###################
############# ####################row###
###
13.2########## 2########
#################################### ########################### ####ACCESS
EXCLUSIVE#################ACCESS SHARE################## ##################
############### #######################ACCESS EXCLUSIVE##################
#######################ACCESS SHARE###############################

#############

ACCESS SHARE (AccessShareLock)

ACCESS EXCLUSIVE##############

SELECT#################################### ##############################
#################

ROW SHARE (RowShareLock)

EXCLUSIVE###ACCESS EXCLUSIVE############

#####FOR ...##############################ACCESS SHARE######SELECT###
###FOR UPDATE#FOR NO KEY UPDATE#FOR SHARE####FOR KEY SHARE#######
##################################

ROW EXCLUSIVE (RowExclusiveLock)

SHARE#SHARE ROW EXCLUSIVE#EXCLUSIVE####ACCESS EXCLUSIVE############

UPDATE#DELETE#INSERT###MERGE#######################ACCESS SHARE#####
############################ #######################################

SHARE UPDATE EXCLUSIVE (ShareUpdateExclusiveLock)

SHARE UPDATE EXCLUSIVE#SHARE#SHARE ROW EXCLUSIVE#EXCLUSIVE###
#ACCESS EXCLUSIVE############ ########################VACUUM#########
###########

#FULL####VACUUM#ANALYZE#CREATE INDEX CONCURRENTLY#CREATE
STATISTICS#COMMENT ON#REINDEX CONCURRENTLY#####ALTER INDEX#ALTER
TABLE####################################

SHARE (ShareLock)

ROW EXCLUSIVE#SHARE UPDATE EXCLUSIVE#SHARE ROW
EXCLUSIVE#EXCLUSIVE####ACCESS EXCLUSIVE############ #################
##############

#CONCURRENTLY####CREATE INDEX##########

SHARE ROW EXCLUSIVE (ShareRowExclusiveLock)

ROW EXCLUSIVE#SHARE UPDATE EXCLUSIVE# SHARE#SHARE ROW
EXCLUSIVE#EXCLUSIVE####ACCESS EXCLUSIVE############ #######1#########
#######################################

CREATE TRIGGER#####ALTER TABLE#################

EXCLUSIVE (ExclusiveLock)

ROW SHARE#ROW EXCLUSIVE# SHARE UPDATE EXCLUSIVE#SHARE# SHARE
ROW EXCLUSIVE#EXCLUSIVE####ACCESS EXCLUSIVE############ ############
##ACCESS SHARE######## ##
##

REFRESH MATERIALIZED VIEW CONCURRENTLY#########

ACCESS EXCLUSIVE (AccessExclusiveLock)

###########ACCESS SHARE#ROW SHARE#ROW EXCLUSIVE#SHARE UPDATE
EXCLUSIVE#SHARE#SHARE ROW EXCLUSIVE#EXCLUSIVE#### ACCESS EXCLUSIVE#
####### ##

DROP TABLE#TRUNCATE#REINDEX#CLUSTER#VACUUM FULL##CONCURRENTLY##
##REFRESH MATERIALIZED VIEW############## ALTER INDEX#ALTER TABLE##
#################### ##################LOCK TABLE################

###
ACCESS EXCLUSIVE######SELECT#FOR UPDATE/SHARE############

################################ ##
############# ####ROLLBACK## PL/
pgSQL######################### ################################

#13.2 #########

#########	#########							
	ACCESS SHARE	**ROW SHARE**	**ROW EXCL.**	**SHARE UPDATE EXCL.**	**SHARE**	**SHARE ROW EXCL.**	**EXCL.**	**ACCESS EXCL.**
ACCESS SHARE								X
ROW SHARE							X	X
ROW EXCL.					X	X	X	X
SHARE UPDATE EXCL.				X	X	X	X	X
SHARE			X	X		X	X	X
SHARE ROW EXCL.			X	X	X	X	X	X
EXCL.		X	X	X	X	X	X	X
ACCESS EXCL.	X	X	X	X	X	X	X	X

13.3.2.

##########################PostgreSQL############################## ########## ######### 13.3######### ## ##### ## ######################### #### ################################ ##

##########

FOR UPDATE

FOR UPDATE###SELECT############################## ######################### ################################## ##############UPDATE#DELETE#SELECT FOR UPDATE#SELECT FOR NO KEY UPDATE#SELECT FOR SHARE#SELECT FOR KEY SHARE##SELECT FOR UPDATE###(###### ###########)# ####REPEATABLE READ####SERIALIZABLE#################### ############################## ########13.4##########

FOR UPDATE###########DELETE############UPDATE######## ######UPDATE### ## #############

FOR NO KEY UPDATE

#############FOR UPDATE##################################SELECT FOR KEY SHARE############## #########FOR UPDATE#########UPDATE###########

FOR SHARE

##################################FOR NO KEY UPDATE############ ######## ##################UPDATE#DELETE#SELECT FOR UPDATE#SELECT FOR NO KEY

UPDATE###############SELECT FOR SHARE#SELECT FOR KEY SHARE#######
######

FOR KEY SHARE

############FOR SHARE############SELECT FOR UPDATE##########SELECT FOR NO KEY UPDATE########## ###################DELETE########UPDATE# ###################UPDATE##SELECT FOR NO KEY UPDATE#SELECT FOR SHARE#SELECT FOR KEY SHARE#######

PostgreSQL## ###################### ############## ###SELECT FOR UPDATE## ###############

#13.3 ##########

##########	#########			
	FOR KEY SHARE	**FOR SHARE**	**FOR NO KEY UPDATE**	**FOR UPDATE**
FOR KEY SHARE				X
FOR SHARE			X	X
FOR NO KEY UPDATE		X	X	X
FOR UPDATE	X	X	X	X

13.3.3.

#####################/## ################################## ## #####################

13.3.4.

################################ #########2#################################### ########################## ############1#####A##################B############ #############2#######B######################A############################## ########## PostgreSQL## ##

################################ ## 2###### ############################## 1####################

```
UPDATE accounts SET balance = balance + 100.00 WHERE acctnum = 11111;
```

############################## ##2####################

```
UPDATE accounts SET balance = balance + 100.00 WHERE acctnum = 22222;
UPDATE accounts SET balance = balance - 100.00 WHERE acctnum = 11111;
```

1###UPDATE## ####2###UPDATE############ ## #######2######################1###### ######### ##########1############

```
UPDATE accounts SET balance = balance - 100.00 WHERE acctnum = 22222;
```

#######1################################## #######2################### ###### #######2################## ############1########2################2#########1## ##################### PostgreSQL################################

####
##########################
##############################
###

###
###

13.3.5.

PostgreSQL################################## ################################
#################################### #######MVCC##########################
#######
########################
############

PostgreSQL#########################2################## #######################
########################### ## ##
##################
################### ## ##############
#################################### ################################### ###
######################################
############ ## ####
##############
##

PostgreSQL##pg_locks############

##################################max_locks_per_transaction#max_connections#######
##############
################### ###############1###10########

###########LIMIT#####################################SQL####################
##################### ##########

```
SELECT pg_advisory_lock(id) FROM foo WHERE id = 12345; -- ####
SELECT pg_advisory_lock(id) FROM foo WHERE id > 12345 LIMIT 100; -- ###
SELECT pg_advisory_lock(q.id) FROM
(
  SELECT id FROM foo WHERE id > 12345 LIMIT 100
) q; -- ####
```

####################LIMIT #################2########## ####################
####################### ################################ ####################
########pg_locks###########

###################9.28.10#######

13.4.

###
###

##MVCC##########################
######/################################ 1#######################################
##################
##################################

########################

13.2.3#####################################/################################## ######## ##1#######################

13.4.1.

###PostgreSQL##### ##############

###
default_transaction_isolation#serializable################### ######### ##

#########13.2.3##########

> ##: ##############################
>
> ##(26.4)####################### ## ## ##

13.4.2.

#################### ################################SELECT FOR UPDATE# #SELECT FOR SHARE#####LOCK TABLE############ #SELECT FOR UPDATE### #SELECT FOR SHARE#########################LOCK TABLE############### ### PostgreSQL##################################

###SELECT FOR UPDATE############## PostgreSQL##SELECT FOR UPDATE##UPDATE## #DELETE################### ##UPDATE##

#########MVCC################################## ###############1########## ## 2##### #SELECT sum(...)############2## 1####################2## ## SHARE###

###SELECT#INSERT#UPDATE#DELETE###MERGE############# ### ######################################

13.5.

###########
##################
##################SQLSTATE####40001(serialization_failure)######

SQLSTATE###40P01(deadlock_detected)#########################

#######SQLSTATE###23505(unique_violation)#############SQLSTATE##
#23P01(exclusion_violation)###################### ########################
##
###########
##
serialization_failure###
################################

##SQL### ######PostgreSQL#
################ ###########################

#######
#################################### ##
#########################

13.6.

DDL###########TRUNCATE############ALTER TABLE######MVCC###########
####DDL###
################## ########DDL###
—#######################ACCESS SHARE##############################DDL####
############ ##
##

###(26.4#####)# ##########
################################## ##
##
##

################################
##
##

13.7.

PostgreSQL####################/##########################PostgreSQL###########
################################## #####################

B-tree#######GiST###SP-GiST######

- ####/##############/#################### ##################################
  ##

##########

- ####/###########/########################## ########################## #######
  ##################################1#######################################
  ####

GIN######

####/##############/#################### ################################## ####GIN####################1################################## #####GIN## ######################

######B-tree################################## ################################## ## #############B-tree################### #########GiST#SP-GiST####GIN##############

#14#

##################### ## ####
PostgreSQL###########################

14.1. EXPLAIN###

PostgreSQL######################### ##
################### ################################### EXPLAIN##############
################################### ############################ ##############
######

#####[93.750000]#######9.3##########VACUUM ANALYZE###########################
####### ### ANALYZE########
###

##############EXPLAIN#######text########### ######EXPLAIN####################
#######################XML#JSON#YAML##################

14.1.1. EXPLAIN###

#################### ###################################### ################
###VALUES##FROM#
##################################
##################################
################ EXPLAIN###############1##################################
############# ### ###1######
########################### ######################

################################

```
EXPLAIN SELECT * FROM tenk1;

                        QUERY PLAN
-------------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..445.00 rows=10000 width=244)
```

#########WHERE################################ ##############################
######## ###########################

- ########## ###########################SORT##################
- ######## ## ###################
  ########################LIMIT#######
- ##
- ############################

#################19.7.2####################### ################################
####seq_page_cost#####1.0#########################
###

#####[82.994455]## ##############
################## ##
############################

############ ### ############
################################ ##
############### ##
#########################

rows## #####
##########WHERE## ####################
#####################################

#######

```
EXPLAIN SELECT * FROM tenk1;

                         QUERY PLAN
-------------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..445.00 rows=10000 width=244)
```

###########################

```
SELECT relpages, reltuples FROM pg_class WHERE relname = 'tenk1';
```

#####[91.773779]#tenk1##358########10000############## ################## * seq_page_cost#+######## * cpu_tuple_cost######### ########seq_page_cost# 1.0#cpu_tuple_cost#0.01### ##########(358 * 1.0) + (10000 * 0.01) = 458######

###WHERE####################

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 7000;

                         QUERY PLAN
-------------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..470.00 rows=7000 width=244)
   Filter: (unique1 < 7000)
```

EXPLAIN#####Seq Scan##########################WHERE################### ####### ### WHERE#################### #### #########10000################################## #####WHERE##########CPU### ########################10000 * cpu_operator_cost################

#####[90.792839]###################7000### ####rows############### #####2######## ################### ##########ANALYZE##################### #####ANALYZE###### ################################

##################

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 100;

                                  QUERY PLAN
-------------------------------------------------------------------
-----------
 Bitmap Heap Scan on tenk1  (cost=5.06..224.98 rows=100 width=244)
   Recheck Cond: (unique1 < 100)
   ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04 rows=100
 width=0)
         Index Cond: (unique1 < 100)
```

##########2#################### ############################(index condition)#### ########## ##################################### ##############################

#2################
##########
##bitmap################

###WHERE################

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 100 AND stringu1 = 'xxx';

                                 QUERY PLAN
-----------------------------------------------------------------
-----------
 Bitmap Heap Scan on tenk1  (cost=5.04..225.20 rows=1 width=244)
   Recheck Cond: (unique1 < 100)
   Filter: (stringu1 = 'xxx'::name)
   ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04 rows=100
 width=0)
         Index Cond: (unique1 < 100)
```

#####[90.105263]######stringu1 = 'xxx'#######################################
########## #########unique1#############stringu1############################
##################################

##################################

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 = 42;

                                 QUERY PLAN
-----------------------------------------------------------------
----------
 Index Scan using tenk1_unique1 on tenk1  (cost=0.29..8.30 rows=1
 width=244)
   Index Cond: (unique1 = 42)
```

############################## ##
################# ################################ ###ORDER BY############
##############################ORDER BY################### ######ORDER BY
unique1###

#####[90.043290]######ORDER BY################ ##########ORDER BY############
######## #########sort###########

```
EXPLAIN SELECT * FROM tenk1 ORDER BY unique1;

                            QUERY PLAN
-----------------------------------------------------------------
 Sort  (cost=1109.39..1134.39 rows=10000 width=244)
   Sort Key: unique1
   ->  Seq Scan on tenk1  (cost=0.00..445.00 rows=10000 width=244)
```

#####[91.812865]##(incremental
sort)##################

```
EXPLAIN SELECT * FROM tenk1 ORDER BY hundred, ten LIMIT 100;

                                                QUERY PLAN
-----------------------------------------------------------------
----------------------------
 Limit  (cost=19.35..39.49 rows=100 width=244)
   ->  Incremental Sort  (cost=19.35..2033.39 rows=10000 width=244)
         Sort Key: hundred, ten
```

```
         Presorted Key: hundred
         ->  Index Scan using tenk1_hundred on tenk1  (cost=0.29..1574.20
 rows=10000 width=244)
```

########LIMIT########### ########## ##

WHERE##AND#OR#################### #####

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 100 AND unique2 > 9000;

                                     QUERY PLAN
-------------------------------------------------------------------
------------------
 Bitmap Heap Scan on tenk1  (cost=25.07..60.11 rows=10 width=244)
   Recheck Cond: ((unique1 < 100) AND (unique2 > 9000))
   ->  BitmapAnd  (cost=25.07..25.07 rows=10 width=0)
         ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04 rows=100
 width=0)
               Index Cond: (unique1 < 100)
         ->  Bitmap Index Scan on tenk1_unique2  (cost=0.00..19.78
 rows=999 width=0)
               Index Cond: (unique2 > 9000)
```

############################## ############1#################################### ### ################################

###LIMIT############

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 100 AND unique2 > 9000 LIMIT
 2;

                                     QUERY PLAN
-------------------------------------------------------------------
------------------
 Limit  (cost=0.29..14.28 rows=2 width=244)
   ->  Index Scan using tenk1_unique2 on tenk1  (cost=0.29..70.27 rows=10
 width=244)
         Index Cond: (unique2 > 9000)
         Filter: (unique1 < 100)
```

################################LIMIT####### ####################### ####### ## ###Limit##########1/5######### ################ ########1/5############################## ###########Limit### ############## #######Limit####Limit##25## ###########

####################2################

```
EXPLAIN SELECT *
FROM tenk1 t1, tenk2 t2
WHERE t1.unique1 < 10 AND t1.unique2 = t2.unique2;

                                      QUERY PLAN
-------------------------------------------------------------------
-------------------
 Nested Loop  (cost=4.65..118.50 rows=10 width=488)
   ->  Bitmap Heap Scan on tenk1 t1  (cost=4.36..39.38 rows=10 width=244)
         Recheck Cond: (unique1 < 10)
```

```
        ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..4.36 rows=10
 width=0)
              Index Cond: (unique1 < 10)
   ->  Index Scan using tenk2_unique2 on tenk2 t2  (cost=0.29..7.90 rows=1
 width=244)
        Index Cond: (unique2 = t1.unique2)
```

#####[93.426573]## ############ #################### #################################### ############## #unique1 < 10 WHERE#########SELECT ... WHERE unique1 < 10##### ####### ######t1.unique2 = t2.unique2################################ # ## ############################ #########t1.unique2#################SELECT ... WHERE t2.unique2 = *constant*######################## ##########t2############################## ############################## ## ############10 * 7.91####################CPU#############

#############2## 2##########WHERE##### ########################## #######

```
EXPLAIN SELECT *
FROM tenk1 t1, tenk2 t2
WHERE t1.unique1 < 10 AND t2.unique2 < 10 AND t1.hundred < t2.hundred;

                                          QUERY PLAN
-----------------------------------------------------------------
--------------------------
 Nested Loop  (cost=4.65..49.36 rows=33 width=488)
   Join Filter: (t1.hundred < t2.hundred)
   ->  Bitmap Heap Scan on tenk1 t1  (cost=4.36..39.38 rows=10 width=244)
         Recheck Cond: (unique1 < 10)
         ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..4.36 rows=10
 width=0)
               Index Cond: (unique1 < 10)
   ->  Materialize  (cost=0.29..8.51 rows=10 width=244)
         ->  Index Scan using tenk2_unique2 on tenk2 t2  (cost=0.29..8.46
 rows=10 width=244)
               Index Cond: (unique2 < 10)
```

##t1.hundred < t2.hundred#tenk2_unique2################# ############## ## ######################################

################################10##############t2############################

##############ON###### ##################NULL################## ################################## #######

###

```
EXPLAIN SELECT *
FROM tenk1 t1, tenk2 t2
WHERE t1.unique1 < 100 AND t1.unique2 = t2.unique2;

                                          QUERY PLAN
-------------------------------------------------------------------
-----------------------
 Hash Join  (cost=226.23..709.73 rows=100 width=488)
   Hash Cond: (t2.unique2 = t1.unique2)
```

```
   ->  Seq Scan on tenk2 t2  (cost=0.00..445.00 rows=10000 width=244)
   ->  Hash  (cost=224.98..224.98 rows=100 width=244)
         ->  Bitmap Heap Scan on tenk1 t1  (cost=5.06..224.98 rows=100
 width=244)
               Recheck Cond: (unique1 < 100)
               ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04
 rows=100 width=0)
                     Index Cond: (unique1 < 100)
```

```
######################### ##########################################################
##################### ############################## tenk1####################
######## ####################################
```

```
############################
```

```
EXPLAIN SELECT *
FROM tenk1 t1, onek t2
WHERE t1.unique1 < 100 AND t1.unique2 = t2.unique2;

                                        QUERY PLAN
-------------------------------------------------------------------
-----------------------
 Merge Join  (cost=0.56..233.49 rows=10 width=488)
   Merge Cond: (t1.unique2 = t2.unique2)
   ->  Index Scan using tenk1_unique2 on tenk1 t1  (cost=0.29..643.28
 rows=100 width=244)
         Filter: (unique1 < 100)
   ->  Index Scan using onek_unique2 on onek t2  (cost=0.28..166.28
 rows=1000 width=244)
```

```
#####[59.016393]################################ ########################tenk1###
###################### ##############################onek#####################
#### ##########################################################################
############## ######################################### ########################
#############################################################-####-and-#########
############################ ############################################
```

```
#####[77.078086]#19.7.1#####enable/disable#########################################
################## ######################### 14.3########### #########onek########
############################################## ###################1######19.7.1##
########/################################################################ 14.3####
####### ################################################
```

```
SET enable_mergejoin = off;

EXPLAIN SELECT *
FROM tenk1 t1, onek t2
WHERE t1.unique1 < 100 AND t1.unique2 = t2.unique2;

                                        QUERY PLAN
-------------------------------------------------------------------
-----------------------
 Hash Join  (cost=226.23..344.08 rows=10 width=488)
   Hash Cond: (t2.unique2 = t1.unique2)
   ->  Seq Scan on onek t2  (cost=0.00..114.00 rows=1000 width=244)
   ->  Hash  (cost=224.98..224.98 rows=100 width=244)
         ->  Bitmap Heap Scan on tenk1 t1  (cost=5.06..224.98 rows=100
 width=244)
               Recheck Cond: (unique1 < 100)
               ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04
 rows=100 width=0)
```

```
                      Index Cond: (unique1 < 100)
```

#####[57.096774]##onek#########12%######
############ ######################### ######EXPLAIN ANALYZE################
################################50%##########################
############## ##########EXPLAIN ANALYZE##############

########################
################################

```
EXPLAIN VERBOSE SELECT unique1
FROM tenk1 t
WHERE t.ten < ALL (SELECT o.ten FROM onek o WHERE o.four = t.four);

                              QUERY PLAN
----------------------------------------------------------------------
 Seq Scan on public.tenk1 t  (cost=0.00..586095.00 rows=5000 width=4)
   Output: t.unique1
   Filter: (ALL (t.ten < (SubPlan 1).col1))
   SubPlan 1
     ->  Seq Scan on public.onek o  (cost=0.00..116.50 rows=250 width=4)
           Output: o.ten
           Filter: (o.four = t.four)
```

#####################################   ##################################
##t.four############-#####################   ##############_##).colN######
#EXPLAIN########## ###########N###############

############ALL## ############
##########################:#

```
EXPLAIN SELECT *
FROM tenk1 t
WHERE t.unique1 NOT IN (SELECT o.unique1 FROM onek o);

                                         QUERY PLAN
-------------------------------------------------------------------
-------------------------
 Seq Scan on tenk1 t  (cost=61.77..531.77 rows=5000 width=244)
   Filter: (NOT (ANY (unique1 = (hashed SubPlan 1).col1)))
   SubPlan 1
     ->  Index Only Scan using onek_unique1 on onek o  (cost=0.28..59.27
 rows=1000 width=4)
(4 rows)
```

#################1#####################################ANY################ #######-
############################ANY####################

initplan:##########

```
EXPLAIN VERBOSE SELECT unique1
FROM tenk1 t1 WHERE t1.ten = (SELECT (random() * 10)::integer);

                             QUERY PLAN
-------------------------------------------------------------------
 Seq Scan on public.tenk1 t1  (cost=0.02..470.02 rows=1000 width=4)
   Output: t1.unique1
   Filter: (t1.ten = (InitPlan 1).col1)
   InitPlan 1
     ->  Result  (cost=0.00..0.02 rows=1 width=4)
           Output: ((random() * '10'::double precision))::integer
```

######initplan###########1#################################### ##############
#)#1########t1.10############################ ##############################
##

14.1.2. EXPLAIN ANALYZE

EXPLAIN#ANALYZE#################################### ###########EXPLAIN###
##EXPLAIN#################### ########
##############

```
EXPLAIN ANALYZE SELECT *
FROM tenk1 t1, tenk2 t2
WHERE t1.unique1 < 10 AND t1.unique2 = t2.unique2;

                                                            QUERY PLAN
-----------------------------------------------------------------
-------------------------------------------------------------
 Nested Loop  (cost=4.65..118.50 rows=10 width=488) (actual
 time=0.017..0.051 rows=10 loops=1)
   ->  Bitmap Heap Scan on tenk1 t1  (cost=4.36..39.38 rows=10 width=244)
 (actual time=0.009..0.017 rows=10 loops=1)
         Recheck Cond: (unique1 < 10)
         Heap Blocks: exact=10
         ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..4.36 rows=10
 width=0) (actual time=0.004..0.004 rows=10 loops=1)
               Index Cond: (unique1 < 10)
   ->  Index Scan using tenk2_unique2 on tenk2 t2  (cost=0.29..7.90 rows=1
 width=244) (actual time=0.003..0.003 rows=1 loops=10)
         Index Cond: (unique2 = t1.unique2)
 Planning Time: 0.485 ms
 Execution Time: 0.073 ms
```

#actual time#####################cost########################## ##############
####### ##################################### ##############################

#####[92.911011]## ######################
######################## ########loops################################1######
loops####################################
####tenk2#####################0.220##########

########EXPLAIN ANALYZE############################## ####################
#############

```
EXPLAIN ANALYZE SELECT *
FROM tenk1 t1, tenk2 t2
WHERE t1.unique1 < 100 AND t1.unique2 = t2.unique2 ORDER BY t1.fivethous;

                                                                QUERY
 PLAN
-----------------------------------------------------------------
-------------------------------------------------------------------------
 Sort  (cost=713.05..713.30 rows=100 width=488) (actual time=2.995..3.002
 rows=100 loops=1)
   Sort Key: t1.fivethous
   Sort Method: quicksort  Memory: 74kB
   ->  Hash Join  (cost=226.23..709.73 rows=100 width=488) (actual
 time=0.515..2.920 rows=100 loops=1)
         Hash Cond: (t2.unique2 = t1.unique2)
         ->  Seq Scan on tenk2 t2  (cost=0.00..445.00 rows=10000
 width=244) (actual time=0.026..1.790 rows=10000 loops=1)
```

```
        ->  Hash  (cost=224.98..224.98 rows=100 width=244) (actual
 time=0.476..0.477 rows=100 loops=1)
              Buckets: 1024  Batches: 1  Memory Usage: 35kB
              ->  Bitmap Heap Scan on tenk1 t1  (cost=5.06..224.98
 rows=100 width=244) (actual time=0.030..0.450 rows=100 loops=1)
                    Recheck Cond: (unique1 < 100)
                    Heap Blocks: exact=90
                    ->  Bitmap Index Scan on tenk1_unique1
 (cost=0.00..5.04 rows=100 width=0) (actual time=0.013..0.013 rows=100
 loops=1)
                          Index Cond: (unique1 < 100)
 Planning Time: 0.187 ms
 Execution Time: 3.036 ms
```

###############
######################################
#####

################################

```
EXPLAIN ANALYZE SELECT * FROM tenk1 WHERE ten < 7;

                                                QUERY PLAN
-------------------------------------------------------------------
--------------------------------------
 Seq Scan on tenk1  (cost=0.00..470.00 rows=7000 width=244) (actual
 time=0.030..1.995 rows=7000 loops=1)
   Filter: (ten < 7)
   Rows Removed by Filter: 3000
 Planning Time: 0.102 ms
 Execution Time: 2.145 ms
```

################################ #Rows Removed################################
##############################

############################## ##########################

```
EXPLAIN ANALYZE SELECT * FROM polygon_tbl WHERE f1 @> polygon '(0.5,2.0)';

                                               QUERY PLAN
-------------------------------------------------------------------
-----------------------------------
 Seq Scan on polygon_tbl  (cost=0.00..1.09 rows=1 width=85) (actual
 time=0.023..0.023 rows=0 loops=1)
   Filter: (f1 @> '((0.5,2))'::polygon)
   Rows Removed by Filter: 7
 Planning Time: 0.039 ms
 Execution Time: 0.033 ms
```

##############################
################ ####################################

```
SET enable_seqscan TO off;

EXPLAIN ANALYZE SELECT * FROM polygon_tbl WHERE f1 @> polygon '(0.5,2.0)';

                                                            QUERY PLAN
-------------------------------------------------------------------
-------------------------------------------------------
 Index Scan using gpolygonind on polygon_tbl  (cost=0.13..8.15 rows=1
 width=85) (actual time=0.074..0.074 rows=0 loops=1)
```

```
   Index Cond: (f1 @> '((0.5,2))'::polygon)
   Rows Removed by Index Recheck: 1
 Planning Time: 0.039 ms
 Execution Time: 0.098 ms
```

##########GiST##########
################ ##

EXPLAIN#######################ANALYZE######BUFFERS###########

```
EXPLAIN (ANALYZE, BUFFERS) SELECT * FROM tenk1 WHERE unique1 < 100 AND
 unique2 > 9000;

                                                           QUERY PLAN
-----------------------------------------------------------------
-----------------------------------------------------------
 Bitmap Heap Scan on tenk1  (cost=25.07..60.11 rows=10 width=244) (actual
 time=0.105..0.114 rows=10 loops=1)
   Recheck Cond: ((unique1 < 100) AND (unique2 > 9000))
   Heap Blocks: exact=10
   Buffers: shared hit=14 read=3
   ->  BitmapAnd  (cost=25.07..25.07 rows=10 width=0) (actual
 time=0.100..0.101 rows=0 loops=1)
         Buffers: shared hit=4 read=3
         ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04 rows=100
 width=0) (actual time=0.027..0.027 rows=100 loops=1)
               Index Cond: (unique1 < 100)
               Buffers: shared hit=2
         ->  Bitmap Index Scan on tenk1_unique2  (cost=0.00..19.78
 rows=999 width=0) (actual time=0.070..0.070 rows=999 loops=1)
               Index Cond: (unique2 > 9000)
               Buffers: shared hit=2 read=3
 Planning:
   Buffers: shared hit=3
 Planning Time: 0.162 ms
 Execution Time: 0.143 ms
```

BUFFERS#############################I/O################

EXPLAIN ANALYZE##################EXPLAIN#################################
########################## ##
###########

```
BEGIN;

EXPLAIN ANALYZE UPDATE tenk1 SET hundred = hundred + 1 WHERE unique1 <
 100;

                                                           QUERY PLAN
-----------------------------------------------------------------
-----------------------------------------------------------
 Update on tenk1  (cost=5.06..225.23 rows=0 width=0) (actual
 time=1.634..1.635 rows=0 loops=1)
   ->  Bitmap Heap Scan on tenk1  (cost=5.06..225.23 rows=100 width=10)
 (actual time=0.065..0.141 rows=100 loops=1)
         Recheck Cond: (unique1 < 100)
         Heap Blocks: exact=90
         ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..5.04 rows=100
 width=0) (actual time=0.031..0.031 rows=100 loops=1)
               Index Cond: (unique1 < 100)
 Planning Time: 0.151 ms
 Execution Time: 1.856 ms
```

```
ROLLBACK;
```

################INSERT#UPDATE#DELETE#MERGE#########################
Insert#Update#Delete#Merge########### ##
##Update###############
##
##

#####[85.869565]#UPDATE#DELETE#MERGE################################   ####
##

```
EXPLAIN UPDATE gtest_parent SET f1 = CURRENT_DATE WHERE f2 = 101;

                                        QUERY PLAN
-----------------------------------------------------------------
---------------------
 Update on gtest_parent  (cost=0.00..3.06 rows=0 width=0)
   Update on gtest_child gtest_parent_1
   Update on gtest_child2 gtest_parent_2
   Update on gtest_child3 gtest_parent_3
   ->  Append  (cost=0.00..3.06 rows=3 width=14)
         ->  Seq Scan on gtest_child gtest_parent_1  (cost=0.00..1.01
 rows=1 width=14)
               Filter: (f2 = 101)
         ->  Seq Scan on gtest_child2 gtest_parent_2  (cost=0.00..1.01
 rows=1 width=14)
               Filter: (f2 = 101)
         ->  Seq Scan on gtest_child3 gtest_parent_3  (cost=0.00..1.01
 rows=1 width=14)
               Filter: (f2 = 101)
```

#####[78.512397]#######Update####################3################### #########
##1##4################ ########Update##
####################3###
############1####3##################
############################

```
EXPLAIN ANALYZE######Planning time#####################################
#############

EXPLAIN   ANALYZE######Execution   time################################
#################################### BEFORE#################Insert#Update#Delete
#############   ####AFTER#######################AFTER################   ####
####BEFORE#AFTER#########################  ################################
##EXPLAIN ANALYZE###################
```

#########-###
includeEXPLAIN ANALYZE#########################SERIALIZE#################
##########################

14.1.3.

#####[84.169279]#EXPLAIN   ANALYZE##
############## ####################################I/O################## #
####EXPLAIN   ANALYZE###########################gettimeofday()#########
##################### pg_test_timing####################################### ##
####EXPLAIN   ANALYZE##2###### #
1####################################### SERIALIZE#######################
#2##EXPLAIN##################ANALYZE####gettimeofday()###############
########## pg_test_timing##################################

EXPLAIN################################ ################################
############1###
############
##############1### ###
#polygon_tbl##################

################################### ###########LIMIT#########################
####### ###########LIMIT#######

```
EXPLAIN ANALYZE SELECT * FROM tenk1 WHERE unique1 < 100 AND unique2 > 9000
 LIMIT 2;

                                                              QUERY PLAN
-----------------------------------------------------------------
-------------------------------------------------------------
 Limit  (cost=0.29..14.33 rows=2 width=244) (actual time=0.051..0.071
 rows=2 loops=1)
   ->  Index Scan using tenk1_unique2 on tenk1  (cost=0.29..70.50 rows=10
 width=244) (actual time=0.051..0.070 rows=2 loops=1)
         Index Cond: (unique2 > 9000)
         Filter: (unique1 < 100)
         Rows Removed by Filter: 287
 Planning Time: 0.077 ms
 Execution Time: 0.086 ms
```

#######Limit#########################
###################################
##

################################ ##
################# ## ############
#####LIMIT########### ##
############### EXPLAIN ANALYZE################################### #######
###

##########BitmapAnd###BitmapOr####################

##EXPLAIN########################## ##
################################ ##########################Append####
MergeAppend###################### ##################EXPLAIN#######
##Subplans Removed: *N*################

14.2.

14.2.1.

#########
##########################

#####1### ####
#pg_class#reltuples#relpages######## ###########################

```
SELECT relname, relkind, reltuples, relpages
FROM pg_class
WHERE relname LIKE 'tenk1%';

       relname        | relkind | reltuples | relpages
----------------------+---------+-----------+----------
 tenk1                | r       |     10000 |      345
```

```
 tenk1_hundred        | i       |     10000 |       11
 tenk1_thous_tenthous | i       |     10000 |       30
 tenk1_unique1        | i       |     10000 |       30
 tenk1_unique2        | i       |     10000 |       30
(5 rows)
```

####tenk1##########10000###
##

########reltuples#relpages##################################   ###
#VACUUM#ANALYZE#CREATE   INDEX#####DDL##############   #############
#VACUUM#ANALYZE################################reltuples################
##################################pg_class#########################

######################WHERE######################### ###########WHERE#######
#WHERE############################## #############pg_statistic#############
pg_statistic######ANALYZE#VACUUM ANALYZE############################
#########

###############pg_statistic############pg_stats################
pg_stats#################### ####pg_stats################pg_statistic###
################## ## pg_stats###
################

```
SELECT attname, inherited, n_distinct,
       array_to_string(most_common_vals, E'\n') as most_common_vals
FROM pg_stats
WHERE tablename = 'road';

 attname | inherited | n_distinct |          most_common_vals
---------+-----------+------------+-----------------------------------
 name    | f         | -0.5681108 | I- 580                        Ramp+
         |           |            | I- 880                        Ramp+
         |           |            | Sp Railroad                       +
         |           |            | I- 580                            +
         |           |            | I- 680                        Ramp+
         |           |            | I- 80                         Ramp+
         |           |            | 14th                          St  +
         |           |            | I- 880                            +
         |           |            | Mac Arthur                    Blvd+
         |           |            | Mission                       Blvd+
...
 name    | t         |    -0.5125 | I- 580                        Ramp+
         |           |            | I- 880                        Ramp+
         |           |            | I- 580                            +
         |           |            | I- 680                        Ramp+
         |           |            | I- 80                         Ramp+
         |           |            | Sp Railroad                       +
         |           |            | I- 880                            +
         |           |            | State Hwy 13                  Ramp+
         |           |            | I- 80                             +
         |           |            | State Hwy 24                  Ramp+
...
 thepath | f         |          0 |
 thepath | t         |          0 |
(4 rows)
```

#####[72.272727]########2#################### 1##road############(inherited=t)
#########1##road######(inherited=f)######### ###############2############# 1#
#road##inherited=t)###################1##road########(inherited=f)#######
##########################10###########

ANALYZE###pg_statistic#######################most_common_vals#
#histogram_bounds############ALTER TABLE SET STATISTICS#######
#####default_statistics_target################################## ############100#####
##pg_statistic###########
###################### ##############################

##########################68##########

14.2.2.

######################
############################## ##
########### #######PostgreSQL##

##############################
###

##########CREATE STATISTICS###### ##
########## ##########ANALYZE ##########################ANALYZE####### ######
###pg_statistic_ext_data################

##################################ANALYZE############### ####################
##
##########

################################

14.2.2.1.

a########b###############
###b##a################ ##########a###########b###################### ########
############################## ############################## ################
########### ##

######################################
######################### ##
############

######################ANALYZE######################### ####################
#################### ##########dependencies##############################
########### ANALYZE###
##dependencies##################

################

```
CREATE STATISTICS stts (dependencies) ON city, zip FROM zipcodes;

ANALYZE zipcodes;

SELECT stxname, stxkeys, stxddependencies
  FROM pg_statistic_ext join pg_statistic_ext_data on (oid = stxoid)
  WHERE stxname = 'stts';
 stxname | stxkeys |             stxddependencies
---------+---------+------------------------------------------
 stts    | 1 5     | {"1 => 5": 1.000000, "5 => 1": 0.423130}
(1 row)
```

######1#zip code######5#city#############1.0### ###city#zip code#42%################
#zip code###########city(58%)################

##

14.2.2.1.1.

##########################IN####################### ########################
######LIKE#########################

####################
######0#################### ################

```
SELECT * FROM zipcodes WHERE city = 'San Francisco' AND zip = '94105';
```

##########################city####### ############0##########################
###

```
SELECT * FROM zipcodes WHERE city = 'San Francisco' AND zip = '90210';
```

##################################

######################### ###########################city#ZIP code#####GUI######
######### ######################################

14.2.2.2. ###N#####

############################ #####GROUP BY a, b############################
##

####################ANALYZE#N################## ############################
################# ndistinct## ###### #
##

##########ZIP code######N################

```
CREATE STATISTICS stts2 (ndistinct) ON city, state, zip FROM zipcodes;

ANALYZE zipcodes;

SELECT stxkeys AS k, stxdndistinct AS nd
  FROM pg_statistic_ext join pg_statistic_ext_data on (oid = stxoid)
  WHERE stxname = 'stts2';
-[ RECORD 1 ]-------------------------------------------------------
k  | 1 2 5
nd | {"1, 2": 33178, "1, 5": 33178, "2, 5": 27435, "1, 2, 5": 33178}
(1 row)
```

######33178############################## ZIP code#state#ZIP code#city#city#state######
############ZIP code########################## ###city#state########27435########
#####

##ndistinct#########
############# ######ANALYZE##############

14.2.2.3. ###MCV###

#########################(most-common value list)### ##############################
##################################

######################MCV####ANALYZE############ ######N#################
######################### MCV############N########################## ###
#mcv##

###########ZIP#########MCV################################MCV##############
#####

```
CREATE STATISTICS stts3 (mcv) ON city, state FROM zipcodes;

ANALYZE zipcodes;

SELECT m.* FROM pg_statistic_ext join pg_statistic_ext_data on (oid =
 stxoid),
                pg_mcv_list_items(stxdmcv) m WHERE stxname = 'stts3';

 index |         values         | nulls | frequency | base_frequency
-------+------------------------+-------+-----------+----------------
     0 | {Washington, DC}       | {f,f} |  0.003467 |        2.7e-05
     1 | {Apo, AE}              | {f,f} |  0.003067 |        1.9e-05
     2 | {Houston, TX}          | {f,f} |  0.002167 |       0.000133
     3 | {El Paso, TX}          | {f,f} |     0.002 |       0.000113
     4 | {New York, NY}         | {f,f} |  0.001967 |       0.000114
     5 | {Atlanta, GA}          | {f,f} |  0.001633 |        3.3e-05
     6 | {Sacramento, CA}       | {f,f} |  0.001433 |        7.8e-05
     7 | {Miami, FL}            | {f,f} |    0.0014 |          6e-05
     8 | {Dallas, TX}           | {f,f} |  0.001367 |        8.8e-05
     9 | {Chicago, IL}          | {f,f} |  0.001333 |        5.1e-05
   ...
(99 rows)
```

#######################DC#Washington##################0.35%#### ############
##########################0.0027%##2#############

###MCV################
######## ######ANALYZE###################

14.3. ####JOIN###########

####JOIN######################### ##################################

###########

```
SELECT * FROM a, b, c WHERE a.id = b.id AND b.ref = c.id;
```

#################################### ####WHERE###a.id = b.id######A#B######
#WHERE##############C##################### #####B#C#########A############
#####A#C#########B############### ################## #######################
#WHERE######A#C############ #PostgreSQL################2##################
####1###################### ##
######## ######################################

##########2#3############################ ################################### 10
6#7#######################
####### #############PostgreSQL##
########geqo_threshold################
###

###################

```
SELECT * FROM a LEFT JOIN (b JOIN c ON (b.ref = c.id)) ON (a.id = b.id);
```

##################################B#C############A#######################
##B#C##########A###############

```
#################################### ################################ ######
#######


SELECT * FROM a LEFT JOIN b ON (a.bid = b.id) LEFT JOIN c ON (a.cid =
 c.id);
```

#####A###B######C########## ######FULL JOIN############### LEFT JOIN#RIGHT JOIN################################

##########INNER JOIN#CROSS JOIN#####JOIN#######FROM################## ###############

#######JOIN###############PostgreSQL##############JOIN#################### ##### #######3#################


```
SELECT * FROM a, b, c WHERE a.id = b.id AND b.ref = c.id;
SELECT * FROM a CROSS JOIN b CROSS JOIN c WHERE a.id = b.id AND b.ref =
 c.id;
SELECT * FROM a JOIN (b JOIN c ON (b.ref = c.id)) ON (a.id = b.id);
```

#########JOIN##############2###3################################## ########3# ##

############JOIN###############join_collapse_limit#########1######### ########### ###########

################################ ########FROM########JOIN################ #### ######


```
SELECT * FROM a CROSS JOIN b, c, d, e WHERE ...;
```

join_collapse_limit = 1##########################A#B#################### ##### ############5#####1######

######################JOIN############################## ################### #######

##############


```
SELECT *
FROM x, y,
    (SELECT * FROM a, b, c WHERE something) AS ss
WHERE somethingelse;
```

########################## ######SELECT################################### ################################


```
SELECT * FROM x, y, a, b, c WHERE something AND somethingelse;
```

################################# ########WHERE###X#A#############A####### ####### ############################## ################### #####2##3###### ###5############ ############################## ########################### #from_collapse_limit##FROM#################################### ##

#################from_collapse_limit#join_collapse_limit############### ############ ## #
##join_collapse_limit#from_collapse_limit############################

####join_collapse_limit#1################################## ############## ##

14.4.

################################

14.4.1.

####INSERT#######################1############## ####SQL######BEGIN#### ##COMMIT################ ################################ ################ ####################### ####################PostgreSQL######################## ## 1#########################1######1## #### ################################

14.4.2. COPY###

#######################INSERT########COPY######### COPY################### ####### #######INSERT## COPY####### ##################1#########################

COPY##############INSERT##PREPARE###############EXECUTE############## #### #########INSERT########################## ########################## ##### #########################

COPY##############################INSERT############### ##################### ############PREPARE####################

COPY#########CREATE TABLE###TRUNCATE########################## ######### ####################################WAL############# ####wal_level#minimal## ######################## #################WAL#################

14.4.3.

##################################COPY#######################################
##

####################

14.4.4.

##
##

#####################################(#######################)################## ########### ## ############################### ## ########### ##

14.4.5. maintenance_work_mem####

##############maintenance_work_mem############################### ####CREATE INDEX#####ALTER TABLE ADD FOREIGN KEY########### COPY################### ################################

14.4.6. max_wal_size####

##############max_wal_size############################ #######PostgreSQL########## ##############checkpoint_timeout################################### ##### ## ############# ###max_wal_size################################

14.4.7. WAL##############################

#######WAL###WAL######## ################################## #####WAL########### ##wal_level#minimal##archive_mode#off##max_wal_senders########################## #################### ## ##

######WAL##############WAL## wal_level#minimal#############WAL################### #######fsync####### ##WAL####################################

14.4.8. ###ANALYZE###

#######################ANALYZE############## ############################## ## ANALYZE####VACUUM ANALYZE## #### ## ######### #########ANALYZE########### ###24.1.3###24.1.6##########

14.4.9. pg_dump###########

pg_dump## pg_dump############## #################### ##################################### psql############## ########pg_dump###########pg_restore#########################

########pg_dump#COPY####### ## ########### ############################## ################

- maintenance_work_mem###max_wal_size#######################
- WAL## ############ ####archive_mode#off##wal_level#minimal##max_wal_senders########### ##################################
- pg_dump#pg_restore## -j############## ##############################
- ## #####psql###pg_restore#-1###--single-transaction##################### ## ########## ## ##### #########WAL###############COPY###############
- ############CPU###########pg_restore#--jobs################## ############## #####################
- ####ANALYZE#########

##########COPY################################ ################### [1] # ## ##########

[1] --disable-triggers######################### ################################### ################## ###############

#max_wal_size########### ####maintenance_work_mem################ ##### ################################## #########ANALYZE############### ###24.1.3## #24.1.6#########

14.5.

####################PostgreSQL######################### ## ###################################### ################################### #########

- ####################################RAM############ ####################I/O# ##
- fsync######## ########################
- synchronous_commit######## ###########WAL################## ################### ####################################
- full_page_writes######## #########################
- max_wal_size###checkpoint_timeout######## ########################/pg_wal####### ########
- WAL###

#15#

PostgreSQL#####################CPU######################## ################### #### ############################## ### ######################## ### ###### ####################2###########4################### ################### ######################### ### ##

15.1.

#####################################*Gather*###*Gather Merge*################### ## #######

```
EXPLAIN SELECT * FROM pgbench_accounts WHERE filler LIKE '%x%';
                                     QUERY PLAN
-----------------------------------------------------------------
------------------
 Gather  (cost=1000.00..217018.43 rows=1 width=97)
   Workers Planned: 2
   ->  Parallel Seq Scan on pgbench_accounts  (cost=0.00..216018.33 rows=1
 width=97)
         Filter: (filler ~~ '%x%'::text)
(4 rows)
```

```
#######Gather###Gather    Merge###################  ######################
## Gather###Gather   Merge#################################### Gather##
#Gather Merge#################################### ########################
###Gather############################## #########Gather#####################
```

EXPLAIN############################## ########Gather######################### ## ############################## #max_parallel_workers_per_gather######## ######################## ##max_worker_processes#max_parallel_workers################ ######################## ## ########################## ############################## ############## ##max_worker_processes#max_parallel_workers########################## ## max_parallel_workers_per_gather## ###

```
#################################################### ######################
############### ################################## ############################
#################################### ######################################
############Gather######Gather Merge#################################### ##################################
```

```
###############Gather#####Gather  Merge################################## ############################## #####Gather################################## ################
```

15.2.

```
############################################## ############################## ############
```

- max_parallel_workers_per_gather#0################### max_parallel_workers_per_gather#######################################

############################## ##
#######

##

- ############################## #####################CTE#################### ######### ##SELECT############### #####
  - CREATE TABLE ... AS
  - SELECT INTO
  - CREATE MATERIALIZED VIEW
  - REFRESH MATERIALIZED VIEW
- ################ ## ### ##DECLARE CURSOR########################## ####FOR x IN query LOOP .. END LOOP#PL/pgSQL##################### ################################### ####################
- ####PARALLEL UNSAFE################## ##############PARALLEL SAFE### ## ###############PARALLEL UNSAFE######### 15.4##########
- ################################## ########################SQL############## ################## ## ############ ################################

############Gather## ##################Gather##########################

- ################max_worker_processes#####################################
- ##############################max_parallel_workers########################## ############
- #######0###############Execute############ ###################### ###libpq##### ####################libpq######################## ########################## ####################################max_parallel_workers_per_gather#0############ ####

15.3.

##############################

15.3.1.

#####################################

- ## ######################### ##########################
- ## ##################################### ################## ## ########## ##################################

- # # # # #

###########enable_parallel_append########

15.3.5.

################################parallel_setup_cost###parallel_tuple_cost############ ## ############## ########0## #### ###################15.2#15.4#############

```
###############EXPLAIN (ANALYZE, VERBOSE)
```
###############################
##
#####

15.4.

#########################parallel safe########parallel restricted#########parallel unsafe####### ################################ ################################## ######################### ###############Gather####Gather Merge########### ########Gather########################### ################################## ###### ##

###############

- ########CTE######
- ##########
- ########### #########IsForeignScanParallelSafeAPI##################
- InitPlan###########
- ####SubPlan##########

15.4.1.

```
#############################################################  ###############
################################  ############################  ##############
#################################################  ###########################
######################  CREATE FUNCTION####ALTER FUNCTION########  ##
#PARALLEL SAFE#PARALLEL RESTRICTED#PARALLEL UNSAFE#################
######## CREATE AGGREGATE####################SAFE#RESTRICTED#UNSAFE####
#PARALLEL###########
```

```
#####[76.051780]
```
###
```
########EXCEPTION########PL/pgSQL#########################PARALLEL UNSAFE#
###############  ################################################################
#########################PARALLEL RESTRICTED###############  ###
##setseed#random###################  ######################################
#################################################################PARALLEL
UNSAFE###############  ##################################################
################################PARALLEL RESTRICTED###############  ##
##setseed#random#####################
```

```
#############################################################################
##################  #############C################################  ########C##
################  ##########################################  ################
#UNSAFE###############
```

```
##########################################################################
############################################## ############################
##########################################PARALLEL RESTRICTED############

#################################################################### ##########
############WHERE#############################################################
## ##########################################WHERE#########Gather#############
################################## ####################
```

III.

#####[73.983740]######PostgreSQL######################### ################################ ######################### PostgreSQL## #### ###########PostgreSQL#################### ### ## PostgreSQL###

#####[67.867868]### ############################## ########## ######################### ########################### VI########## ############ ######################### ############################ ######################### ############# ########## VI###############

#####[86.610879]## ################ ################PostgreSQL####################################### ############# I#### II### ############ ## ######################### #########PostgreSQL################################### ############## I#### II#################

##

16. ############ 478
17. ############# 479
17.1. #### 479
17.2. ###### 480
17.3. Autoconf#Make########### 481
17.3.1. ### 481
17.3.2. ######## 481
17.3.3. `configure`##### 483
17.3.4. `configure`#### 490
17.4. Meson############ 492
17.4.1. ### 492
17.4.2. ######## 492
17.4.3. `meson setup`###### 494
17.4.4. `meson` Build Targets 501
17.5. ########### 502
17.5.1. ####### 502
17.5.2. #### 503
17.6. ############### 504
17.7. ############ 504
17.7.1. Cygwin 504
17.7.2. macOS 505
17.7.3. MinGW 506
17.7.4. Solaris 506
17.7.5. Visual Studio 507
18. ######### 510
18.1. PostgreSQL######## 510
18.2. ############ 510
18.2.1. ################ 511
18.2.2. ######## 511
18.3. ########### 512
18.3.1. ######## 514
18.3.2. ########### 514
18.4. ########### 515
18.4.1. ########## 515
18.4.2. systemd RemoveIPC 518
18.4.3. ####### 519
18.4.4. Linux############ 519
18.4.5. Linux#HugePages 520
18.5. ########### 521
18.6. PostgreSQL############## 522
18.6.1. pg_dumpall############### 523
18.6.2. pg_upgrade############## 524
18.6.3. ################## 524
18.7. ########### 524
18.8. ######## 525
18.9. SSL######TCP/IP## 526
18.9.1. ###### 526
18.9.2. OpenSSL### 526
18.9.3. ############ 527
18.9.4. #######SSL######### 527
18.9.5. ###### 527
18.10. GSSAPI######TCP/IP## 529
18.10.1. ###### 529
18.11. SSH###########TCP/IP## 529
18.12. Windows####Event Log### 530

19. ##### 531
19.1. ######## 531
19.1.1. ########## 531
19.1.2. ################ 531
19.1.3. SQL########### 532
19.1.4. ############# 533
19.1.5. ############ 533
19.2. ####### 534
19.3. ##### 535
19.3.1. #### 535
19.3.2. TCP Settings 537
19.3.3. ## 537
19.3.4. SSL 538
19.4. ##### 540
19.4.1. ### 540
19.4.2. #### 544
19.4.3. ######## 544
19.4.4. #######Vacuum## 544
19.4.5. ########### 545
19.4.6. ##### 546
19.5. ########WAL# 548
19.5.1. ### 548
19.5.2. ######## 552
19.5.3. ####### 553
19.5.4. #### 554
19.5.5. ############ 554
19.5.6. ######### 555
19.5.7. WAL Summarization 556
19.6. ######## 557
19.6.1. ###### 557
19.6.2. ######## 558
19.6.3. ######## 559
19.6.4. ######## 562
19.7. ####### 563
19.7.1. ########## 563
19.7.2. ######### 564
19.7.3. ################ 566
19.7.4. ############# 567
19.8. ########## 568
19.8.1. ###### 568
19.8.2. ########## 571
19.8.3. ##### 573
19.8.4. CSV########## 578
19.8.5. JSON####### 579
19.8.6. ####### 580
19.9. ####### 580
19.9.1. ######################## 580
19.9.2. ####### 581
19.10. ##Vacuum## 581
19.11. ############# 583
19.11.1. #### 583
19.11.2. ######### 589
19.11.3. ############# 590
19.11.4. ######### 592
19.12. ##### 592
19.13. ################## 593
19.13.1. ###PostgreSQL##### 593
19.13.2. ################### 594

19.14. ##### 595
19.15. ######### 595
19.16. ######## 597
19.17. ######### 597
19.18. ####### 601
20. ######## 603
20.1. pg_hba.conf#### 603
20.2. ####### 610
20.3. #### 611
20.4. Trust## 612
20.5. ####### 612
20.6. GSSAPI## 613
20.7. SSPI## 614
20.8. Ident## 615
20.9. Peer## 615
20.10. LDAP## 616
20.11. RADIUS## 618
20.12. ##### 619
20.13. PAM## 619
20.14. BSD## 620
20.15. ######## 620
21. ######### 622
21.1. ######## 622
21.2. ###### 623
21.3. ######## 624
21.4. ###### 626
21.5. ####### 626
21.6. ######## 628
22. ######## 629
22.1. ## 629
22.2. ######## 629
22.3. ########### 630
22.4. ######## 631
22.5. ######## 631
22.6. ###### 632
23. ########## 634
23.1. ######## 634
23.1.1. ## 634
23.1.2. ## 635
23.1.3. ####### 635
23.1.4. ######## 636
23.1.5. ICU#### 637
23.1.6. ### 638
23.2. ######## 638
23.2.1. ## 638
23.2.2. ####### 639
23.2.3. ICU######## 643
23.3. ######## 646
23.3.1. ########### 647
23.3.2. ######## 649
23.3.3. ################### 650
23.3.4. ############ 651
23.3.5. #### 655
24. ############## 656
24.1. ########## 656
24.1.1. ######### 656
24.1.2. ######## 656
24.1.3. ############ 657

24.1.4. ######## 658
24.1.5. ########ID######### 658
24.1.6. ########## 662
24.2. ############## 663
24.3. ######## 663
25. ########## 666
25.1. SQL###### 666
25.1.1. ####### 666
25.1.2. pg_dumpall### 667
25.1.3. ########### 667
25.2. ################## 668
25.3. ########################PITR# 669
25.3.1. WAL######## 670
25.3.2. ########### 672
25.3.3. Making an Incremental Backup 672
25.3.4. ####API################# 673
25.3.5. ######################### 674
25.3.6. ###### 676
25.3.7. ##### 676
25.3.8. ## 677
26. ##################### 679
26.1. ######## 679
26.2. ############### 681
26.2.1. ## 682
26.2.2. ########## 682
26.2.3. ##################### 682
26.2.4. ########## 683
26.2.5. ############### 683
26.2.6. ########### 684
26.2.7. ############# 685
26.2.8. ########## 686
26.2.9. ################### 688
26.3. ######## 688
26.4. ######## 689
26.4.1. ######## 689
26.4.2. ############## 690
26.4.3. ######## 692
26.4.4. ################### 694
26.4.5. ## 694
27. ############# 696
27.1. ####Unix### 696
27.2. ######## 697
27.2.1. ########### 697
27.2.2. ####### 697
27.2.3. `pg_stat_activity` 700
27.2.4. `pg_stat_replication` 714
27.2.5. `pg_stat_replication_slots` 716
27.2.6. `pg_stat_wal_receiver` 717
27.2.7. `pg_stat_recovery_prefetch` 718
27.2.8. `pg_stat_subscription` 718
27.2.9. `pg_stat_subscription_stats` 719
27.2.10. `pg_stat_ssl` 719
27.2.11. `pg_stat_gssapi` 720
27.2.12. `pg_stat_archiver` 720
27.2.13. `pg_stat_io` 721
27.2.14. `pg_stat_bgwriter` 723
27.2.15. `pg_stat_checkpointer` 723
27.2.16. `pg_stat_wal` 724

27.2.17. pg_stat_database 724
27.2.18. pg_stat_database_conflicts 726
27.2.19. pg_stat_all_tables 727
27.2.20. pg_stat_all_indexes 728
27.2.21. pg_statio_all_tables 729
27.2.22. pg_statio_all_indexes 730
27.2.23. pg_statio_all_sequences 730
27.2.24. pg_stat_user_functions 730
27.2.25. pg_stat_slru 731
27.2.26. ###### 732
27.3. ###### 734
27.4. ######## 735
27.4.1. ANALYZE######### 735
27.4.2. CLUSTER######### 736
27.4.3. COPY######### 737
27.4.4. CREATE INDEX######### 738
27.4.5. VACUUM######### 739
27.4.6. ################## 741
27.5. #### 742
27.5.1. ############ 742
27.5.2. ########## 742
27.5.3. ####### 748
27.5.4. ######## 748
27.6. Monitoring Disk Usage 749
27.6.1. Determining Disk Usage 749
27.6.2. Disk Full Failure 750
28. ########### 752
28.1. ### 752
28.2. ######## 753
28.2.1. ################ 754
28.3. #######(WAL) 754
28.4. ####### 754
28.5. WAL### 755
28.6. WAL### 758
29. ######### 759
29.1. ####### 759
29.2. ######## 760
29.2.1. ############## 760
29.2.2. #: ############ 761
29.2.3. #: ################ 764
29.3. Logical Replication Failover 765
29.4. ##### 766
29.4.1. ####### 767
29.4.2. #### 767
29.4.3. UPDATE## 767
29.4.4. ########### 767
29.4.5. ####### 767
29.4.6. ######### 768
29.4.7. # 768
29.5. #### 773
29.5.1. # 774
29.6. ###### 775
29.7. #### 776
29.8. ####### 777
29.8.1. ######### 777
29.9. ## 777
29.10. ###### 777
29.11. #### 778

29.11.1. ####### 778
29.11.2. ######## 778
29.12. ##### 779
30. ########(JIT) 780
30.1. JIT########## 780
30.1.1. JIT########### 780
30.1.2. #######(Inlining) 780
30.1.3. ### 780
30.2. ######JIT####### 780
30.3. ## 781
30.4. ### 782
30.4.1. ################# 782
30.4.2. #####JIT##### 782
31. ########## 783
31.1. ###### 783
31.1.1. ######################## 783
31.1.2. ######################## 783
31.1.3. ########## 784
31.1.4. ########## 785
31.1.5. ######### 786
31.1.6. ###### 786
31.2. ###### 786
31.2.1. ########### 787
31.2.2. ####### 787
31.2.3. ######## 787
31.2.4. ######### 787
31.2.5. ####### 787
31.2.6. ######## 788
31.2.7. #### ### 788
31.2.8. ####### 788
31.3. ########## 788
31.4. TAP### 789
31.4.1. #### 789
31.5. ############ 790
31.5.1. Autoconf#Make####### 790
31.5.2. Meson####### 790

#16#

PostgreSQL## ############################
###PostgreSQL################## ################17#####PostgreSQL###############
#######

##https://www.postgresql.org/download/#
PostgreSQL##

#17#

######PostgreSQL################################ RPM#Debian#################### ##########################16###########

17.1.

######Unix#############PostgreSQL################# ############################ ##17.6######

PostgreSQL###########################

- GNU make######3.81######## ##make###GNU make######### #GNU make#####`gmake`## ################ GNU make######################

  ```
  make --version
  ```

- #####[61.746988]#######Meson[1]#####PostgreSQL############ ###############Git### ################tarball################## Meson##############GNU make######## #################### ###########PostgreSQL#Meson[2]################ ############ ###### Meson############GNU#make########################

  Meson##############0.54###

- ISO/ANSI C########C99-######### GCC #################PostgreSQL############# ########################

- ############tar###gzip#bzip2###########

- ######Flex 2.5.35###Bison 2.3######## ##lex#yacc##############

- ################################Perl 5.14#############PL/Perl################# #############

- GNU Readline#################### #####psql#PostgreSQL####### SQL## ######### ############## ###########`configure#--without-readline`################ ## #########BSD######`libedit`################## ############NetBSD######### ## `libedit`######GNU#Readline#######`libreadline`###########`configure`##### ##`--with-libedit-preferred`################ #########Linux################# ##############`readline#readline-devel`################################## ##

- zlib#################### ###########`configure#--without-zlib`############## #### ##############pg_dump###pg_restore######################

- #######ICU############ ###########`configure#######--without-icu`########## ### ##############ICU##### 23.2 #################

  ICU############ICU4C######################## ICU4C#################4.2###

  ############################pkg-config######## ###ICU4C#####4.6############## ##############pkg-config###########`ICU_CFLAGS#ICU_LIBS#configure`####### # ######

[1] https://mesonbuild.com/
[2] https://mesonbuild.com/

```
./configure ... ICU_CFLAGS='-I/some/where/include' ICU_LIBS='-L/some/
where/lib -licui18n -licuuc -licudata'
```

#ICU4C#######################pkg-config############ICU_CFLAGS=' '######## ###############

################ ##

- ############PL/Perl########libperl##################Perl################ ##### #########Perl 5.14### PL/Perl#############################libperl############ ########### ############Perl###Perl ######################### PL/Perl################libperl#########configure### ######## #######PL/Perl################Perl######################### Perl####### ##############

  ###PL/Perl##################Perl##########usemultiplicity################## #########(perl -V################)#

- PL/Python##########################sysconfig########Python################ ###### ########Python 3.2###

  PL/Python################################libpython#################### ##### #####Python## ##### PL/Python################libpython#########configure######### ##### ##################################Python################################### ########## ############--enable-shared#######Python#configure#########

- PL/Tcl###############Tcl################ ##############Tcl 8.4###

- ########NLS#######################################Gettext API######### ##### ################Linux#NetBSD#Solaris################### #################https://www.gnu.org/software/gettext/############# GNU C######gettext#################### ##############GNU Gettext############## #################

- #####[88.963211]#########################OpenSSL###### OpenSSL##/dev/urandom############Windows#################### ##############1.0.1### ##### #########################OpenSSL###### OpenSSL##/dev/urandom############# #Windows#################### ############1.0.2###

- MIT Kerberos#GSSAPI###OpenLDAP########PAM##############################

- LZ4############################## default_toast_compression#wal_compression######## ##

- Zstandard############################## wal_compression########## ############## 1.4.0###

- PostgreSQL############################## J.2##########

GNU##################GNU##########################https://www.gnu.org/prep/ftp#### ### ####ftp://ftp.gnu.org/gnu/##########

17.2.

#############PostgreSQL########Web###https://www.postgresql.org/ftp/source/############ ####### #####postgresql-*version*.tar.gz###postgresql-*version*.tar.bz2#### ############## #################

```
tar xf postgresql-version.tar.bz2
```

##################postgresql-*version*##############PostgreSQL########### ## ##################################

#####Git#################### ###I.1##########

17.3. Autoconf#Make#############

17.3.1.

```
./configure
make
su
make install
adduser postgres
mkdir -p /usr/local/pgsql/data
chown postgres /usr/local/pgsql/data
su - postgres
/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
/usr/local/pgsql/bin/pg_ctl -D /usr/local/pgsql/data -l logfile start
/usr/local/pgsql/bin/createdb test
/usr/local/pgsql/bin/psql test
```

################

17.3.2.

1. ##

   ## configure######## ############ ################################

   ./configure

   ##

   ######################################configure############ #####*VPATH*## ####### ####################

   mkdir build_dir
   cd build_dir

   /path/to/source/tree/configure [#########]
   make

   #########################C## ###### #########/usr/local/pgsql############

   configure#############1############################## ################## ######################### configure##################17.3.3#########

   ###configure##17.3.4######################## ######################

2. ##

   ############################

   make

make all

#GNU make################## ###########################

#########(HTML#man)########(contrib)################################

make world

##########(contrib)##

make world-bin

##############Makefile#####################MAKELEVEL#######0###########
##

```
build-postgresql:
        $(MAKE) -C postgresql MAKELEVEL=0 all
```

##

3. **#########**

   ### ####################
   ###PostgreSQL################################### ###########

   make check

   #### root ######### ################## #31########################## ##########
   ###########################

4. **###########**

   ##

   #######################DB#######################18.6##########

   PostgreSQL#######################

   make install

   ############# 1###################### ############################### ######
   ####root########### ##

   #######HTML#man#######################

   make install-docs

   ###########world##############################

   make install-world

   #######################

   ##

make install-world-bin

make install#####make install-strip################################ ####strip###### ################## ## ############# ################################ install-strip########### ## ##################### ##################

##C########################### #####################

###############: ##

make -C src/bin install
make -C src/include install
make -C src/interfaces install
make -C doc install

src/bin##############################

########: ##############make uninstall######### #######################

######: ############make clean ## configure#######################make################ ##################### ##make distclean######### ## ################# ##

#########configure#######################configure######################## ########################make distclean############ ######################## ################

17.3.3. configure#####

configure##################### ##########################./configure --help######### ##Autoconf######## ########

17.3.3.1.

########make install################## #######--prefix########## ######### ## ########################### ## #man#doc######### ######## ##################--disable-rpath################

--prefix=*PREFIX*

/usr/local/pgsql#####*PREFIX*########################## ################ ###########*PREFIX*############################

--exec-prefix=*EXEC-PREFIX*

##############*PREFIX*#########*EXEC-PREFIX*################## ########### ################### #######*EXEC-PREFIX*#*PREFIX*######################## ########################## ##################

--bindir=*DIRECTORY*

######################## #######*EXEC-PREFIX*/bin######/usr/local/pgsql/bin######

--sysconfdir=*DIRECTORY*

######################## #######*PREFIX*/etc###

--libdir=*DIRECTORY*

################################## ######*EXEC-PREFIX*/lib###

--includedir=*DIRECTORY*

C###C++############################ ######*PREFIX*/include###

--datarootdir=*DIRECTORY*

#################################### ################################### ### ###*PREFIX*/share###

--datadir=*DIRECTORY*

################################## ######*DATAROOTDIR*### ################ ################################

--localedir=*DIRECTORY*

######*DATAROOTDIR*/locale#

--mandir=*DIRECTORY*

PostgreSQL##########################man*x*################### ##### #*DATAROOTDIR*/man###

--docdir=*DIRECTORY*

#man## ########################## ##############*DATAROOTDIR*/doc/postgresql###

--htmldir=*DIRECTORY*

PostgreSQL#HTML################################ ######*DATAROOTDIR*###

> ##
>
> #/usr/local/include#######################################PostgreSQL## ##################### ##################postgres###pgsql############# #####/postgresql###########datadir#sysconfdir#docdir####### ###### ####/usr/local##############/usr/local/doc/postgresql############ ####/opt/postgres####/opt/postgres/doc########### ############### ###C#######includedir####################### ###################### ##includedir######################### ################################## ######################## ########################libdir############### ########

17.3.3.2. PostgreSQL###

###########################PostgreSQL##################### ##########17.1####### ##################################

`--enable-nls[=LANGUAGES]`

########NLS#################################### *LANGUAGES*###############
#############################--enable-nls='de fr'####### ##############
#################### ####################################

################gettext API#########

`--with-perl`

PL/Perl##############

`--with-python`

PL/Python##############

`--with-tcl`

PL/Tcl##############

`--with-tclconfig=DIRECTORY`

Tcl##Tcl##############################tclConfig.sh############## #######
##########################Tcl################tclConfig.sh############
##########

`--with-llvm`

#####[59.285714]#LLVM#####JIT#######30#################### #####LLVM#####
################### LLVM################3.9### ######LLVM####JIT########
##############30##### ####LLVM######################### ###LLVM######10###

#####################llvm-config###### llvm-config################
###llvm-config-$major-$minor#PATH##### ####################llvm-
config###########LLVM_CONFIG######## ############

```
./configure ... --with-llvm LLVM_CONFIG='/path/to/llvm/bin/llvm-config'
```

LLVM#####clang################CLANG##############C++#############CXX#
###############

`--with-lz4`

LZ4#################

`--with-zstd`

Zstandard#################

`--with-ssl=LIBRARY`

SSL###################### ###########*LIBRARY*#openssl### #####OpenSSL##
###################### configure#########OpenSSL######################
#############

`--with-openssl`

--with-ssl=openssl###########

`--with-gssapi`

GSSAPI############# GSSAPI#######MIT Kerberos################### #######
##GSSAPI######MIT Kerberos##########################/usr/include#/usr/
lib############### #####--with-includes#--with-libraries############

########### `configure`#########GSSAPI################################## ##############

`--with-ldap`

###############LDAP############### ####32.18###20.10########## Unix ###OpenLDAP######################## Windows#######WinLDAP########## ## `configure`#########OpenLDAP## #######

`--with-pam`

PAM############################

`--with-bsd-auth`

BSD################## #BSD##############OpenBSD##########

`--with-systemd`

#####[73.205742]#systemd#################### #######systemd############### ##########################18.3######### ##################libsystemd####### ######################## ######systemd############## #######systemd######### ######################## ###18.3######### ##############libsystemd########## #################

`--with-bonjour`

Bonjour######################## #################Bonjour################ macOS######

`--with-uuid=`*`LIBRARY`*

#####UUID#########(UUID###########)uuid-ossp############ *`LIBRARY`*######### #########

- `bsd`#FreeBSD#####BSD########UUID#######
- `e2fs`#`e2fsprogs`##########UUID########## ############Linux#####macOS## ##########################
- `ossp`#OSSP UUID#####[3]######

`--with-ossp-uuid`

`--with-uuid=ossp`###########

`--with-libxml`

libxml2#########SQL/XML########### #########libxml2#####2.6.23########

pkg-config#################libxml2## ###PostgreSQL#`pkg-config`######## #######libxml2#############`xml2-config`############# ################################`pkg-config`########### ###

###############libxml2##################`pkg-config`#################### #############`XML2_CONFIG`#############`xml2-config`################## #`XML2_CFLAGS`#`XML2_LIBS`###### #`pkg-config`############libxml2########## ###############`XML2_CONFIG`#####`XML2_CFLAGS`#`XML2_LIBS`############### #########

[3] http://www.ossp.org/pkg/lib/uuid/

`--with-libxslt`

XML#XSL#######xml2##########libxslt####### `--with-libxml`#############

`--with-selinux`

######SElinux ############## sepgsql #########

17.3.3.3.

##PostgreSQL## ########### ################################

`--without-icu`

#####[52.331606]#ICU##################ICU###############23.2##### ######ICU # ############ICU############23.2####

`--without-readline`

Readline#########libedit########### ########psql#######################

`--with-libedit-preferred`

GPL######Readline####BSD######libedit################ ##Readline########

`--without-zlib`

Zlib############## ####pg_dump#pg_restore#######################

`--disable-spinlocks`

PostgreSQL############CPU################################ ################ ########### ## ###### ################PostgreSQL#####################PostgreSQL############# #####

`--disable-atomics`

CPU################ ################################## ################### ################ ############################

17.3.3.4.

`--with-includes=`*`DIRECTORIES`*

`DIRECTORIES`## #GNU Readline#### ##`--with-libraries`###### ############

`--with-includes=/opt/gnu/include:/usr/sup/include`

`--with-libraries=`*`DIRECTORIES`*

`DIRECTORIES`#################################### ######################### ####################`--with-includes`##################

`--with-libraries=/opt/gnu/lib:/usr/sup/lib`

`--with-system-tzdata=`*`DIRECTORY`*

PostgreSQL################################## ################## FreeBSD#Linux#Solaris#######################IANA################## ########

################# ##############*DIRECTORY*#####################PostgreSQL ######################### *DIRECTORY*################# /usr/share/zoneinfo######################## ### ####### ##########################PostgreSQL################################ ##############

############PostgreSQL################### ################################ ###########PostgreSQL########################

`--with-extra-version=`*`STRING`*

PostgreSQL#######*STRING*###### #################Git################## ##`git describe`## ###########

`--disable-rpath`

PostgreSQL#############################`--libdir`########################## ########## ## ############## ### ################### ##################################17.5.1##########

17.3.3.5.

###########`--with-pgport`################################ ################# ########

`--with-pgport=`*`NUMBER`*

#######################*NUMBER*####### ######5432### ######################### ### ########### ######################### ###########PostgreSQL###########

`--with-krb-srvnam=`*`NAME`*

GSSAPI######Kerberos###################### #######`postgres`### ######## Windows########################## Windows##################`POSTGRES`##### ########

`--with-segsize=`*`SEGSIZE`*

###################### ### ## ## ###############1################ ################ ## ################ ## ########### ## tar###################### ###################### ##################2############### ################## ##########################`pg_upgrade`################################

`--with-blocksize=`*`BLOCKSIZE`*

#################### ################I/O###### 8############################ ############ ##1##32###########2############### ############################ ################`pg_upgrade`##############################

`--with-wal-blocksize=`*`BLOCKSIZE`*

########*WAL*############# ###WAL###########I/O###### 8####################### ################### ##1##64##########2############### ######################## ########################`pg_upgrade`####WAL##########################

17.3.3.6.

###############PostgreSQL######################### --enable-debug########## ##########--enable-debug## DTrace #################--enable-dtrace##################

##--enable-debug#--enable-cassert############

--enable-debug

################################## ################################### ## ##################################GCC################################### ########## ## #######GCC ###### ## ################################## ##

--enable-cassert

##

--enable-tap-tests

#####[57.627119]#Perl TAP################# ####Perl########Perl#####IPC::Run# ##### ###31.4######### ######Perl TAP################## ####Perl####### #IPC::Run########## ###31.4#########

--enable-depend

############### ## ##makefile ######## ## # ######GCC #################

--enable-coverage

#####[66.343042]#GCC## #### ## ###31.5######### ########GCC## ######################### ######GCC## ############# ## ###31.5######### # #######GCC#######################

--enable-profiling

GCC## ######################## #gmon.out##################### ########GCC############################

--enable-dtrace

######DTrace###############PostgreSQL######### ###27.5#########

dtrace############DTRACE########### dtrace####PATH############/usr/sbin################################

###dtrace##################DTRACEFLAGS########### Solaris#64######## DTrace#########DTRACEFLAGS="-64"######### ###GCC#######################

```
./configure CC='gcc -m64' --enable-dtrace DTRACEFLAGS='-64' ...
```

Sun#######################

```
./configure CC='/opt/SUNWspro/bin/cc -xtarget=native64' --enable-dtrace
 DTRACEFLAGS='-64' ...
```

--enable-injection-points

######PostgreSQL################################## ####################### ############################## ################################ ######### ############### #######36.10.13######### ############################

--with-segsize-blocks=SEGSIZE_BLOCKS

########################### --with-segsize############################ ####### ################################

17.3.4. configure####

######################configure############### #######configure########## ########

./configure CC=/opt/bin/gcc CFLAGS='-O2 -pipe'

################################## ##########################

```
export CC=/opt/bin/gcc
export CFLAGS='-O2 -pipe'
./configure
```

##

#########################CC#CFLAGS### configure#######C############### ##CC ######################## #######configure##########gcc################ #########cc######## ########################CFLAGS###############

##########################

BISON

Bison######

CC

C######

CFLAGS

C##############

CLANG

--with-llvm#####################################clang#########

CPP

C#######

CPPFLAGS

C###############

CXX

C++######

CXXFLAGS

C++##############

DTRACE

dtrace########

DTRACEFLAGS

dtrace##############

FLEX

Flex######

LDFLAGS

################################

LDFLAGS_EX

########################

LDFLAGS_SL

#########################

LLVM_CONFIG

LLVM#######################llvm-config######

MSGFMT

########NLS###msgfmt######

PERL

Perl############ ####PL/Perl######################## ######perl###

PYTHON

Python############ ####PL/Python######################## ###########python3 python##########

TCLSH

Tcl############ ####PL/Tcl#####################Tcl############## ############# #########tclsh tcl tclsh8.6 tclsh86 tclsh8.5 tclsh85 tclsh8.4 tclsh84

XML2_CONFIG

libxml2#######################xml2-config#######

configure## ######configure## #CFLAGS#gcc#-Werror#########################configure################### ###############make######COPT############ configure######CFLAGS##### #LDFLAGS#########COPT########## ##############

make COPT='-Werror'

###

```
export COPT='-Werror'
make
```

> ##
>
> GCC##########-O1##################################(-O0) ###########(##
> ##############)########## ####################################1#1########
> ################### ##-O0######
> ######## ################make PROFILE=-O0 file.o#####make###########
> ####
>
> COPT#PROFILE#######PostgreSQL#makefile################ #################
> #########################PROFILE###############COPT######

17.4. Meson###############

17.4.1.

```
meson setup build --prefix=/usr/local/pgsql
cd build
ninja
su
ninja install
adduser postgres
mkdir -p /usr/local/pgsql/data
chown postgres /usr/local/pgsql/data
su - postgres
/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
/usr/local/pgsql/bin/pg_ctl -D /usr/local/pgsql/data -l logfile start
/usr/local/pgsql/bin/createdb test
/usr/local/pgsql/bin/psql test
```

```
################
```

17.4.2.

1. ##

   ### ##################
   ##meson setup##########

   meson setup build

   ##########builddir#srcdir####### srcdir###########Meson##########
   #meson.build#######srcdir####### builddir######

   meson setup## ##################Meson
   ########## ######################### ###

   ```
   # configure with a different installation prefix
   meson setup build --prefix=/home/user/pg-install

   # configure to generate a debug build
   meson setup build --buildtype=debug
   ```

```
# configure to build with OpenSSL support
meson setup build -Dssl=openssl
```

##################meson configure##########

```
meson configure -Dcassert=true
```

meson configure#########################17.4.3######

2. ##

   #######Meson#Ninja[4]########### Meson#####PostgreSQL#######################
   #ninja#############

   ```
   ninja
   ```

   Ninja########CPU#################### ########-j#####################
   ##

   ###########ninja################### ################################
   ###Meson####################### ############################ #########1###
   ## ###############Ninja##########
   ###########

   ninja##################--backend#########configure####################
   #meson compile###### ##########ninja################Meson#######[5]#######
   ##

3. #########

   ### ####################
   ###PostgreSQL################################### ##########

   meson test

   #### root ######## ################## #31########################## #########
   ##########################

   ####postgres#########pg_regress#pg_isolation_regress###########**--setup**
   running#meson test###########

4. ##########

   ##

   ######################DB######################18.6#########

   PostgreSQL#########ninja install######################

   ```
   ninja install
   ```

   ############# 1###################### ##############################
   ## ##########root########### ##

[4] https://ninja-build.org/
[5] https://mesonbuild.com/Running-Meson.html#building-from-the-source

##C#######################
#######################

ninja install##########################--quiet##################
#meson install########## meson install[6]############Meson###############

#######: ##############ninja uninstall##########

######: #######ninja clean##

17.4.3. meson setup######

meson setup#################### ##########################meson configure --help######### ##
#Meson######[7]######### #######meson setup########

17.4.3.1.

#######ninja install###meson install################## ####
####--prefix######17.4.1######## ##
################ ##
########################## #man#doc################## ##################-Drpath=false#################

--prefix=*PREFIX*

/usr/local/pgsql#Unix###########*current drive letter*:/usr/local/pgsql#Windows###########*PREFIX*######################## ##############
###############*PREFIX*############################

--bindir=*DIRECTORY*

######################## ######*PREFIX*/bin###

--sysconfdir=*DIRECTORY*

###################### #######*PREFIX*/etc###

--libdir=*DIRECTORY*

################################## ######*PREFIX*/lib###

--includedir=*DIRECTORY*

C###C++########################### ######*PREFIX*/include###

--datadir=*DIRECTORY*

################################## ######*PREFIX*/share### ###############
################################

--localedir=*DIRECTORY*

######*DATADIR*/locale###

--mandir=*DIRECTORY*

PostgreSQL##########################man*x*################## #####
#*DATADIR*/man###

[6] https://mesonbuild.com/Commands.html#install
[7] https://mesonbuild.com/Commands.html#configure

##

#/usr/local/include#####################################PostgreSQL## ##################### ##################postgres###pgsql############# #####/postgresql###########datadir#sysconfdir#docdir####### ###### ####/usr/local##############/usr/local/doc/postgresql############ ####/opt/postgres####/opt/postgres/doc########### ############### ###C#######includedir####################### ###################### ##includedir###libdir################ ########

17.4.3.2. PostgreSQL###

##########################PostgreSQL##################### #########17.1##### ## ############enabled######## #disabled####################

PostgreSQL#########################-D###########

-Dnls={ auto | enabled | disabled }

########NLS############# ####Gettext API########################## ######## ##Gettext API#######################

-Dplperl={ auto | enabled | disabled }

PL/Perl############### ###########

-Dplpython={ auto | enabled | disabled }

PL/Python############### ###########

-Dpltcl={ auto | enabled | disabled }

PL/Tcl############### ###########

-Dtcl_version=*TCL_VERSION*

PL/Tcl#########Tcl############

-Dicu={ auto | enabled | disabled }

#####[70.833333]#ICU##########################ICU################# #23.2#### ####### ##########ICU4C######################## ICU4C################4.2### ######ICU###########ICU############23.2#### ##########ICU4C################ ######## ICU4C############4.2###

-Dllvm={ auto | enabled | disabled }

#####[64.119601]#LLVM#####JIT#######30##################### #####LLVM##### ################### LLVM################3.9### ############ ######LLVM#### JIT#######################30##### ####LLVM######################### ######### ####LLVM######10### ################

#######################llvm-config###### llvm-config################### # llvm-config-$version#PATH###### #######################llvm-config### #########LLVM_CONFIG######## #############

`-Dlz4={ auto | enabled | disabled }`

LZ4################# ###########

`-Dzstd={ auto | enabled | disabled }`

Zstandard################# ###########

`-Dssl={ auto | LIBRARY }`

SSL####################### ###########*LIBRARY*#`openssl`### #####OpenSSL## ####################### ################################OpenSSL########### ####################### ##################

`-Dgssapi={ auto | enabled | disabled }`

GSSAPI############# GSSAPI#######MIT Kerberos################### ######## ##GSSAPI#######MIT Kerberos##########################`/usr/include`#`/usr/lib`############### #######PostgreSQL#`pkg-config`######################### ##### ########### `meson configure`#########################GSSAPI######### ###################

`-Dldap={ auto | enabled | disabled }`

###############LDAP############### ####32.18###20.10########### Unix# ##OpenLDAP######################### Windows#######WinLDAP############# # ########## `meson configure`#########################OpenLDAP############# ##############

`-Dpam={ auto | enabled | disabled }`

PAM############################## ###########

`-Dbsd_auth={ auto | enabled | disabled }`

BSD################### #BSD###############OpenBSD########### ###########

`-Dsystemd={ auto | enabled | disabled }`

#####[73.853211]#systemd####################### ########systemd############### ######################### ###18.3########## ########### ################ ##libsystemd################################ ######systemd############### ###### ##systemd################################### ###18.3########## ########### #### ###########libsystemd############################

`-Dbonjour={ auto | enabled | disabled }`

Bonjour######################### ######################Bonjour############### ## macOS#######

`-Duuid=`*`LIBRARY`*

#####UUID##########(UUID###########)uuid-ossp########### *`LIBRARY`*######### ##########

- `none`#uuid################### ###########
- `bsd`#FreeBSD#####BSD#########UUID########
- `e2fs`#`e2fsprogs`###########UUID########### #############Linux#####macOS## #########################

- `ossp`#OSSP UUID#####[8]#######

`-Dlibxml={ auto | enabled | disabled }`

libxml2#########SQL/XML########### ########### #########Libxml2#####2.6.23#### #####

################libxml2##################`pkg-config`######################## ##########

`-Dlibxslt={ auto | enabled | disabled }`

XML#XSL########xml2###########libxslt####### `-Dlibxml`############## ######## ###

`-Dselinux={ auto | enabled | disabled }`

######SElinux ############## sepgsql ########## ###### auto ###

17.4.3.3.

`-Dreadline={ auto | enabled | disabled }`

Readline#########libedit############ ##################psql#################### ## #########

`-Dlibedit_preferred={ true | false }`

GPL######Readline####BSD######libedit################ ######################## ########################Readline########

`-Dzlib={ auto | enabled | disabled }`

Zlib################ ##########pg_dump#pg_restore#pg_basebackup################ ####### #################

`-Dspinlocks={ true | false }`

###############true########## false###################PostgreSQL#CPU########### ################ ########################## ################################ ## ################PostgreSQL########### ###########PostgreSQL#################

`-Datomics={ true | false }`

################true########## false#######CPU################ ################# ################## ##################################### ################### ##########

17.4.3.4.

`--auto_features={ auto | enabled | disabled }`

##auto############## ################# ###optional######################## ###################

`--backend=`*`BACKEND`*

#####[70.270270]#Meson##################ninja#################### ####Visual Studio#############`BACKEND`#`vs`####### ##################Meson#ninja########### ############ ####Visual Studio#############*`BACKEND`*#`vs`########

[8] http://www.ossp.org/pkg/lib/uuid/

-Dc_args=*OPTIONS*

#########C##########################

-Dc_link_args=*OPTIONS*

#########C#######################

-Dextra_include_dirs=*DIRECTORIES*

DIRECTORIES## #GNU Readline#### ##-Dextra_lib_dirs###### ###########

#: -Dextra_include_dirs=/opt/gnu/include,/usr/sup/include#

-Dextra_lib_dirs=*DIRECTORIES*

DIRECTORIES################################### ######################### ####################-Dextra_include_dirs##################

#: -Dextra_lib_dirs=/opt/gnu/lib,/usr/sup/lib#

-Dsystem_tzdata=*DIRECTORY*

PostgreSQL################################## ################## FreeBSD#Linux#Solaris######################IANA################## ######## ################# ##############*DIRECTORY*####################PostgreSQL ######################## *DIRECTORY*################## /usr/share/zoneinfo######################## ## ####### ##########################PostgreSQL################################ #############

###########PostgreSQL################### ################################ ###########PostgreSQL#######################

-Dextra_version=*STRING*

PostgreSQL#######*STRING*###### #################Git################## ##git describe## ###########

-Drpath={ true | false }

###############true######### false#######PostgreSQL########################## ###--libdir################################### ########################## ## ##################### ##################### ## #17.5.1#########

-D*BINARY_NAME=PATH*

PostgreSQL###meson configure############### ##########################meson configure###### #########:

```
meson configure -DBISON=PATH_TO_BISON
```

17.4.3.5.

####################J.2#########

`-Ddocs={ auto | enabled | disabled }`

HTML###man#################### ###########

`-Ddocs_pdf={ auto | enabled | disabled }`

PDF################### ###########

`-Ddocs_html_style={ simple | website }`

##CSS################### ######`simple`### ###`website`#######postgresql.org[9]#### #####HTML##############

17.4.3.6.

`-Dpgport=`*`NUMBER`*

#######################*NUMBER*####### ######5432### ######################### ### ########### ######################### ###########PostgreSQL###########

`-Dkrb_srvnam=`*`NAME`*

GSSAPI######Kerberos###################### #######`postgres`### ######### Windows########################## Windows##################`POSTGRES`##### #######

`-Dsegsize=`*`SEGSIZE`*

####################### ### ## ## ###############1################ ################ ## ################ ## ########### ### tar######################## ######################## #################2##############

`-Dblocksize=`*`BLOCKSIZE`*

*#######*KB######### ##############I/O###### ######8KB######################## ################ ##1##32(KB)###2##############

`-Dwal_blocksize=`*`BLOCKSIZE`*

*WAL#######*KB######### ###WAL##########I/O###### ######8KB############### ######################### ##1##64(KB)###2##############

17.4.3.7.

###############PostgreSQL######################### `--enable-debug`########### ###########`--enable-debug`## DTrace #################`-Ddtrace`###################

###`--buildtype=debug`#`-Dcassert` options############

`--buildtype=`*`BUILDTYPE`*

################################## ######`debugoptimized`### ################ ############################`--debug`#`--optimization`############

[9] https://www.postgresql.org/docs/current/

#################plain#debug#debugoptimized#release### ##########Meson #######[10]#########

--debug

################################## ################################## ## ##################################GCC################################## ########## ## #######GCC######## ## ################################## ################################

--optimization=*LEVEL*

############ LEVEL#{0,g,1,2,3,s}############

--werror

################################ ##############

-Dcassert={ true | false }

##

-Dtap_tests={ auto | enabled | disabled }

#####[58.549223]#Perl TAP################# #############Perl########Perl#### #IPC::Run###### ###31.4######### ######Perl TAP################## ######## ##Perl########IPC::Run########## ###31.4#########

-DPG_TEST_EXTRA=*TEST_SUITES*

################################## ############################## ###31.1.3# ########

-Db_coverage={ true | false }

#####[66.343042]#GCC## #### ## ###31.5######### ########GCC## ######################## ######GCC## ############# ## ###31.5######### # #######GCC########################

-Ddtrace={ auto | enabled | disabled }

#####[52.976190]########DTrace##########PostgreSQL######### ########27.5##### ##### ################PostgreSQL#########DTrace######## ###27.5#########

dtrace############DTRACE########## dtrace####PATH############/usr/sbin###############################

-Dinjection_points={ true | false }

######PostgreSQL################################### ###################### ################################ ################################ ######### ############### #######36.10.13######### ###########################

-Dsegsize_blocks=SEGSIZE_BLOCKS

############################ -Dsegsize################################## ## ################################

[10] https://mesonbuild.com/Running-Meson.html#configuring-the-build-directory

17.4.4. meson Build Targets

##################ninja *target*############ ################################### ###########*target*#####/#################

17.4.4.1. Code Targets

`all`

Build everything other than documentation

`backend`

Build backend and related modules

`bin`

Build frontend binaries

`contrib`

Build contrib modules

`pl`

Build procedural languages

17.4.4.2. Developer Targets

`reformat-dat-files`

Rewrite catalog data files into standard format

`expand-dat-files`

Expand all data files to include defaults

`update-unicode`

Update unicode data to new version

17.4.4.3. Documentation Targets

`html`

Build documentation in multi-page HTML format

`man`

Build documentation in man page format

`docs`

Build documentation in multi-page HTML and man page format

`doc/src/sgml/postgres-A4.pdf`

Build documentation in PDF format, with A4 pages

`doc/src/sgml/postgres-US.pdf`

Build documentation in PDF format, with US letter pages

`doc/src/sgml/postgres.html`

Build documentation in single-page HTML format

`alldocs`

Build documentation in all supported formats

17.4.4.4. Installation Targets

`install`

Install postgres, excluding documentation

`install-docs`

Install documentation in multi-page HTML and man page formats

`install-html`

Install documentation in multi-page HTML format

`install-man`

Install documentation in man page format

`install-quiet`

Like "install", but installed files are not displayed

`install-world`

Install postgres, including multi-page HTML and man page documentation

`uninstall`

Remove installed files

17.4.4.5. Other Targets

`clean`

Remove all build products

`test`

Run all enabled tests (including contrib)

`world`

Build everything, including documentation

`help`

List important targets

17.5.

17.5.1.

##########
FreeBSD# Linux# NetBSD# OpenBSD####Solaris###

##################################### ###############LD_LIBRARY_PATH#######
################ Bourne ####sh#ksh#bash#zsh####

```
LD_LIBRARY_PATH=/usr/local/pgsql/lib
export LD_LIBRARY_PATH
```

csh###tcsh##############

```
setenv LD_LIBRARY_PATH /usr/local/pgsql/lib
```

/usr/local/pgsql/lib##### 1#--libdir################# /etc/profile#~/.bash_profile################################ ##################
##http://xahlee.info/UnixResource_dir/_/ldpath.html######

################LD_RUN_PATH####################

Cygwin##############PATH#######.dll#####bin############

##############################ld.so#rld########## ##########################
############

```
psql: error in loading shared libraries
libpq.so.2.1: cannot open shared object file: No such file or directory
```

################

##Linux########root#######

```
/sbin/ldconfig /usr/local/pgsql/lib
```

########ldconfig##
################ FreeBSD#NetBSD###OpenBSD#################

```
/sbin/ldconfig -m /usr/local/pgsql/lib
```

#########################

17.5.2.

##/usr/local/pgsql#################################/usr/local/pgsql/bin######## 1#--bindir########PATH############ ################### #######
###PostgreSQL##############

############~/.bash_profile#####################/etc/profile##########
##############

```
PATH=/usr/local/pgsql/bin:$PATH
export PATH
```

csh#tcsh###########################

```
set path = ( /usr/local/pgsql/bin $path )
```

################################man###
#######

```
MANPATH=/usr/local/pgsql/share/man:$MANPATH
export MANPATH
```

####PGHOST#PGPORT###
##PGHOST#############
######### ##

17.6.

#########CPU###
#######################################PostgreSQL############################
##########################PostgreSQL#######[11]#################### ############
#######################################PostgreSQL##############################
###########################

#####PostgreSQL####CPU#####################
x86#PowerPC#S/390#SPARC#ARM#MIPS#RISC-V#PA-RISC# ########################
##32####64################# `--disable-spinlocks`################CPU########
###################

#####[93.448940]#PostgreSQL############################
Linux#Windows#FreeBSD#OpenBSD#NetBSD#DragonFlyBSD#macOS#AIX#Solaris#illumos# #
#Unix################################## ################################CPU##
################# ######################17.7#################################
###

###
##<pgsql-bugs@lists.postgresql.org>######### #############PostgreSQL#####
########<pgsql-hackers@lists.postgresql.org>############

#####[93.788820]#PostgreSQL###POSTGRES########
##Alpha#Itanium#M32R#M68K#M88K#NS32K#SuperH#VAX###CPU######
##4.3BSD#BEOS#BSD/OS#DG/UX#Dynix#HP-
UX#Irix#NeXTSTEP#QNX#SCO#SINIX#Sprite#SunOS#Tru64 UNIX#ULTRIX##############
###########

17.7.

###PostgreSQL### ###########17.1#############
######################31###########

###

17.7.1. Cygwin

######PostgreSQL##Windows##Linux########Cygwin###########################
Windows###########Cygwin########################

##############Cygwin#########Unix#######################`./configure;make`; #
###

- Windows##############Cygwin#bin############### ########################
- `adduser`################ Windows########################### ################
  ###########

[11] https://buildfarm.postgresql.org/

- su################ Windows##su##########ssh######### ######################
  ####
- OpenSSL############
- ############cygserver###### #############/usr/sbin/cygserver&###### ##
  ######PostgreSQL##############initdb########################## #########
  ######PostgreSQL###############cygserver######SEMMNS##################
  ####
- ###########C############################ ###############export
  LANG=C.utf8#########C#####PostgreSQL####################
- ############make check####################listen()##################
  ############################ make ##MAX_CONNECTIONS####################
  #######

  ```
  make MAX_CONNECTIONS=5 check
  ```

  #################10#########

Windows NT######cygserver#PostgreSQL############## ##########Cygwin#
PostgreSQL#############README############### ###/usr/share/doc/Cygwin#####
############

17.7.2. macOS

PostgreSQL#macOS############Apple#######################################

```
xcode-select --install
```

######GUI########################## Xcode##########################

###macOS##sysroot################ ####
##configure###SDK#################configure################### ################
###sysroot########
############# #######PG_SYSROOT####################

```
make PG_SYSROOT=/desired/path all
```

################################

```
xcrun --show-sdk-path
```

######################sysroot##
#######ABI##############

configure#PG_SYSROOT########configure##########sysroot#############

```
./configure ... PG_SYSROOT=/desired/path
```

############macOS#################### ##################################

-isysroot#########################(###########################)#

```
./configure ... PG_SYSROOT=none
```

###Apple##############################PostgreSQL############################## ##

macOS##System Integrity Protection# (SIP)####`DYLD_LIBRARY_PATH`################## ###########`make check`###### `make check`###`make install`############ ## ##PostgreSQL###########SIP##########

17.7.3. MinGW

######PostgreSQL for Windows##Microsoft#############Unix############MinGW####### ###### MinGW################################

######`configure`######################Unix############MinGW##Unix######## MSYS##http://www.mingw.org/############ ########################## ########### #######

MinGW####64##################https://mingw-w64.org/##64################## ##`PATH`###bin################`--host=x86_64-w64-mingw32`#######`configure`## #####

MSYS##################################`CMD.EXE`##psql#############

17.7.3.1. Collecting Crash Dumps

##Windows##PostgreSQL###########Unix##################################minidumps# ####### ######Windows Debugger Tools#Visual Studio############Windows############# ##`crashdumps`#### ########################## ##################################

17.7.4. Solaris

PostgreSQL#Solaris################## ####################################

17.7.4.1.

GCC####Sun################## #############SPARC########Sun############### Sun ################`/usr/ucb/cc`######`/opt/SUNWspro/bin/cc`################

https://www.oracle.com/technetwork/server-storage/solarisstudio/downloads/##Sun Studio###### ###### ####GNU####Solaris 10########Solaris companion CD######## Solaris#### ##############################http://www.sunfreeware.com###### #############https://www.gnu.org/prep/ftp#########

17.7.4.2. ################configure######

##`configure`####################################libz#libreadline####libssl######### ################### ###############`configure`########`LDFLAGS`################ ###

```
configure ... LDFLAGS="-R /usr/sfw/lib:/opt/sfw/lib:/usr/local/lib"
```

########ld##################

17.7.4.3.

SPARC##################Sun Studio######## ######################`-xO5`######### ######## ##########`-fast`#####`errno`#########################

SPARC#64#######################32############## 64###########64########32###### ###### ####AMD64 CPU######32##########################CPU######32############ ###

17.7.4.4. PostgreSQL##########DTrace##

########DTrace###################27.5##########

```
##postgres#########################################


Undefined                       first referenced
 symbol                             in file
AbortTransaction                    utils/probes.o
CommitTransaction                   utils/probes.o
ld: fatal: Symbol referencing errors. No output written to postgres
collect2: ld returned 1 exit status
make: *** [postgres] Error 1
```

DTrace############################## DTrace########Solaris 10u4########

17.7.5. Visual Studio

#################PostgreSQL#######https://www.postgresql.org/download/################ ###########Windows########################## ##########PostgreSQL############## ##############

######Visual###PostgreSQL for Windows##17.4###########meson############# ##### Windows######32######64####Windows 10########

######psql################################ Cygwin######################### ##Windows###########psql###########################

######PostgreSQL##Microsoft#Visual C++##################### ###########Visual Studio#Visual Studio Express####Microsoft Windows SDK#################### Visual Studio# #######################Visual Studio 2022########Microsoft##############Windows SDK 10################

Both 32-bit and 64-bit builds are possible with the Microsoft Compiler suite. 32-bit PostgreSQL builds are possible with Visual Studio 2015 to Visual Studio 2022, as well as standalone Windows SDK releases 10 and above. 64-bit PostgreSQL builds are supported with Microsoft Windows SDK version 10 and above or Visual Studio 2015 and above.

##########################Microsoft Windows SDK################################## ###10##################### https://www.microsoft.com/download#############

######SDK#Windows Headers and Libraries#################### Visual C++ Compilers##### Windows SDK####################Visual Studio######### #####8.0a###Windows SDK#### ##############################

17.7.5.1.

#####[63.809524]#PostgreSQL########################### ######Windows##PostgreSQL# ####################

ActiveState Perl

> ########################ActiveState Perl###### MinGW###Cygwin#Perl######## # ##PATH############# #####https://www.activestate.com##############:#####5.14##### ### ###Standard Distribution#######

Bison and Flex

######Bison#Flex###### Bison######2.3########## Flex######2.5.35############

######Bison###Flex#####MinGW############http://www.mingw.org/wiki/MSYS###### ##msys##############

```
######flex.exe#bison.exe#########PATH################ MinGW#########
#MinGW############\msys\1.0\bin##########
```

```
##

######GnuWin32###Bison#############C:\Program Files\GnuWin32####
######################################## ####C:\GnuWin32#########
###PATH#######GnuWin32##NTFS###############C:
\PROGRA~1\GnuWin32##########
```

###

ActiveState Tcl

######PL/Tcl########## #########8.4###### ################

Diff

#######################diff###### http://gnuwin32.sourceforge.net#############

Gettext

######NLS##############gettext###### http://gnuwin32.sourceforge.net############# ## ##################################

MIT Kerberos

######GSSAPI################# MIT Kerberos#https://web.mit.edu/Kerberos/dist/index.html#############

libxml2 and libxslt

######XML############# #####https://zlatkovic.com/pub/libxml#######http://xmlsoft.org# ############ libxml2#iconv################## #################

LZ4

######LZ4################## #########https://github.com/lz4/lz4/releases########### ##

Zstandard

######Zstandard################## #########https://github.com/facebook/zstd/releases## ###########

OpenSSL

######SSL############# #####https://slproweb.com/products/Win32OpenSSL.html###### #https://www.openssl.org#############

ossp-uuid

######UUID-OSSP#####contrib######### ####http://www.ossp.org/pkg/lib/uuid/######

Python

######PL/Python############# #####https://www.python.org#############

zlib

######pg_dump###pg_restore#################### #####https://www.zlib.net#########
####

17.7.5.2. Special Considerations for 64-Bit Windows

######PostgreSQL#64###Windows##x64##################

###############32#####64####################### #######32#############64#########
####################PostgreSQL####### ####################################

######Python#OpenSSL##64############### 64
#########32############################## PostgreSQL######################32####
################## #######64#####PostgreSQL##############

17.7.5.3. Collecting Crash Dumps

##Windows##PostgreSQL###########Unix##################################minidumps#
####### ######Windows Debugger Tools#Visual Studio############Windows#############
##crashdumps## ##########################
####################################

#18#

##

##########################PostgreSQL##
############ PostgreSQL###
#################################### ##############################

18.1. PostgreSQL########

#######################PostgreSQL########################### #################
####################### ########################## #####nobody#############
##############################
######################PostgreSQL######################

##########PostgreSQL######################################

#####Unix#####################useradd#adduser######### postgres###########
################################

18.2.

################################## #########################SQL#############
########### ## #######
#############postgres################### ################################
######################## #############postgres##############################
############### #######template1#template0###2#################### ######
###########
###############22###########

##1################## ############
############## ######################## ############################/usr/
local/pgsql/data#/var/lib/pgsql/data### ##############PostgreSQL########
#######initdb#################

##########PostgreSQL## ###########
############################ ######initdb############################## #####
##########################

#####################-D##################################initdb####### #####
#######

```
$ initdb -D /usr/local/pgsql/data
```

##############PostgreSQL######################################

###
-D###########PGDATA################

#########pg_ctl#######initdb############

```
$ pg_ctl -D /usr/local/pgsql/data initdb
```

pg_ctl##pg_ctl#########(18.3##)# ##################

#####################initdb##################### ######################## #initdb###### PostgreSQL## ######### ############ ## ##################### ##root############ ##################

```
root# mkdir /usr/local/pgsql
root# chown postgres /usr/local/pgsql
root# su postgres
postgres$ initdb -D /usr/local/pgsql/data
```

##############################initdb## ##

##initdb#PostgreSQL################################ ########################### ###### ##

#####################################PostgreSQL############################### ################################### ##################################### # ##################################0700#########0600### ##################### ##################0750#########0640###

################initdb#-W#--pwprompt#--pwfile#########1####### ################################ ########trust##############-A scram-sha-256######### #####initdb##################pg_hba.conf############# ###########peer################################## #######20###########

initdb################################ ################################### ## ######################## #######23.1######### ############################## #initdb####### ##################################initdb######################## ######################### ###C#POSIX########################### ############## ##########

##initdb################################## ############################### ## #23.3#########

#C####POSIX## ################### ########## ## ##

18.2.1.

PostgreSQL## ##############pg_upgrade#####

18.2.2.

#####POSIX#########################PostgreSQL######## ######################### ############################## ## ##

18.2.2.1. NFS

PostgreSQL##################NFS############# PostgreSQL#NFS#################### ########NFS#################################### PostgreSQL############NFS##### #######################

NFS#PostgreSQL##########hard########################### hard############### ### soft################### #############PostgreSQL##I/O######## ###########

sync#################### async#################PostgreSQL################### ######fsync############# ########################## ####sync############### #####Linux###NFS########################## #####NFS#######fsync############ ##########################fsync#################################### ########## ### ##################

#################NFS####iSCSI############################ ################### ## ########NFS############DBA######### ##

18.3.

##############postgres########

###########PostgreSQL## ############ ## #### ###########################

#######################-D#######################postgres############ ###### ###

```
$ postgres -D /usr/local/pgsql/data
```

########################### ####PostgreSQL################################ -D## ##################PGDATA############################## ######################

###########postgres############## ##############Unix###########

```
$ postgres -D /usr/local/pgsql/data >logfile 2>&1 &
```

#############24.3##########

postgres## ###postgres############19##### ######

######################## pg_ctl ################################### ##########

```
pg_ctl start -l logfile
```

-D##########postgres############## pg_ctl####################

##########PostgreSQL#contrib/start-scripts############## ##############root

############################# ########/etc/rc.local#####/etc/rc.d/rc.local######### #######init.d#rc.d############## ###############PostgreSQL

###################### *root*######################## ######su postgres -c '...'######################## ##########

```
su postgres -c 'pg_ctl start -D /usr/local/pgsql/data -l serverlog'
```

############################## ##

- FreeBSD###PostgreSQL###########contrib/start-scripts/freebsd##########
  ###
- OpenBSD########/etc/rc.local##############

  ```
  if [ -x /usr/local/pgsql/bin/pg_ctl -a -x /usr/local/pgsql/bin/
  postgres ]; then
      su -l postgres -c '/usr/local/pgsql/bin/pg_ctl start -s -l /var/
  postgresql/log -D /usr/local/pgsql/data'
      echo -n ' postgresql'
  fi
  ```

- Linux#######

  ```
  /usr/local/pgsql/bin/pg_ctl start -l logfile -D /usr/local/pgsql/data
  ```

  #/etc/rc.d/rc.local#/etc/rc.local######### ####PostgreSQL##########
  #contrib/start-scripts/linux##############

  systemd###########################/etc/systemd/system/
  postgresql.service##########

  ```
  [Unit]
  Description=PostgreSQL database server
  Documentation=man:postgres(1)
  After=network-online.target
  Wants=network-online.target

  [Service]
  Type=notify
  User=postgres
  ExecStart=/usr/local/pgsql/bin/postgres -D /usr/local/pgsql/data
  ExecReload=/bin/kill -HUP $MAINPID
  KillMode=mixed
  KillSignal=SIGINT
  TimeoutSec=infinity

  [Install]
  WantedBy=multi-user.target
  ```

  Type=notify##############configure --with-systemd################

  ######################## ##############systemd##############90################
  ################### ####PostgreSQL#######################################
  ##################### ##########infinity########################

- NetBSD###FreeBSD#Linux######################
- Solaris###/etc/init.d/postgresql#################1##########

  ```
  su - postgres -c "/usr/local/pgsql/bin/pg_ctl start -l logfile -D /usr/
  local/pgsql/data"
  ```

```
  ####/etc/rc3.d###S99postgresql##############################
```

##############PID############postmaster.pid############# ################# ##

18.3.1.

############################## ## ########################## ################################

```
LOG:  could not bind IPv4 address "127.0.0.1": Address already in use
HINT:  Is another postmaster already running on port 5432? If not, wait a
 few seconds and retry.
FATAL:  could not create any TCP/IP sockets
```

######################## ##################################### ############ ####Address already in use############################## ################ ################################

```
$ postgres -p 666
LOG:  could not bind IPv4 address "127.0.0.1": Permission denied
HINT:  Is another postmaster already running on port 666? If not, wait a
 few seconds and retry.
FATAL:  could not create any TCP/IP sockets
```

##################

```
FATAL:  could not create shared memory segment: Invalid argument
DETAIL:  Failed system call was shmget(key=5440001, size=4011376640,
 03600).
```

###########################PostgreSQL##################4011376640############ ####### ###shared_memory_type#sysv###################### ################# ######shared_buffers## ################## ##

############:

```
FATAL:  could not create semaphores: No space left on device
DETAIL:  Failed system call was semget(5440126, 17, 03600).
```

################################### ########System V#########PostgreSQL######### ###################### ################max_connections########################## ##

System V IPC############18.4.1#########

18.3.2.

##

```
psql: error: connection to server at "server.joe.com" (123.123.123.123),
 port 5432 failed: Connection refused
        Is the server running on that host and accepting TCP/IP
 connections?
```

################################## TCP/IP################### ###########TCP/IP#################

#############Unix#######################

```
psql: error: connection to server on socket "/tmp/.s.PGSQL.5432" failed:
 No such file or directory
        Is the server running locally and accepting connections on that
 socket?
```

##/tmp######unix_socket_directories#####################

##Connection refused####No such file or directory###### ######Connection refused######################### ################### #########20.15######################## Connection timed out###

18.4.

PostgreSQL## ################ #####PostgreSQL## #############

18.4.1.

PostgreSQL#####################(IPC)########################### Unix######### ###System V# IPC###POSIX# IPC############## Windows############################ ####

#######PostgreSQL#########System V###############mmap############ ############ ##System V############## #shared_memory_type#### ####System V##POSIX############ ################## ###Linux#FreeBSD######POSIX#######################System V# ###########

System V IPC########################### PostgreSQL################################ ########################### #18.3.1 ########### ################################### ### # 18.1######### ###################### ##########################

#18.1 System V IPC#####

##	##	###PostgreSQL###########
SHMMAX	#####################	####1###########################
SHMMIN	#####################	1
SHMALL	#########################	#########SHMMAX############ #ceil(SHMMAX/PAGE_SIZE)# + ### ###########
SHMSEG	#####################	#####1############################ ##
SHMMNI	#####################	SHMSEG### + ##############
SEMMNI	Maximum number of semaphore identifiers (i.e., sets)	at least ceil((max_connections + autovacuum_max_workers + max_wal_senders + max_worker_processes + 7) / 16) plus room for other applications

##	##	###**PostgreSQL**##########
SEMMNS	Maximum number of semaphores system-wide	`ceil((max_connections + autovacuum_max_workers + max_wal_senders + max_worker_processes + 7) / 16) * 17` plus room for other applications
SEMMSL	#############	##17
SEMMAP	#############	#####
SEMVMX	########	##1000##########32767########### ###########

PostgreSQL##########System V##########64##############48########### ######### ################################ ######################System V############### ##############shared_memory_type###dynamic_shared_memory_type############System V#######SHMALL################## #########SHMALL########################### ##

#######################SHMMIN###PostgreSQL########32############1#### ###### ##########SHMMNI#####################SHMSEG################0############### ##############

#####[93.559322]#System V #############PostgreSQL#########max_connections########## ############autovacuum_max_workers######WAL#######max_wal_senders############## ##(max_worker_processes)###1##########16################## ##################### ##17############### ############## #SEMMNS######################## #max_connections#autovacuum_max_workers#max_wal_senders#max_worker_processes# ##################16#######1################### 18.1############# SEMMNI##### ################################ ##################ceil((max_connections + autovacuum_max_workers + max_wal_senders + max_worker_processes + 6) / 16)############ ####################################No space left on device## #########semget##########

#######SEMMAP######SEMMNS########################## ######################## ## ########## ### ######################## ############### ##

SEMMNU#SEMUME############semaphore undo########PostgreSQL##########

POSIX##############System V################ ##########max_connections########## ############autovacuum_max_workers######WAL#######max_wal_senders############## ##(max_worker_processes)###1############ ######################POSIX########## ############

FreeBSD

shared_memory_type#sysv################################ System V ######## ################

######IPC###sysctl###loader######################### ####sysctl########## ########

```
# sysctl kern.ipc.shmall=32768
# sysctl kern.ipc.shmmax=134217728
```

####################/etc/sysctl.conf#######

shared_memory_type#sysv##########System V######RAM#################### ############################ ###sysctl#####kern.ipc.shm_use_phys###### #########

FreeBSD jail#########sysvshm######new########System V#################### #### #FreeBSD 11.0####jail##IPC###

NetBSD

shared_memory_type#sysv################################ ## ##kern.ipc.semmni#kern.ipc.semmns###############NetBSD############## ######

########IPC######sysctl###############

```
# sysctl -w kern.ipc.semmni=100
```

####################/etc/sysctl.conf#######

shared_memory_type#sysv##########System V######RAM#################### ############################ ###sysctl#####kern.ipc.shm_use_phys###### #########

OpenBSD

shared_memory_type#sysv################################ ## ##kern.seminfo.semmni#kern.seminfo.semmns###############OpenBSD#### ################

########IPC######sysctl###############

```
# sysctl kern.seminfo.semmni=100
```

####################/etc/sysctl.conf#######

Linux

shared_memory_type#sysv## ############# System V #######################

############sysctl################## ###16####################

```
$ sysctl -w kernel.shmmax=17179869184
$ sysctl -w kernel.shmall=4194304
```

####################/etc/sysctl.conf#########

macOS

shared_memory_type#sysv####################################

macOS################################/etc/sysctl.conf################## ##

```
kern.sysv.shmmax=4194304
kern.sysv.shmmin=1
kern.sysv.shmmni=32
kern.sysv.shmseg=8
kern.sysv.shmall=1024
```

#########macOS##/etc/sysctl.conf###########5#########################
#############

SHMMAX#4096###########

############SHMALL#4############

SHMMNI#######sysctl######################## #########/etc/sysctl.conf###
############### ###################

Solaris
illumos

###PostgreSQL################################ Solaris#######SHMMAX######
RAM#1/4####### ################postgres########################### #####
#root###########

```
projadd -c "PostgreSQL DB User" -K "project.max-shm-
memory=(privileged,8GB,deny)" -U postgres -G postgres user.postgres
```

#######user.postgres##########postgres###############8GB####### ####
###################PostgreSQL############################## ###PostgreSQL
#postgres#######postgres######################## ############

##

```
project.max-shm-ids=(priv,32768,deny)
project.max-sem-ids=(priv,4096,deny)
project.max-msg-ids=(priv,4096,deny)
```

#########PostgreSQL##################################### projects#prctl##
###System Administrator's Guide##2# ###################

18.4.2. systemd RemoveIPC

systemd####################IPC## ##
#PostgreSQL######################## PostgreSQL########################
#postgres##################################

logind.conf#RemoveIPC####################IPC#################### #########
###########systemd######################################

###
##

```
WARNING:  could not remove shared memory segment "/PostgreSQL.1450751626":
 No such file or directory
```

IPC############## vs. #####System V vs. POSIX##systemd################IPC#########
############# ############################

#####################################postgres##############################
###########

############/etc/login.defs#SYS_UID_MAX######systemd##############

################useradd -r,adduser --system#############postgres########
##############

##

```
RemoveIPC=no
```

/etc/systemd/logind.conf###################

##
##################################PostgreSQL##################

18.4.3.

Unix#################PostgreSQL################################## ########### ## ####################2####### ##### ## ######root################## ## setrlimit######################## ###########ulimit#Bourne ####### #limit#csh############################## BSD########/etc/login.conf###### ######################### ########################## ######### #maxproc#openfiles#datasize### #########

```
default:\
...
        :datasize-cur=256M:\
        :maxproc-cur=256:\
        :openfiles-cur=256:\
...
```

#-cur######## #############-max#########

#####################################

- Linux########fs.file-max############################## ###sysctl-w fs.file-max=*N*####### ################/etc/sysctl.conf########### # ################################## ###/usr/src/linux/Documentation/proc.txt#########

PostgreSQL#########1###
################1###

###
######## ##1###

##################################
############################PostgreSQL#max_files_per_process################
#################

###########################1######################### ####################
#####PostgreSQL############################## ###########Resource temporarily unavailable###Connection refused######################## ####################
###128### #######sysctl#################PostgreSQL########## #######Linux#
#net.core.somaxconn####FreeBSD##kern.ipc.soacceptqueue#macOS###BSD##
#kern.ipc.somaxconn###############

18.4.4. Linux#############

Linux################PostgreSQL########### ############################
##PostgreSQL################################PostgreSQL postmaster ################
##############

############################## ##

```
Out of Memory: Killed process 12345 (postgres).
```

####postgres############################## ################################### #######PostgreSQL##############

#########1########PostgreSQL## ### ##########################OOM#killer## #############

PostgreSQL## ################## ####shared_buffers#work_mem###hash_mem_multiplier################### ## ## ######max_connections######## ############################

########OOM killer[1]################### ############################## ##########sysctl################################

```
sysctl -w vm.overcommit_memory=2
```

###/etc/sysctl.conf############## #######vm.overcommit_ratio########### #### ############https://www.kernel.org/doc/Documentation/vm/overcommit-accounting##### #####

vm.overcommit_memory##########################*OOM*#######postmaster###### #-1000######### #####OOM killer################# ################PostgreSQL#### ######postgres################

```
echo -1000 > /proc/self/oom_score_adj
```

#####root###################### ############## ####root########################## ##### ##############postgres#########################

```
export PG_OOM_ADJUST_FILE=/proc/self/oom_score_adj
export PG_OOM_ADJUST_VALUE=0
```

###############postmaster#######OOM killer############OOM################## #### ####OOM################PG_OOM_ADJUST_VALUE################ #PG_OOM_ADJUST_VALUE############################## PG_OOM_ADJUST_FILE## ############postmaster###OOM##############postmaster########################## #####

18.4.5. Linux#HugePages

PostgreSQL###########################shared_buffers#########huge pages########### ######## PostgreSQL#############CONFIG_HUGETLBFS=y## #CONFIG_HUGETLB_PAGE=y############ ##############huge pages################## ################ ###huge pages#########postgres######## #shared_memory_size_in_huge_pages######## ## ########## ##################

```
$ postgres -D $PGDATA -C shared_memory_size_in_huge_pages
3170
```

[1] https://lwn.net/Articles/104179/

```
$ grep ^Hugepagesize /proc/meminfo
Hugepagesize:       2048 kB
$ ls /sys/kernel/mm/hugepages
hugepages-1048576kB  hugepages-2048kB
```

##########2MB#######2MB###1GB#huge_page_size#### ##shared_memory_size_in_huge_pages#################### #########3170# huge page##################huge page#################### ###############

```
# sysctl -w vm.nr_hugepages=3170
```

#################/etc/sysctl.conf#################### ####huge page###### ##############

```
# echo 3170 > /sys/kernel/mm/hugepages/hugepages-2048kB/nr_hugepages
```

hugepagesz=2M hugepages=3170#####################################

#####################huge pages## ####### ####################huge pages############# #########huge pages########## ##############

```
$ cat /sys/kernel/mm/hugepages/hugepages-2048kB/nr_hugepages
```

sysctl####vm.hugetlb_shm_group#########ulimit -l######################## ####OS####huge pages#####################

PostgreSQL#huge pages##############################huge pages################### #### postgresql.conf#huge_pages#on########huge pages############### ########## #huge pages##########PostgreSQL##################

Linux#huge pages######https://www.kernel.org/doc/Documentation/vm/hugetlbpage.txt####### ##

18.5.

########################## ####################postgres################### ##

##########PostgreSQL## ###### ########################

#############postgres#####################################

SIGTERM

> ################### SIGTERM## ## ################################ ## ####################################

SIGINT

> ################# ############################SIGTERM################### ###################### ### ##################################

SIGQUIT

> ################# ##############SIGQUIT################### 5########### ##SIGKILL###### ## ############ #########WAL########################## ###################

pg_ctl## ####Windows#######
#kill####################### postgres#####PID##ps###################
#postmaster.pid################### ##############################

```
$ kill -INT `head -1 /usr/local/pgsql/data/postmaster.pid`
```

> ##
>
> ################SIGKILL############# ############################## ##
> ##SIGKILL####################postgres####### ########################
> ####################

#############################pg_terminate_backend()## 9.94#################
##########SIGTERM#########

18.6. PostgreSQL###############

####PostgreSQL##

###PostgreSQL## ##########10.1##10#####
#######1####################10################## PostgreSQL#10.0##########, #####
#############9.5.3### ######################################9.5########## #########
#############3##############9.5####################

#####
###10.1######10.0######10.6######### #######9.5.3#9.5.0#9.5.1#9.5.6######### #######
#####
################

PostgreSQL############################## ##################### ################
############pg_upgrade#########
############################## ############PostgreSQL##########################
###################### ################################

###
####################E########### ###################### ######################
##

################
################################## PostgreSQL##################################
#######

##

##

SQL

##########SQL############# ##############################

#####API

##################################libpq##########################

########

##################################

###C##API

#####C#####################API############# ################################ ########

18.6.1. pg_dumpall################

PostgreSQL################PostgreSQL## - ###### #####pg_dumpall############################ ########################### ######## ##PostgreSQL######### ########################

#########PostgreSQL#pg_dump#pg_dumpall############ ############################ #### #################9.2#############################

####################/usr/local/pgsql#############/usr/local/pgsql/data# ############# #####################

1. ## ############################ #################### #######/usr/local/pgsql/data/pg_hba.conf######## ## ##############20######### ##

   ##

   pg_dumpall > *outputfile*

   ###########################pg_dumpall####################25.1.2 ########## #### ###########PostgreSQL 17beta3#pg_dumpall############# ######################### #################### ## ## ############################ ########## #################

2. ############

   pg_ctl stop

   ####PostgreSQL## ###Red Hat Linux## ####################

   /etc/rc.d/init.d/postgresql stop

   ################18###########

3. ## ########## ## ################################## ######## ######################################

   mv /usr/local/pgsql /usr/local/pgsql.old

   ####################################

4. ####17#################PostgreSQL##############

5. ############################ ## ############################

```
/usr/local/pgsql/bin/initdb -D /usr/local/pgsql/data
```

6. ###pg_hba.conf#postgresql.conf##################

7. ##

```
/usr/local/pgsql/bin/postgres -D /usr/local/pgsql/data
```

8. ##########################

```
/usr/local/pgsql/bin/psql -d postgres -f outputfile
```

###psql#######

################################

```
pg_dumpall -p 5432 | psql -d postgres -p 5433
```

18.6.2. pg_upgrade###############

pg_upgrade#########PostgreSQL## ##--link################################## ####pg_dumpall############## ####initdb### ################## pg_upgrade ##################

18.6.3.

#########################PostgreSQL######################### ################### ####PostgreSQL################################ ################################ ########## #########PostgreSQL## ######################### ####################################

#######################################pglogical#Slony#Londiste#Bucardo########## #####################

18.7.

####PGDATA## ################ ###################

local##Unix############## #unix_socket_directories########## ## ## ########################/tmp## ########################/tmp/.s.PGSQL.5432#################### ########### ################/tmp##########################

local####################requirepeer#####################################
##

TCP###############SSL################################GSSAPI########## ####### ########################

SSL#################hostssl###20.1########SSL##########18.9########### TCP# #####sslmode=verify-ca####verify-full#################################

######32.19.1## #####sslrootcert=system########CA##############
##sslmode=verify-full#######################CA##########################

#########scram-sha-256#####################################SSL#################
##############1############## ####libpq#SCRAM###################
##channel_binding=require#################################### ###########
SCRAM###

GSSAPI#################hostgssenc###20.1########gss############ TCP######
#gssencmode=require#################

18.8.

PostgreSQL###
####### ##

#########

#################password_encryption###
################## SCRAM##MD5###
SCRAM#############PostgreSQL###
MD5###########################

###########

pgcrypto################################## ########################## #########
#####################################

################
###

################

Linux####################eCryptfs
#EncFS########FreeBSD##PEFS###### ############################Linux##dm-
crypt + LUKS#FreeBSD##GEOM######geli#gbde###### ############################
############## ###################Windows###################

##############
####################### ##################################### ###############
############################
#############

###############

SSL## pg_hba.conf#####
############################host#######SSL################hostssl######
####### ###SSL#####################################

GSSAPI## #############
########### ####pg_hba.conf#######################(host)######GSSAPI#
##########(hostgssenc)####### #######GSSAPI###########################
(gssencmode=require)#

Stunnel#SSH####################

SSL#####

################################ ##
#################### ### ############
###

############

##
##

18.9. SSL######TCP/IP##

PostgreSQL####SSL##############/####################### ######OpenSSL#######
#######################PostgreSQL############################17############

##SSL#TLS##TLS## SSL######TLS####
########SSL######SSL################################ SSL#PostgreSQL#TLS#####
########

18.9.1.

SSL####################PostgreSQL#####`postgresql.conf`####ssl######on#####
##TLS## ######TCP##########SSL##########
##########SSL############### ######################### ############SSL########
#################20.1##########

SSL## ################`server.crt`##
#`server.key`##
#ssl_cert_file#ssl_key_file#########################

Unix#######`server.key`########################### ###`chmod 0600 server.key`#
############# ################root##################################0640#######
######## ### PostgreSQL#######
##

##
##########################
###################

##########################
SSL#########################ssl_passphrase_command_supports_reload######### #######
#############Windows#############

`server.crt`#################################### ############################
################v3_ca## #####
##CA##########`true`######## #########################

`server.crt`################### ##

18.9.2. OpenSSL###

PostgreSQL########OpenSSL############## ###############`openssl.cnf`#####
##`openssl version -d`################## ########`OPENSSL_CONF`###############
##################

OpenSSL################################## #######OpenSSL####################
########`postgresql.conf`#ssl_ciphers############

##
`NULL-SHA`####`NULL-MD5`############################## ################ ############################ ################################## ####### ##NULL###########

18.9.3.

################################CA#####################
##postgresql.conf#ssl_ca_file################clientcert=verify-ca##
#clientcert=verify-full#pg_hba.conf####hostssl####### ######SSL#######
################ #######################32.19###########

clientcert=verify-ca#####hostssl#######################################
########## clientcert=verify-full#######################################
#################cn (Common Name)################# cert#####################
##################### #20.12####

##ssl_ca_file######################
#v3_ca############# ssl_crl_file####ssl_crl_dir#######################CRL########
##

#######clientcert####################pg_hba.conf#hostssl################
clientcert###
############

##

########pg_hba.conf#hostssl#cert############# ########SSL################
#################### ###20.12######## #cert################clientcert########
########## ###########cn (Common Name)############################

##########clientcert#######verify-ca####verify-full##########hostssl##
################################## ##################################cn
(Common Name)################################

18.9.4. #######SSL#########

18.2##########SSL##################
##########

#18.2 SSL###########

####	##	##
ssl_cert_file ($PGDATA/server.crt)	######	#######################
ssl_key_file ($PGDATA/server.key)	#######	######################## #######################
ssl_ca_file	########	######################## ######
ssl_crl_file	#############	########################

Windows#######################
############################

##################################
SSL############# Windows##SSL###
######## ################################

18.9.5.

365###########################OpenSSL#############*dbhost.yourdomain.com*#
##################

```
openssl req -new -x509 -days 365 -nodes -text -out server.crt \
  -keyout server.key -subj "/CN=dbhost.yourdomain.com"

############


chmod og-rwx server.key

#################################### ##############################OpenSSL#
############

###########################(CA)########CA#######################

########################################(CSR) ##############


openssl req -new -nodes -text -out root.csr \
  -keyout root.key -subj "/CN=root.yourdomain.com"
chmod og-rwx root.key

##############################Linux########OpenSSL##############


openssl x509 -req -in root.csr -text -days 3650 \
  -extfile /etc/ssl/openssl.cnf -extensions v3_ca \
  -signkey root.key -out root.crt

##################################


openssl req -new -nodes -text -out server.csr \
  -keyout server.key -subj "/CN=dbhost.yourdomain.com"
chmod og-rwx server.key

openssl x509 -req -in server.csr -text -days 365 \
  -CA root.crt -CAkey root.key -CAcreateserial \
  -out server.crt

server.crt#server.key########root.crt############# ####################
######################### root.key####################################

############################


# root
openssl req -new -nodes -text -out root.csr \
  -keyout root.key -subj "/CN=root.yourdomain.com"
chmod og-rwx root.key
openssl x509 -req -in root.csr -text -days 3650 \
  -extfile /etc/ssl/openssl.cnf -extensions v3_ca \
  -signkey root.key -out root.crt

# intermediate
openssl req -new -nodes -text -out intermediate.csr \
  -keyout intermediate.key -subj "/CN=intermediate.yourdomain.com"
chmod og-rwx intermediate.key
openssl x509 -req -in intermediate.csr -text -days 1825 \
  -extfile /etc/ssl/openssl.cnf -extensions v3_ca \
  -CA root.crt -CAkey root.key -CAcreateserial \
  -out intermediate.crt

# leaf
openssl req -new -nodes -text -out server.csr \
  -keyout server.key -subj "/CN=dbhost.yourdomain.com"
```

```
chmod og-rwx server.key
openssl x509 -req -in server.csr -text -days 365 \
  -CA intermediate.crt -CAkey intermediate.key -CAcreateserial \
  -out server.crt
```

server.crt#intermediate.crt############################## server.key### ################ ## ##root.crt################### root.key#intermediate.key################## #################

18.10. GSSAPI######TCP/IP##

##################################PostgreSQL###GSSAPI################ ######## GSSAPI####MIT Kerberos################################PostgreSQL############## ##########17##########

18.10.1.

PostgreSQL##########GSSAPI##############TCP#######################GSSAPI##### ##################### ### ## ##############GSSAPI################20.1#######

#######################GSSAPI#################GSSAPI########GSSAPI######## ###### ###########################PostgreSQL####################

#################GSSAPI######GSSAPI################## #############20.6###### ###

18.11. SSH###########TCP/IP##

#######PostgreSQL#####################SSH########### #######SSL################ ##########################

##SSH####PostgreSQL####################ssh############################## # ########################## ## ## ####################localhost## ############# #########################foo.com################

```
ssh -L 63333:localhost:5432 joe@foo.com
```

-L###1######63333###IANA#49152##65535### ################### ########IP##############################localhost#### 2## ####5432################################ ## 63333#######

```
psql -h localhost -p 63333 postgres
```

#####################foo.com#####joe####localhost######################### ##################################### ###SSH####PostgreSQL################### #######SSL########################## ######################################

################ssh######################joe@foo.com####ssh################ ###########

#############################

```
ssh -L 63333:foo.com:5432 joe@foo.com
```

####################foo.com##############################
#listen_addresses = 'localhost'######## ##################

###

```
ssh -L 63333:db.foo.com:5432 joe@shell.foo.com
```

shell.foo.com##db.foo.com#########SSH############## SSH################## ######################### ###SSH############

> ###
>
> ##

18.12. Windows####Event Log###

Windows#OS#event log#########################:

```
regsvr32 pgsql_library_directory/pgevent.dll
```

#######PostgreSQL##

###########event_source###########/n###/i#############:

```
regsvr32 /n /i:event_source_name pgsql_library_directory/pgevent.dll
```

OS##event log#########################:

```
regsvr32 /u [/i:event_source_name] pgsql_library_directory/pgevent.dll
```

> ##
>
> ############################eventlog#####
> #postgresql.conf#log_destination#########

#19#

################################## ##################################### ##### ####################

19.1.

19.1.1.

########################## ##############################enum######5######## ####### ########################

- *###:* ##on#off#true#false#yes#no#1#0#################################### #################
- *####:* ################## ########################## ########################## ######### ############SQL############################
- *###(##########):* ############################ ############################## ## ###########16#####0x#######8#####0############### ######################### 1000#################### 16##################
- *#######:* ############################## #################### ################ ###8################## ##################### pg_settings.unit ############## ## ############'120 ms'##################### #################### ######### ### ################## ################ #######
  - ##########B (###) kB (#####)#MB (#####)# GB (#####), TB (#####)### #########1024# ##1000########
  - ######## us (#####)#ms (###)#s (#)#min (#)#h (##)#d (##) ### ## #####30.1 GB#32319628902 B####30822 MB######## ##################################
- *###:* ############################ ########################## ######## pg_settings.enumvals ######### ###################

19.1.2.

#######################postgresql.conf################# data ################ ##

```
# This is a comment
log_connections = yes
log_destination = 'syslog'
search_path = '"$user", public'
shared_buffers = 128MB
```

1#####1############## ################ ### ### ################################### ### ## ##########################2#################-######### ######################### ############################

############################## ############################## ################## ####################

##################SIGHUP#################### ##########################
#pg_ctl reload#######SQL###pg_reload_conf()####### ####################
##(############################
####)# ################################## ################################
################################## ####################################
##SIGHUP ###########

postgresql.conf####PostgreSQL########### postgresql.auto.conf###########
postgresql.conf ################################## #######ALTER
SYSTEM################# #######postgresql.conf ###########################
######### postgresql.auto.conf##postgresql.conf###########

#####[76.170213]#######postgresql.auto.conf############ ALTER SYSTEM######
#################################### ##
(ALTER SYSTEM#####)################### ###########PostgreSQL.auto.conf###
######### allow_alter_system##################################### #######ALTER
############################### ##
###########ALTER############

########pg_file_settings##########################SIGHUP##################
###############

19.1.3. SQL############

PostgreSQL#3##SQL################# #######ALTER SYSTEM######SQL############
############; postgresql.conf##
####:

- ALTER DATABASE##############################
- ALTER ROLE##

ALTER DATABASE#ALTER ROLE##################################
##
#########################(######################)#

###################PostgreSQL####2##SQL(########)###########################

- SHOW################################ ######
  #current_setting(setting_name text)###9.28.1###########
- SET### ############
  ##################################SET######################### ####SQL##
  #set_config(setting_name, new_value, is_local)### #9.28.1###########

##########pg_settings################################

- ###############SHOW ALL######################## ###########################
  ##########
- #########UPDATE###########setting##########SET############### #####

  SET configuration_parameter TO DEFAULT;

  #########

  UPDATE pg_settings SET setting = reset_val WHERE name =
   'configuration_parameter';

19.1.4.

##PostgreSQL############## ## ##libpq##################################

- #####[60.944206]########-c#######################postgres############### ##########################-c##=#####-#############--##=########### #POSTGRES############## ######

  ```
  postgres -c log_connections=yes --log-destination='syslog'
  ```

  ###############postgresql.conf#ALTER SYSTEM############ ############### ###########################

- #####[80.847145]#libpq#####################PGOPTIONS################### # ## #########PGOPTIONS### #postgres####################-c################## ###################libpq #######PGOPTIONS###################### ################################## ################## ########PGOPTIONS#######POSTGRES################### #### ###-c######--################### ####

  ```
  env PGOPTIONS="-c geqo=off --statement-timeout=5min" psql
  ```

  ####################################SQL################################## ##

19.1.5.

PostgreSQL####postgresql.conf################################ ############## ################################

 ###########postgresql.conf#####*include*################## ################### ## ################################## ###### Include######################

```
include '#####'
```

Include##################

  include_if_exists############ ################################## ##include################# ###include##############include_if_exists##### ##################################

 include#######################include_dir########postgresql.conf######## ####### ##########

```
include_dir '#######'
```

#.conf###########include##### #####. ## ######### include################################(###### C ############### ####### ##################)#

include#########################postgresql.conf############################## ################ ###################################### ################ ################ ################################## ######################

####include#####
#postgresql.conf################

include 'shared.conf'
include 'memory.conf'
include 'server.conf'

##########shared.conf############# ######################memory.conf####
RAM#8GB############memory.conf#1####16GB###########################
######server.conf#######################

#####conf.d######
#postgresql.conf###############

include_dir 'conf.d'

####conf.d#########################

00shared.conf
01memory.conf
02server.conf

################################## ##
########### ######conf.d/02server.conf######conf.d/01memory.conf######
######

#####################################

00shared.conf
01memory-8GB.conf
02server-foo.conf

##################################
######################## ##

19.2.

#######postgresql.conf#######PostgreSQL####################2############
####################20######### ###3###########################data############
###############
##

data_directory (string)

#################### ####################

config_file (string)

####################postgresql.conf####### #######postgres###########
######

hba_file (string)

#########HBA###############pg_hba.conf####### ####################

ident_file (string)

########################pg_ident.conf####### #################### 20.2#
#######

`external_pid_file` (string)

#####################################PID############## ##################
###

####################################### ######data #######-D##############
#PGDATA################### data ##############

data#############################postgres#-D###############PGDATA##########
###########data######################postgresql.conf#############
##data_directory################## data_directory###############data ######
#######-D###PGDATA################

###########config_file#hba_file#ident_file############################
config_file#postgres##################################### ###3######
#data_directory#############-D###PGDATA#############

###########################postgres############################

19.3.

19.3.1.

`listen_addresses` (string)

############################TCP/IP########## ########################/######
####IP####### *##################IP############## ####0.0.0.0####IPv4#######
########::####IPv6############ ################IP################Unix###
################### ###############1##TCP/IP######################## ####
##########TCP/IP################ #######localhost######TCP/IP #loopback####
######

########(#20#)###########################listen_addresses#############
########## ## #############
##########

`port` (integer)

#######TCP########## 5432### ##########IP################################
#######################

`max_connections` (integer)

########################## #########100###############initdb################
################# #######################

######PostgreSQL#####_########################## ########################
#############

###
###################

`reserved_connections` (integer)

pg_use_reserved_connections#################################### #####
#####superuser_reserved_connections######
#superuser_reserved_connections#reserved_connections##############
#############pg_use_reserved_connections####################
superuser_reserved_connections#######################################
#####

########0##### #### max_connections - superuser_reserved_connections## ############ ######################

superuser_reserved_connections (integer)

PostgreSQL## ###max_connections######## ################ ########max_connections# #superuser_reserved_connections################################## ### ################reserved_connections##

########3##### #### max_connections##reserved_connections############## ####### ######################

unix_socket_directories (string)

####################listen#######Unix###################### ################### ########################### ############### ################################ ##### ####Unix########listen########### #####TCP/IP########################## ##

@##############Unix#########################Linux############## ######## ## ################### ############################@/tmp#######################

#########/tmp############### Windows##################Unix############### ############## ######################

.s.PGSQL.*nnnn*##############*nnnn*###########.s.PGSQL.*nnnn*.lock######## ######unix_socket_directories############### ####################### # ############################

unix_socket_group (string)

Unix## unix_socket_permissions###########Unix################################ ## ################################## ######################

########Windows############# ############# ############################## ##############

unix_socket_permissions (integer)

Unix################################ Unix###########Unix################### ####### ########chmod###umask############################ #######8######## #######0##################

######################0777####### #######0770########### ##unix_socket_group##########0700############ #Unix################### ##

########### #20##################

######################

########Solaris 10#####Solaris################################## ############### #########################unix_socket_directories##################### #########

##

bonjour (boolean)

Bonjour#############################off### ######################

`bonjour_name` (`string`)

Bonjour########### ###########''################################ ####
Bonjour########################### #######################

19.3.2. TCP Settings

`tcp_keepalives_idle` (`integer`)

################################TCP#keepalive########################### ###
########################### 0#################################### Windows
###########################0##############2########## #######
#TCP_KEEPIDLE###########################Windows########### #############
######### Unix###

`tcp_keepalives_interval` (`integer`)

TCP#keepalive## ####################
######### 0############################## Windows#######################
###0##############1######## #######TCP_KEEPINTVL#######################
####Windows########### ####################### Unix#######################
##########################

`tcp_keepalives_count` (`integer`)

##########################TCP keepalive############## 0####################
################## #######TCP_KEEPCNT##################################
#Windows######## ####################### Unix##############################
###################

`tcp_user_timeout` (`integer`)

###################################### ################################ 0##
################################### ########TCP_USER_TIMEOUT##########
#Windows##########################0########### UNIX#######################
############0########

`client_connection_check_interval` (`integer`)

########################
###

#########Linux#macOS#illumos#BSD##
################

############################ #######0################## ######################
##

#############TCP###
#####TCP keepalive#######PostgreSQL
#tcp_keepalives_idle#tcp_keepalives_interval#tcp_keepalives_count####################
##

19.3.3.

`authentication_timeout` (`integer`)

####################### ## #######
######################### ############################## ######1##1m#### ###
#####postgresql.conf###########################

`password_encryption` (enum)

CREATE ROLE####ALTER ROLE#################################### #### #####SCRAM-SHA-256###########scram-sha-256######MD5######### #md5### ######scram-sha-256###

#########SCRAM#########################SCRAM-SHA-256################## ########## ##########20.5########

`scram_iterations` (integer)

SCRAM-SHA-256################################ ######4096### ############ ## ########################## SCRAM-SHA-256######################## ################################

`krb_server_keyfile` (string)

Kerberos################### ######FILE:/usr/local/pgsql/etc/krb5.keytab### #############sysconfdir########## pg_config --sysconfdir############# ## ## ######postgresql.conf########################## ###20.6########

`krb_caseins_users` (boolean)

GSSAPI############################## ######off############### ####### #postgresql.conf##########################

`gss_accept_delegation` (boolean)

#########GSSAPI################## ######off### ########################## ####### ###on################################## ####### ##postgresql.conf#########################

19.3.4. SSL

SSL########18.9######### TLS##############################SSL############### ##################ssl########## SSL#####TLS############

`ssl` (boolean)

SSL############ ########postgresql.conf######################### ##### #off###

`ssl_ca_file` (string)

SSL######CA################## ############################## ####### ##postgresql.conf######################### ############CA############ ##################

`ssl_cert_file` (string)

SSL######################## ############################## ####### ##postgresql.conf######################### ###### server.crt ###

`ssl_crl_file` (string)

SSL###############CRL################## ############################## #### #####postgresql.conf######################### ############CRL######### #####ssl_crl_dir############

`ssl_crl_dir` (`string`)

SSL##############CRL################## ############################## ## ######postgresql.conf######################## ############CRL###### ######ssl_crl_file############

#########OpenSSL#####openssl rehash####c_rehash############## ######## ############

######################CRL################## ###CRL###################### ######## ################################ssl_crl_file######## ################

`ssl_key_file` (`string`)

SSL######################## ############################## ###### ##postgresql.conf######################## ###### server.key ###

`ssl_ciphers` (`string`)

SSL##########SSL################ ##################OpenSSL###### ciphers# ########### TLS#####1.2############################## ####TLS#####1.3###### ################### #######HIGH:MEDIUM:+3DES:!aNULL### ############### #########

#########postgresql.conf#########################

#########:

HIGH

HIGH####(####AES, Camellia, 3DES)##########

MEDIUM

MEDIUM####(#### RC4, SEED)##########

+3DES

OpenSSL#HIGH###################### 3DES#AES128############## 3DES# AES128############################## +3DES##########HIGH#MEDIUM### ##########

!aNULL

###################### ############MITM######################

OpenSSL############################## openssl ciphers -v 'HIGH:MEDIUM: +3DES:!aNULL' ##################OpenSSL#################### ######### ####################################

`ssl_prefer_server_ciphers` (`boolean`)

####SSL############################ ########postgresql.conf############ ############# ######on###

#####9.4####PostgreSQL########################## ######################### ####### ##

`ssl_ecdh_curve` (`string`)

ECDH################### ################################## ############# ################### ########postgresql.conf######################### # ######prime256v1###

OpenSSL################### `prime256v1` (NIST P-256), `secp384r1` (NIST P-384), `secp521r1` (NIST P-521). ###############`openssl ecparam -list_curves`##### ######TLS######################

`ssl_min_protocol_version` (`enum`)

####SSL/TLS#################### ######### #TLSv1#TLSv1.1#TLSv1.2#TLSv1.3### #######OpenSSL################### # ######################### TLS 1.0##################SSL#####2####3######### ##

######`TLSv1.2`#############################

#########`postgresql.conf`##########################

`ssl_max_protocol_version` (`enum`)

####SSL/TLS#################### #######ssl_min_protocol_version############### ########## ############################## ################################## ############################

#########`postgresql.conf`##########################

`ssl_dh_params_file` (`string`)

######DH#SSL###################################### ######################## ########DH############ ###################DH#########################DH## ################### `openssl dhparam -out dhparams.pem 2048`####### DH###################

#########`postgresql.conf`##########################

`ssl_passphrase_command` (`string`)

######SSL## ###################### #################

###########################0############# ########%p################## #%# #######%%######### ## ########### ##########

################################

########`postgresql.conf`############################

`ssl_passphrase_command_supports_reload` (`boolean`)

#####################################`ssl_passphrase_command`########## ###################### ########off#########`ssl_passphrase_command`###### ##################SSL########### ######################TTY############## ######## ##############################on#########

########`postgresql.conf`############################

19.4.

19.4.1.

`shared_buffers` (`integer`)

#################################### ##########128#####(`128MB`)### ########## ####################################initdb############ ########128############

##############################
#######BLCKSZ########8kB### #BLCKSZ############################## #######
###############

1GB###RAM#######################shared_buffers##################25%#
shared_buffers############################### ####PostgreSQL#########
################shared_buffers#RAM#40%##############################
shared_buffers###############max_wal_size####################
#######################################

1GB###RAM################################RAM######################

`huge_pages` (`enum`)

#####[72.516556]############huge page############### #####`try` (#####)#`on`#`off`#
`huge_pages`#`try`##########huge page###################### `on`###########
####################### `off`##huge page######### ################huge pages#
############### ########################### `huge_`#####################
huge pages################################ ###########huge pages##############
######### #####huge pages######## huge pages#########huge_pages_status########

##########Linux#Windows############# ########`try`############ Linux######
#`shared_memory_type`#`mmap`############################

huge page############################CPU################# ####18.4.5######
##

huge page#Windows##large page########## #########PostgreSQL#####Windows#####
############################## ##########################Windows###########
(gpedit.msc)######## Windows##
##################################(UAC)################# UAC##############
##################################

################################ Linux#FreeBSD#Illumos################
##PostgreSQL###########huge page##super# page#####large#page##################
########## Linux#######transparent huge pages# (THP)######## ########Linux######
#########PostgreSQL#################### ##############`huge_pages`#########
##############

`huge_page_size` (`integer`)

huge_pages#######huge page########## ########(`0`)### `0`################huge page
########## ######################

###64## `2MB`#`1GB` (Intel#
AMD)#`16MB`#`16GB` (IBM POWER)#`64kB`#`2MB`#`32MB`###`1GB` (ARM)# ###############
######18.4.5#########

###Linux#######################

`temp_buffers` (`integer`)

##################################
########### ######################################BLCKSZ########8kB### #
#####8######8MB#### #BLCKSZ#8kB################### #######################
###

#######`temp_buffers`######################## ############################
##########`temp_buffers`#####1##########64####### ######################
#####8192##########BLCKSZ###########

max_prepared_transactions (integer)

##################################PREPARE TRANSACTION########### #######
#####################

####
##
##max_prepared_transactions######max_connections###################

###
####################

work_mem (integer)

###################
############## #######4MB(4MB)### ######################################
###########################
########## ##################work_mem############## ####################
######ORDER BY#DISTINCT##################
#######memoize#######IN#########################

##############
##work_mem#hash_mem_multiplier############## ###################
#work_mem############################

hash_mem_multiplier (floating point)

#################################### ######
#work_mem#hash_mem_multiplier########## #######2.0##############
#work_mem######2############

##################################work_mem##############################
#########################hash_mem_multiplier############## ######2.0###
############################ #####2.0##8.0############work_mem####40MB####
####################

maintenance_work_mem (integer)

VACUUM#CREATE INDEX####ALTER TABLE ADD FOREIGN KEY#########
################# ################################## #######64######64MB##
1##################1###
#######work_mem################## ###########vacuum####################
#######

#######################autovacuum_max_workers##############################
####### ######autovacuum_work_mem################

autovacuum_work_mem (integer)

#################################### ################################### ####
##-1##maintenance_work_mem################ #########VACUUM############## ##
######postgresql.conf############################

vacuum_buffer_usage_limit (integer)

#####[85.672798]#VACUUM#######ANALYZE################################ 0##
################shared_buffers############# ###############128 kB#
#16 GB####### ########shared_buffers####1/8#####################
#######256 kB### ##################################
########BUFFER_USAGE_LIMIT##############VACUUM###ANALYZE###
######## ########VACUUM###ANALYZE#####################################
###################### ##############################*ANALYZE*############

####### 0##################shared_buffers####### ##############128## ###16############### ########shared_buffers#####1/8################## ###### ######2######## ########################### ################## ####_USAGE_LIMIT#########VACUUM###ANALYZE########### ############ ##ANALYZE###

logical_decoding_work_mem (integer)

######64#####(64MB)### ########################## ##################################max_wal_senders############work_mem#

commit_timestamp_buffers (integer)

######pg_###_ts############################## 65.1########## ######### ################BLCKSZ######8############## #######0####1024#####16###### #shared_buffers/512####### ########################

multixact_member_buffers (integer)

######pg_multixact/################################### 65.1########## ### ######################BLCKSZ######8############## #######32### ########## ##############

multixact_offset_buffers (integer)

######pg_multixact/offsets################################ 65.1####### ### ####units#################BLCKSZ######8############## #######16### #### ####################

notify_buffers (integer)

######pg_notify############################ 65.1########## ############### ############BLCKSZ######8############## #######16### ###################### ###

serializable_buffers (integer)

######pg_################################### 65.1########## ############## #############BLCKSZ######8############## #######32### ################### #####

subtransaction_buffers (integer)

######pg_subtrans################################## 65.1########## ####### ###################BLCKSZ######8############## #######0####1024#####16#### ###shared_buffers/512####### ########################

transaction_buffers (integer)

######pg_xact################################### 65.1########## ######### ################BLCKSZ######8############## #######0####1024#####16###### #shared_buffers/512####### ########################

max_stack_depth (integer)

######################## ####################################ulimit -s#### ###################1######################## ############################## ## ############################## ### #######2######2MB############################ ###################### ######### ###########SET##########################

max_stack_depth##
####### PostgreSQL## ###########
######################### ###################

shared_memory_type (enum)

PostgreSQL## ####
#mmap#mmap################sysv #shmget#######System V######windows
#Windows######### ##################################### ###################
################ ######################sysv################### ##########
################################ #18.4.1####

dynamic_shared_memory_type (enum)

###########################posix (shm_open#####POSIX#####)#sysv(shmget##
###System V#####)#windows (Windows#####)# mmap(############################
########)### ##################################### #########################
#######################mmap#####################################I/O###
######################### ##############pg_dynshmem#######RAM############
##########################

min_dynamic_shared_memory (integer)

################################ ##
###dynamic_shared_memory_type#######################################
####min_dynamic_shared_memory########
#######################huge_pages########### huge page###################
########################### #######0#none#### #######################

19.4.2.

temp_file_limit (integer)

##################################### ####################################
-1##
################### ################SET################

########## PostgreSQL ## ##############
##

max_notify_queue_pages (integer)

######NOTIFY/LISTEN######################## #######1048576### 8############
##8######################

19.4.3.

max_files_per_process (integer)

######1000#######
############################ ########################BSD##################
###Too many
open files########################### #######################

19.4.4. #######Vacuum##

VACUUM ### ANALYZE ################I/O#########################
######## ##########vacuum_cost_limit##########################
#vacuum_cost_delay##

##I/O######################## VACUUM ## # ANALYZE#################################### ################################ ################################ ######## vacuum#########################

#######VACUUM############################## ########vacuum_cost_delay###### ##########

vacuum_cost_delay (floating point)

################################ ################################ ########0## ########vacuum############ ############vacuum########

#######################vacuum_cost_delay####################1########### ## vacuum_cost_delay#1### ### ##############VACUUM# #########1#####################VACUUM#################### ############# #################vacuum_cost_delay################### ##############

vacuum_cost_page_hit (integer)

##################vacuum## ##############1###

vacuum_cost_page_miss (integer)

#######################vacuum## ##2###

vacuum_cost_page_dirty (integer)

vacuum################################ ################################I/O#### ##########20###

vacuum_cost_limit (integer)

#####[59.854015]#vacuum################################ ########200### ######## #######_###_################################ ######200###

> ##
>
> ##vacuum################## ##################################### ########################## #vacuum_cost_delay * 4 ################## vacuum_cost_delay * accumulated_balance / vacuum_cost_limit

19.4.5.

######################I/O############## ################################ ### ###########

bgwriter_delay (integer)

#####[83.203560]########################### ############################## ##################### ###bgwriter_delay##################### ######

#####################bgwriter_delay################ ###############
################ #######200####200ms#### ##################10####
bgwriter_delay#####10##############10##############################
#######postgresql.conf#########################
######################### ##
######### ###bgwriter_##################### #######################
##bgwriter_################### ########################## #######200##
#200 ms### ########################10###### bgwriter_#######10########
##########10##################### ########PostgreSQL.conf############
#############

bgwriter_lru_maxpages (integer)

############################
######100#######
#postgresql.conf############################

bgwriter_lru_multiplier (floating point)

###################
#############bgwriter_lru_multiplier########## ######################
########################## ####1###bgwriter_lru_maxpages##############
######1.0###
####################### ############################### ######2.0### #######
#postgresql.conf############################

bgwriter_flush_after (integer)

#######################OS##################### ########################
###########################fsync###########OS########################
################## ##
#shared_buffers######OS############################## ##################
####################################BLCKSZ########8kB###
############0###2MB##### #######Linux##512kB######0### (BLCKSZ#8kB######
##############BLCKSZ###########) #######postgresql.conf##############
###############

bgwriter_lru_maxpages###bgwriter_lru_multiplier#######################
#######I/O###

19.4.6.

backend_flush_after (integer)

#####################################OS################## ###############
#####################################fsync##############################
##OS############################## ####################################
##########shared_buffers######OS############################## ##########
########## ####################################BLCKSZ########8kB### ###
###################0###2MB##### ######0##(###############)# (BLCKSZ#8kB##
#######BLCKSZ###########)

effective_io_concurrency (integer)

PostgreSQL####################I/O########## #################PostgreSQL####
############I/O########## #######1##1000########I/O################## #####
########################

###########################RAID 0######RAID 1###########################
#RAID 5################### ##

############################ ################################CPU##########
######## SSD###

###I/O#####posix_fadvise######### ########################## ##############
###################### ##################Solaris###########################

######################1########0### ##
###################### (ALTER TABLESPACE#######)#

maintenance_io_concurrency (integer)

effective_io_concurrency##
##

####################10#######0### ##
################ALTER TABLESPACE###########

io_combine_limit (integer)

######I/O###########I/O########## ######128########

max_worker_processes (integer)

#####[76.923077]################################## ######################## ###
###8### ## ######################### ###
###8###

################
##################

#########
##max_parallel_workers#max_parallel_maintenance_workers#max_parallel_workers_per_gather#
################

max_parallel_workers_per_gather (integer)

###Gather###Gather Merge############################ ########
##max_parallel_workers#######max_worker_processes######################## ########
################################## ##
############### #######2### ######0######################

###########
###############
#######################work_mem##############################work_mem####
################################### ##
################# ###############4##################################5##CPU#
######I/O##################

######################15#########

max_parallel_maintenance_workers (integer)

####################################### ##################################
#####CREATE INDEX#B-tree##############FULL#######VACUUM### ########
##max_worker_processes###################max_parallel_workers########### ######
##################################### ##################################
#######2### 0################################

###############
########################## ##
#maintenance_work_mem############ ##############################CPU#
####I/O################

`max_parallel_workers` (integer)

#####[83.073930]#################################### #######8### ##########
##max_parallel_maintenance_workers#max_parallel_workers_per_gather################
#########max_worker_processes########################## max_worker_processes##
################################## ######################################
#######8###
##max_parallel_maintenance_workers#max_parallel_workers_per_gather############ ###
######max_worker_processes#################### ############################
##################

`parallel_leader_participation` (boolean)

###############Gather####Gather Merge################################
######on### ####off##
############################
########################

19.5. #########WAL#

###################28.5#########

19.5.1.

`wal_level` (enum)

wal_level########WAL########### ######replica##WAL################
minimal######
################################## ####logical##########################
####### ########################### #####################

minimal#########WAL########### ######################################
#######################14.4.7###########

```
ALTER ... SET TABLESPACE
CLUSTER
CREATE TABLE
REFRESH MATERIALIZED VIEW (CONCURRENTLY##)
REINDEX
TRUNCATE
```

####minimal WAL#################################replica#############
(archive_mode)################################ ###max_wal_senders#0######
####################### wal_level#minimal##############################
##############################

logical#####replica##########################WAL####################
logical####WAL####### ############REPLICA IDENTITY FULL#######(##:
ALTER TABLE##)####UPDATE#DELETE##################

9.6###################archive#hot_standby############ ###############
##replica#########

`fsync` (boolean)

##############PostgreSQL####fsync()##########################
#wal_sync_method################################## ##########################
##

fsync## ######### ########################fsync#########

fsync############################ ## ## ############################ ####################fsync######################## ######

fsync###(off)####(on)## ########### ########################fsync#######initdb --sync-only#### ##sync###

######################synchronous_commit################################ ##fsync#####################################

fsync #postgresql.conf############################## ############## ##full_page_writes####################

synchronous_commit (enum)

#####################################WAL######################## #### #remote_apply#on(#####)#remote_write#local#off###

synchronous_standby_names##############on#off##### remote_apply#remote_write#local####on############## off############## ###WAL######################### off############## ######################## ## #######wal_writer_delay#3## ### fsync############off################################## ## ################### ######synchronous_commit############################## ############################ ###28.4##########

synchronous_standby_names###########synchronous_commit##WAL############## ##################################

remote_apply## ## WAL#################### ############################## on## ################################## ## ################################## remote_write############################## ## #######PostgreSQL######## ################################## ## ####### ##################################### ####local################### ##################################### ## ##################

################### ##################################### ################### ##################################### ## ######################SET LOCAL synchronous_commit TO OFF#######

synchronous_commit########## 19.1######

#19.1 synchronous_commit###

synchronous_commit## #	local########	PG######## ########### ###	OS######## ########### ###	############
remote_apply	•	•	•	•
on	•	•	•	

synchronous_commit##	local########	PG######## ########### ###	OS######## ########### ###	############
remote_write	•	•		
local	•			
off				

`wal_sync_method` (enum)

WAL######################## `fsync`######################WAL########### ######################

- `open_datasync`#`open()`######`O_DSYNC`#WAL#########
- `fdatasync`######`fdatasync()`######
- `fsync`######`fsync()`######
- `fsync_writethrough`##################################`fsync()`######
- `open_sync`#`open()`######`O_SYNC`#WAL#########

################################## ## ###### ###Linux#FreeBSD##`fdatasync`######## ####################### ###### ## ####### 28.1######## ########`postgresql.conf`############################

`full_page_writes` (boolean)

##############PostgreSQL##WAL ######## ## ############################## ##WAL#################################### ################ ##WAL##### ######################## #WAL## ########### ########################1############################

###########`fsync`##################`fsync`################################

###########################PITR###WAL################ 25.3##########

########`postgresql.conf`############################ ######`on`###

`wal_log_hints` (boolean)

########`on`####PostgreSQL#######################################WAL###### ## ##

############################WAL## ##############WAL#########################

###################### #######`off`###

`wal_compression` (enum)

######################WAL########## #######PostgreSQL####`full_page_writes`### ###############WAL#################### ############WAL########## ##### #######`pglz`#`lz4`(PostgreSQL#`--with-lz4`##########)###`zstd`(PostgreSQL#`--`

with-zstd##########)### #######off### ##########SET#################### ##

#####################################WAL############## ####WAL######### #####WAL################CPU#################

wal_init_zero (boolean)

########on#################WAL############### #################WAL##### ################################ ####*Copy-On-Write* (COW)################################### ################################ off### ##

wal_recycle (boolean)

########on##WAL############# COW#### ##

wal_buffers (integer)

################WAL######################## ###########-1##shared_buffers#1/32## 3%################## ####64kB############16MB###WAL#################### ## ### ####32kB##############32kB# ########## #################WAL####################XLOG_BLCKSZ######### 8kB### ######################

WAL################################### ########################### ####### ## ########-1## ###################################

wal_writer_delay (integer)

#####[87.179487]#WAL####WAL################### WAL######################## ################wal_writer_delay######## ##########wal_writer_delay# ###########wal_writer_flush_after###WAL###########WAL############## ########################## ################################ #######200### #200ms#### ######################10###### 10########wal_writer_delay# ############10#################### #######postgresql.conf########### ################## ######WAL####WAL################### WAL########## ####wal_###_####################### ################################### ####### #############wal_###_##################wal_###_#####_after## WAL########WAL## ################ ########## ######200###200 ms### ##########################10#### ## wal_###_#######10####################10######################## ####### #PostgreSQL.conf##########################

wal_writer_flush_after (integer)

WAL####WAL################## ##########wal_writer_delay############## #wal_writer_flush_after###WAL###########WAL######################## ##############wal_writer_flush_after#0###########WAL##########WAL# ############ #################WAL####################XLOG_BLCKSZ###### ###8kB### #######1MB### ########postgresql.conf########################### ###

wal_skip_threshold (integer)

wal_level#minimal## ############## ################WAL############################fsync##### ## ## ## ############################ ######2#####(2MB)###

commit_delay (integer)

commit_delay##########WAL#################### ####################
######################################WAL############################
#################### ###########WAL##########commit_delay########
##
#commit_siblings########################### ####fsync################ #
################################ ######commit_delay########### ########
##SET#####################

9.3####PostgreSQL###commit_delay########################## ###WAL######
#######################WAL############################ PostgreSQL 9.3###
###

commit_siblings (integer)

commit_delay###################################### ####################
##1################################ ######5##########

19.5.2.

checkpoint_timeout (integer)

###WAL################### ############################## ######30###
1###### ######5##5min#### ################################## ######
#postgresql.conf###########################

checkpoint_completion_target (floating point)

######0.9###############
##################I/O## #########
################################ ################I/O#######################
#################I/O###### ########postgresql.conf######################
######

checkpoint_flush_after (integer)

###########################OS#################### ####################
##############################fsync###########OS######################
#################### ##
#shared_buffers######OS############################## ##################
######################################BLCKSZ########8kB###
###########0###2MB##### #######Linux##256kB######0### (BLCKSZ#8kb######
##############BLCKSZ###########) #######postgresql.conf##############
###############

checkpoint_warning (integer)

WAL###
#####max_wal_size################### ############################
#####30##30s#### ############# checkpoint_timeout#checkpoint_warning#
############### #######postgresql.conf###########################

max_wal_size (integer)

##WAL##########WAL############ ########################
##archive_command####archive_library####wal_keep_size##############
######WAL####max_wal_size########### ##############################
######1GB### ###################################
##postgresql.conf##############################

min_wal_size (integer)

#######WAL#################WAL#######################################
########################WAL##################WAL################
############ ################################ ######80MB### ######
##postgresql.conf##############################

19.5.3.

archive_mode (enum)

archive_mode#######archive_command####archive_library##############WAL#####
################# #######off####2##########on#always### ########2######
##########always##########################WAL############# always####
###(##)######### ###26.2.9#
########

archive_command#archive_library#######################
##archive_mode##archive_command#archive_library############### ####
################## wal_level#minimal##########archive_mode###########
###

archive_command (string)

####WAL################################ ########%p##################
######%f############# ## ###
###%############%%####### ################################ ####25.3.1######
##

#######postgresql.conf############################ ######
#archive_mode####archive_library###############
archive_command#archive_library################## archive_mode######
#######archive_command################archive_library#########
##WAL##WAL############# ####/
bin/true#Windows##REM#####################archive_command##########
##############################WAL#######################################
##

archive_library (string)

############WAL##################### ####################################
###archive_command####### archive_command#archive_library############
WAL##########################
postmaster########### #######25.3.1####49##########

#######postgresql.conf############################

archive_timeout (integer)

archive_command###archive_library ######WAL################## #########WAL##
##
########## ########################archive_timeout############WAL###
############################ ########0#################################
##
#################### ##
############ ##########archive_timeout############################
###1####archive_timeout#######
#######
#postgresql.conf############################

19.5.4.

##
#######

recovery_prefetch (enum)

###################WAL############################ #####off#on#try(####
#)### try################posix_fadvise########################### ######
#################### ##

#######################################I/O############ #################
#wal_decode_buffer_size###maintenance_io_concurrency###########

wal_decode_buffer_size (integer)

#######################WAL####################### #######################
######512KB###

19.5.5.

############################## ################################

##########################
########################### #####################################

###################standby.signal######################## #################
##WAL#################primary_conninfo##################
##restore_command######WAL#################################### ########
###19.6.3################ 19.5.6################################

#######################recovery.signal#######################
standby.signal#recovery.signal########################## ################
#WAL##########recovery_target############ ############19.5.6##############
##

restore_command (string)

###WAL## ##########################
######################## #####%f#######################%p################
%r#########################
####### ##
########################### %r####################### #26.2#### %###########%
%########

################################## ############################## ###########
########################

restore_command = 'cp /mnt/server/archivedir/%f "%p"'
restore_command = 'copy "C:\\server\\archivedir\\%f" "%p"' # Windows

############################SIGTERM#####################################
##

######## postgresql.conf##############################

archive_cleanup_command (string)

archive_cleanup_command###
###########################WAL########################## %r##############

################# ####################################%r################
#########
###archive_cleanup_command################pg_archivecleanup###########

archive_cleanup_command = 'pg_archivecleanup /mnt/server/archivedir %r'

##WAL######
############################ archive_cleanup_command##################
#26.2#### %##########%%########

##################################
##################################

#########postgresql.conf###########################

recovery_end_command (string)

################
recovery_end_command##
%r##archive_cleanup_command#######################WAL############

#############
###
###

#########postgresql.conf###########################

19.5.6.

#######WAL################ ################################## #
#recovery_target#recovery_target_lsn#recovery_target_name#recovery_target_time#r
########## ######################### ########################

recovery_target = 'immediate'

########################
###################

##########################'immediate'#############

recovery_target_name (string)

##############pg_create_restore_point()################################
##

recovery_target_time (timestamp)

################################## #######recovery_target_inclusive########

#timezone_abbreviations##
#timestamp with time zone########################## ########UTC########
##EEST####Europe/Helsinki#############

recovery_target_xid (string)

####################ID############## ########ID############################
################################## ########################### (#############
#########) ################ #######recovery_target_inclusive########

recovery_target_lsn (pg_lsn)

####################(WAL)####LSN############## ######
##recovery_target_inclusive######### ######### ########pg_lsn###########

##

recovery_target_inclusive (boolean)

#####################(on)#############(off)######
recovery_target_lsn#recovery_target_time###recovery_target_xid################# ####
#########WAL###(LSN)###################ID################################
######## ######on###

recovery_target_timeline (string)

######################## ###########ID######## current##################
######################### latest#######################################
######latest###

######ID#16#########(#####WAL######################)#0x######## #####WAL
#######00000011000000A10000004F###########ID#0x11(###10###17)######

########
#25.3.6##########

recovery_target_action (enum)

############################## ######pause################### promote#####
################################## ####shutdown#######################

pause###
############## #########pg_wal_replay_resume()(# 9.97##)#################
################### ###
#######################

shutdown################################## ############WAL###########(#
########################WAL################)#

recovery.signal#recovery_target_action#shutdown####################
##########recovery.signal################################

################################## hot_standby############pause####shutdown#
########### ##################pause#promote###########

###

19.5.7. WAL Summarization

##WAL#####

summarize_wal (boolean)

#####[58.867925]#SSL############ ########postgresql.conf################
########## ######off### ######WAL################ ######WAL###############
################## ########PostgreSQL.conf######################### #####
######

######wal_####minimal##############summarize_wal=on############## ####
#########wal_###=minimal#summarize_wal=on#######################wal_
###=minimal#####WAL################

wal_summary_keep_time (integer)

######WAL############WAL#################### ############################
##################### ##
################## WAL######################WAL##########################

############################ ####################WAL#################### ## ########PostgreSQL.conf####### ################### ######################### ######10##### summarize_wal= ######WAL#############################WAL###########

19.6.

#############################26.2.5#########################29############## 26.2.5

#############26.2.7## ##### ### #########################

######################CREATE PUBLICATION######################CREATE SUBSCRIPTION############### ################################## ########### ################### ####################29.11#########

19.6.1.

######################################

max_wal_senders(integer)

WAL############# ######10### 0########################## ############## ## ################################## ## ##################### ############ ##########wal_level#replica################

######

max_replication_slots(integer)

####################(26.2.6##)########## ######10### ################### ## ## ######################## ##wal_level#replica##############

##

wal_keep_size(integer)

pg_wal#############WAL################ ################### #wal_keep_size#######################################WAL############## #### ##WAL######## ##

pg_wal######################### #####WAL################################ ##################### ##wal_keep_size################################### ############### ##################WAL######################WAL######## ########## ############################## ########postgresql.conf####### #####################

max_slot_wal_keep_size(integer)

#####################pg_wal############## ###WAL############## max_slot_wal_keep_size#-1 (#######)##################WAL#########

####################restart_lsn####LSN############################ ########WAL################################ #############WAL######### #pg_replication_slots######### ############################## ####### ##postgresql.conf###########################

`wal_sender_timeout` (`integer`)

################################## ## ######## ################################ ########60#### ################### ##

#####

`track_commit_timestamp` (`boolean`)

###################### ########postgresql.conf######################### ### ######off###

`synchronized_standby_slots` (`string`)

##WAL################ #### WAL#########################WAL################################## ###### ## # ## ## #####synchronized_###_slots#################################### ############ ################## pg_replication_slot_advance, pg_logical_slot_get_changes, and pg_logical_slot_peek_changes, when used with logical failover slots, will block until all physical slots specified in synchronized_standby_slots have confirmed WAL receipt.

######synchronized_#####_slots############################sync_##### ###_slots=##

19.6.2.

##wal_level################################WAL##################19.5.3######## ### ##

`synchronous_standby_names` (`string`)

26.2.8## ###############1###### ##### ## ########### ## #pg_stat_replication ######streaming############# ################### ##################################### ## ############

##application_name##### ######### ########primary_conninfo##### ######cluster_name##########walreceiver### ## ###################### ##########################

##

[FIRST] *num_sync* (*standby_name* [, ...])
ANY *num_sync* (*standby_name* [, ...])
standby_name [, ...]

####*num_sync*################################ *standby_name*############
FIRST#ANY##############################

#####FIRST#*num_sync*##*num_sync*##
#######WAL#################################### ####FIRST 3 (s1, s2, s3, s4)#####s1#s2#s3#s4##############3########################## #####
###################################### ################################
###################### ##
#####FIRST########

#####ANY#*num_sync*##
#*num_sync*#########WAL#################################### ####ANY 3 (s1, s2, s3, s4)#####s1#s2#s3#s4########3####################

FIRST#ANY############### ############################*standby_name*#####
#############

3#######PostgreSQL9.6#################################### #####
##FIRST#*num_sync*#1######### #####FIRST 1 (s1, s2)#s1, s2####### s1#s2#
###############

#######*##################

######################## ##

> ##
>
> ###*standby_name*##*########SQL################## ##############
> ## ##################*standby_name*################################
> ############

###
##########################synchronous_commit#####local ### off############
#################################

#######postgresql.conf############################

19.6.3.

######################

primary_conninfo (string)

################################## #######32.1.1################## ############
######################## (32.15 ##) ######## ############################

#####[65.024631]###
########## ## (26.2.5.1 ##)# ##
################################ ###### primary_conninfo#############
##################~/.pgpass########### (######## replication ###
##)# primary_conninfo######################## ####################
############
############################26.2.5.1##### ################################

########_conninfo#####################~/.pgpass####################

######################47.2.3##############_conninfo######dbname######## #### ################### ###############

########postgresql.conf###################### WAL###################### ####################################primary_conninfo############# #### #########################

primary_slot_name (string)

###
########26.2.6##### ########postgresql.conf###################### WAL#### ##
primary_conninfo####################

hot_standby (boolean)

26.4###on### ############ ######### ##

max_standby_archive_delay (integer)

##########################26.4.2######################WAL################ ###################### max_standby_archive_delay##WAL####WAL######## ############################## ############################## ######30#### #-1 ################################## ########postgresql.conf############## ##############

max_standby_archive_delay### # ########WAL########################### #############WAL############## ####################################

max_standby_streaming_delay (integer)

########################## 26.4.2######################WAL################ ###################### max_standby_streaming_delay################### WAL################ ############################ ######30#### #-1######## ########################## ########postgresql.conf################### #########

max_standby_streaming_delay### ## ####################WAL####################### ####################
###

wal_receiver_create_temp_slot (boolean)

###############(primary_slot_name####)################WAL################## ########################## ######off### ########postgresql.conf########## ########### WAL###
##

wal_receiver_status_interval (integer)

#########WAL## #### ######pg_stat_replication################# ######################### ## ################### ########### ## ############# ############################## ########WAL########

##synchronous_commit#remote_apply############ ################### ########### ############################ #######10#### ###### ##postgresql.conf##########################

hot_standby_feedback (boolean)

############wal_receiver_status_interval###2###### #### #######off### ########postgresql.conf##########################

###
#####################

wal_receiver_timeout (integer)

#################################### ##################################### ################ ############################### #######60#### ############ ######## #######postgresql.conf##########################

wal_retrieve_retry_interval (integer)

WAL######(###################pg_wal####WAL#####)################### WAL################################ ########postgresql.conf########## ################# ################################ #######5#### ###### ##postgresql.conf##############################

##########################WAL################################### ######## ####################################WAL############################## WAL ##############################WAL######################## ############### ########################

###

recovery_min_apply_delay (integer)

##############################WAL############# ########################## ######################## ##################################### ######## #5min###5############## ################### ###################### ######0###########

#################WAL#### ## ################################ ############################ ## ###################### ##

################WAL############ ######################## ################## ##########################MVCC## ########

###################

WAL################################### #############WAL############## ####pg_wal#####################

###
######## ############hot_standby_feedback############################
######################

##
synchronous_commit#remote_apply################################ ###COMMIT#################

#########postgresql.conf#########################

sync_replication_slots (boolean)

##
###########################

#####[78.102190]##########postgresql.conf######################### ######
################## #########PostgreSQL.conf#########################

19.6.4.

#################################### ##################### See 29.11 for more details.

max_replication_slots (integer)

####################(origin)##48######################### ####################
################################ ####################
#pg_replication_origin_status##################################
max_replication_slots###
########### #48#

##

max_logical_replication_workers (integer)

######################### ##

################max_worker_processes#################

#######4### #######################

max_sync_workers_per_subscription (integer)

######################### ##
#############

##############################

########max_logical_replication_workers#################

#######2### #########postgresql.conf###########################

max_parallel_apply_workers_per_subscription (integer)

########################## #######################streaming = parallel#
##

##########max_logical_replication_workers#################

#######2### #########postgresql.conf###########################

19.7.

19.7.1.

########################
################################1################################## ########
#######################################19.7.2#####ANALYZE#######
##default_statistics_target###############ALTER TABLE SET STATISTICS###########
#######################

enable_async_append (boolean)

######on###

enable_bitmapscan (boolean)

##on###

enable_gathermerge (boolean)

######on###

enable_group_by_reordering (boolean)

##BY##############################
###################BY###############BY###################
################ ###### ###########

enable_hashagg (boolean)

######on###

enable_hashjoin (boolean)

##on###

enable_incremental_sort (boolean)

################################### ######on###

enable_indexscan (boolean)

##on###

enable_indexonlyscan (boolean)

(11.9########) ######on###

enable_material (boolean)

############################## ##############################off##############
##on###

enable_memoize (boolean)

#######
##########################
######################### ######on###

enable_mergejoin (boolean)

##on###

enable_nestloop (boolean)

################################
######on###

enable_parallel_append (boolean)

######on###

enable_parallel_hash (boolean)

######on###

enable_partition_pruning (boolean)

##################
######on### ###5.12.4######
##

enable_partitionwise_join (boolean)

############(partitionwise join)############################ ################
########################
#####
##################work_mem##
##################CPU#####
############ #######off###

enable_partitionwise_aggregate (boolean)

###partitionwise###
######################## GROUP BY#######################################
################ #######################work_mem##########################
######################## ## #######
############CPU################# #######off###

enable_presorted_aggregate (boolean)

################ORDER BY/DISTINCT#################################### #
################ORDER BY###DISTINCT#################################
################# ### #
######on###

enable_seqscan (boolean)

################################
######on###

enable_sort (boolean)

################################
######on###

enable_tidscan (boolean)

##########TID######################### ######on###

19.7.2.

######################## ##
############ ###################################### ####seq_page_cost####

#1.0########################## ##
##

> **##**
>
> ## ##############################
> ################# ###############################

seq_page_cost (floating point)

######1.0###
###ALTER
TABLESPACE##########

random_page_cost (floating point)

######4###
##ALTER TABLESPACE##########

seq_page_cost ##
################ ##############CPU##########I/O################### ######
################

######################################4############ ##############4.0#######
##########
#####################40################90%################################
######

#############90###
random_page_cost ############## ################################random_page_cost
###SSD######################
##########################random_page_cost ###1.1#######################

> **###**
>
> #####random_page_cost#seq_page_cost############################
> ############## ##############RAM############################## #####
> ################################## ######################CPU########
> ############## RAM##################################

cpu_tuple_cost (floating point)

######0.01###

cpu_index_tuple_cost (floating point)

######0.005###

cpu_operator_cost (floating point)

##0.0025###

parallel_setup_cost (floating point)

######1000###

parallel_tuple_cost (floating point)

################1#################################### ######0.1###

min_parallel_table_scan_size (integer)

################################ ######################################
##########################
###########BLCKSZ########8kB### ######8###### 8MB####

min_parallel_index_scan_size (integer)

######################
######################### ## ######
#################VACUUM####################### VACUUM####### #######
##############################BLCKSZ########8kB### ######512######512kB##
##

effective_cache_size (integer)

##########################
##
###############PostgreSQL#######PostgreSQL############################
################################### ####################################
################ ########PostgreSQL####################################
################ ################ ######################################
######################################BLCKSZ########8kB### ######4#####
#4GB#### (BLCKSZ#8kb####################BLCKSZ###########)

jit_above_cost (floating point)

JIT#######30##########JIT######################## JIT####################
################## ###-1#####JIT############# ######100000###

jit_inline_above_cost (floating point)

######JIT################################### ############################
######### ###jit_above_cost################## ###-1################### #
#####500000###

jit_optimize_above_cost (floating point)

######JIT############################### ##################################
###jit_above_cost################jit_inline_above_cost########
######### ###-1################## ######500000###

19.7.3.

###############GEQO## ################
######60##########

geqo (boolean)

################################## ####################geqo_threshold##
##GEQO#####################

geqo_threshold (integer)

############FROM## #FULL OUTER JOIN#
#####FROM########################## ######12### ##########################

##
###################GEQO#################

`geqo_effort` (`integer`)

GEQO#####################################1##10################### ########
5###

##`geqo_effort`#############GEQO#######################################
#######################################

`geqo_pool_size` (`integer`)

GEQO#####################################2###########100##1000#########
#######################`geqo_effort`################################

`geqo_generations` (`integer`)

GEQO##################################1############################## #
##0#####################`geqo_pool_size`###########

`geqo_selection_bias` (`floating point`)

GEQO################# #########################1.50##2.00##2.00########

`geqo_seed` (`floating point`)

#############GEQO################################### ##0########1########
###

19.7.4.

`default_statistics_target` (`integer`)

`ALTER TABLE SET STATISTICS`#################################### ######
#`ANALYZE`############################## ######100### PostgreSQL###########
######################14.2#########

`constraint_exclusion` (`enum`)

#################################### `constraint_exclusion`########on##
#################off################partition#############`UNION ALL`###
############### `partition`########### ###########################

#################################CHECK################################
#############

```
CREATE TABLE parent(key integer, ...);
CREATE TABLE child1000(check (key between 1000 and 1999))
 INHERITS(parent);
CREATE TABLE child2000(check (key between 2000 and 2999))
 INHERITS(parent);
...
SELECT * FROM parent WHERE key = 2400;
```

#############`SELECT`###`child1000`###############

####
###
#########################
######enable_partition_pruning########

##################################5.12.5########

`cursor_tuple_fraction` (`floating point`)

############################## ######0.1### ################################
###################### ##
########1.0###########################
##

`from_collapse_limit` (`integer`)

######FROM## ####################
#################### ######8### ###14.3#########

####geqo_threshold############GEQO###########################19.7.3########
##

`jit` (`boolean`)

PostgreSQL#######JIT###################30###### ######on###

`join_collapse_limit` (`integer`)

############################JOIN###FULL JOIN#####FROM########### ######
##

###########from_collapse_limit######################### ###1########
#JOIN################ ##
################################ ################1#########################
###14.3#########

####geqo_threshold############GEQO###########################19.7.3########
##

`plan_cache_mode` (`enum`)

######################PL/pgSQL####################################(generic)##
############# ################################## #######################
############### ##
################ ####################plan_cache_mode######## #####auto##
######force_custom_plan#force_generic_plan### #######################
################## ###PREPARE########

`recursive_worktable_factor` (`floating point`)

##############
#######10.0### 1.0###############
##############################(fanout)########### ############################
#######

19.8.

19.8.1.

`log_destination` (`string`)

PostgreSQL##stderr#jsonlog#csvlog#jsonlog###syslog##############################
Windows###eventlog######### ###################################
###stderr######### ########postgresql.conf##############################

csvlog#`log_destination`#######################################CSV#######
###19.8.4########## CSV###############logging_collector##############

jsonlog#`log_destination`###########################JSON########## ###19.8.5#
######### JSON###############logging_collector##############

stderr#csvlog####jsonlog##################`current_logfiles`#################
##################################### ###################################
#################:

```
stderr log/postgresql.log
csvlog log/postgresql.csv
jsonlog log/postgresql.json
```

`current_logfiles`##################################`log_destination`##
################ stderr#csvlog#jsonlog#####`log_destination`##################
####################

> ##
>
> `log_destination`#syslog#######################Unix##########syslog
> ####################### PostgreSQL#####`LOCAL0`##`LOCAL7`###syslog######
> ###########syslog_facility########## #######################syslog#####
> ################# ##########
>
> ```
> local0.*    /var/log/postgresql
> ```
>
> syslog#######################
>
> Windows#`log_destination`###`eventlog`############Windows Event
> Viewer ##
> ### ###18.12#######

`logging_collector` (`boolean`)

#################### ###stderr##
#####syslog############### ############syslog################## #############
##########################`archive_command`##############################
####################

> ##
>
> ############stderr############## ############stderr#################### ##
> ### ###############
> ##
> ######################

> ##
>
> ################################ ####################################
> ################################## ########syslog####################
> ######### ##

log_directory (string)

logging_collector## ########## ### #######postgresql.conf###### ##################### #####log###

log_filename (string)

logging_collector#################################### ##strftime### ##########%## ########%########### ##log_timezone################## ########%-#####strftime[1] ########## #### #strftime#################################### #####postgresql-%Y-%m-%d_%H%M%S.log###

8.4########PostgreSQL###%####################################

CSV#####log_destination#######################.csv########CSV###### ############ #log_filename#.log####################

JSON#####log_destination#######################.json########JSON### ############### #log_filename#.log####################

#########postgresql.conf##########################

log_file_mode (integer)

Unix#########logging_collector## #Microsoft Windows################# #######chmod ### umask################# ##################### #####8##################0################

##############0600################################### ##############0640## #################### ################log_directory######################### #####################

#########postgresql.conf##########################

log_rotation_age (integer)

logging_collector################################## ################### ########### ############################## #####24##### ################### ################### ########postgresql.conf############################ ##

log_rotation_size (integer)

logging_collector################################## ################### ##################### ################################## #####10####### # ##################################### #######postgresql.conf########### ###################

log_truncate_on_rotation (boolean)

logging_collector################PostgreSQL########################## #################### ## ##################### ########################## ########postgresql-%H.log####log_filename###########24################################### ### #######postgresql.conf##########################

[1] https://pubs.opengroup.org/onlinepubs/009695399/functions/strftime.html

##7############### server_log.Mon#server_log.Tue################
####################### log_filename # server_log.%a#
##log_truncate_on_rotation # on######log_rotation_age # 1440######

##24########1#####1#########################1#######################
##log_filename#server_log.%H%M###log_truncate_on_rotation#on#
##log_rotation_age#60#####log_rotation_size#1000000######
log_filename#%M###

`syslog_facility` (enum)

syslog####################syslog##facility##############
LOCAL0#LOCAL1#LOCAL2#LOCAL3#LOCAL4#LOCAL5#LOCAL6#LOCAL7###########
######LOCAL0### ###########syslog################### #######
#postgresql.conf#############################

`syslog_ident` (string)

syslog###################syslog####PostgreSQL##############################
##postgres### ########postgresql.conf#############################

`syslog_sequence_numbers` (boolean)

syslog##[2]##################### ###
######syslog############--- last message repeated N times ---##
############## ######syslog#####################################rsyslog####
#$RepeatedMsgReduction################## ##############################
##

#########postgresql.conf#############################

`syslog_split_messages` (boolean)

syslog################################syslog############ ####################
############syslog##########1024############ ######PostgreSQL##############
##syslog########## ############################syslog###########

##syslog## ################# ###
syslog## ####syslog##############
#####################################

########postgresql.conf#############################

`event_source` (string)

event log###########################PostgreSQL##############################
#####PostgreSQL###########postgresql.conf###########################
##

19.8.2.

`log_min_messages` (enum)

############################ ####
#DEBUG5#DEBUG4#DEBUG3#DEBUG2#DEBUG1#INFO#NOTICE#WARNING#ERROR#LOG#FATAL#
###PANIC### ## #####
#WARNING### ####LOG######client_min_messages################## ###########
#SET###########################

`log_min_error_statement` (enum)

##############SQL################### #######################SQL#######
####

##DEBUG5#DEBUG4#DEBUG3#DEBUG2#DEBUG1#INFO#NOTICE#WARNING#ERROR#LOG#FATAL#PANIC#
######ERROR### ##
###################PANIC######### ###########SET#######################
###

log_min_duration_statement(integer)

####250ms#######250ms#####
###########SQL########### ##
######### ############################### 0######################### -1#####
################### ###########SET##########################

###log_min_duration_sample##
###

#########################Parse#Bind#Execute#######################

##

########log_statement#########log_statement#########################
####### syslog##############ID######ID################################
###log_line_prefix#####PID########ID###############

log_min_duration_sample(integer)

################################ #####log_min_duration_statement#####
##log_statement_sample_rate## #####100ms#
######100#########SQL############### ####################################
################### ######################## ###########################
-1################################## ###########SET#################
#########

#####log_min_duration_statement#############
##log_min_duration_statement##############################

log_min_duration_statement####################

log_statement_sample_rate(floating point)

log_min_duration_sample#################### ################0.5#2######
######################## ######1.0######################## #######
##log_min_duration_sample#-1########################### ###########
#SET###########################

log_transaction_sample_rate(floating point)

##################################### ############################### ####
#############0.1#10################################### ######0#############
################### 1################################
log_transaction_sample_rate########################### ###########
#SET###########################

##

##

`log_startup_progress_interval` (integer)

#10#### 0### ################### # #######postgresql.conf#############################

#################25########unlogged###########8######### ###########10### ###########################10########20################unlogged########### ##############

19.2##PostgreSQL######################## #####syslog###Windows#eventlog###########

#19.2

###	####	syslog	eventlog
DEBUG1 .. DEBUG5	##########################	DEBUG	INFORMATION
INFO	VACUUM VERBOSE############# ##################	INFO	INFORMATION
NOTICE	############################## ######	NOTICE	INFORMATION
WARNING	##############COMMIT######## ########	NOTICE	WARNING
ERROR	############################	WARNING	ERROR
LOG	############################## ###	INFO	INFORMATION
FATAL	#############################	ERR	ERROR
PANIC	############################### #####	CRIT	ERROR

19.8.3.

> **##**
>
> ################################ 24.3#########

`application_name` (string)

application_name#NAMEDATALEN##################64############## ######## ######################### #####pg_stat_activity########CSV############ ##log_line_prefix######################### application_name#######ASCII## ######### #####C#####16################

`debug_print_parse` (boolean)
`debug_print_rewritten` (boolean)
`debug_print_plan` (boolean)

############################## ## ############ #########LOG## ## client_min_messages#log_min_messages######################### ############ ########

debug_pretty_print (boolean)

######
##debug_pretty_print#debug_print_parse#debug_print_rewritten###
#debug_print_plan################## ##################################
################

log_autovacuum_min_duration (integer)

###############autovacuum###################### ###############autovacuum##
######### -1#autovacuum############ ############################## ######
#250ms######250ms##########autovacuum#analyze########### ####-1##########
##################################autovacuum########################## ###
###10min### ################autovacuum############ #######
#postgresql.conf############################## ########################
############################

log_checkpoints (boolean)

#################################### ####################################
############ #######postgresql.conf########################## ######on#
##

log_connections (boolean)

###########SET##############
#################################### ######off###

> ##
>
> psql#####################################2##############connection received#########################

log_disconnections (boolean)

############## #######log_connections######################## #########
##SET## ######off###

log_duration (boolean)

############################### ######off### ###########SET###############
###########

#########################Parse#Bind#Execute#######################

> ##
>
> log_duration#######log_min_duration_statement#0##########
> ##log_min_duration_statement################################
> ################## #####log_duration#on##
> ##log_min_duration_statement################################
> ########################## ################################

log_error_verbosity (enum)

#####TERSE#DEFAULT###
#VERBOSE################################## TERSE#DETAIL#HINT#QUERY###

#CONTEXT########### VERBOSE####SQLSTATE########A#################### ###################### ###########SET##########################

log_hostname (boolean)

#########################IP############ ################################ ### ### ####### #postgresql.conf############################

log_line_prefix (string)

################printf######## %## ################ ################### ##################################### ## #####%############################## ############## ## ################### ##############

########postgresql.conf################################## ############ ######ID#######'%m [%p] '###

#####	##	#######
%a	#########	#
%u	####	#
%d	#######	#
%r	##########IP#############	#
%h	##########IP####	#
%b	#########	×
%p	#######	×
%P	######################## ######################	×
%t	############	×
%m	###########	×
%n	############Unix###### ##	×
%i	######################	#
%e	SQLSTATE ######	×
%c	#####ID#####	×
%l	#######################1# #######	×
%s	##############	×
%v	Virtual transaction ID (procNumber/localXID); see 66.1	no
%x	########ID############## #0## 66.1##	×
%q	######### ############### ################ ######## ##########	×
%Q	######################## ################### ### #compute_query_id########	#

#####	##	#######
	######################## ######################## #######	
%%	%######	×

##########pg_stat_activity####backend_type############################
########

%c#######2##4####16### #############
########ID### ######%c#####################################
##pg_stat_activity##############################

```
SELECT to_hex(trunc(EXTRACT(EPOCH FROM backend_start))::integer) || '.'
 ||
       to_hex(pid)
FROM pg_stat_activity;
```

> ###
>
> log_line_prefix###
> ######## ###############
>
> ###
>
> Syslog################ID######## #########Syslog#######################
> #############
>
> ###
>
> %q## %q
>
> log_line_prefix = '%m [%p] %q%u@%d/%a '
>
> ##
>
> ##################log_statement#######################%Q#####
> ##log_statement######################

log_lock_waits (boolean)

#################deadlock_timeout################################## ##########
########################### ######off### ###########SET####################
#######

log_recovery_conflict_waits (boolean)

############deadlock_timeout## #
##################WAL############################

######off### #######postgresql.conf##########################

log_parameter_max_length (integer)

-1############################## ######################## ###### ###########SET##########################

######log_statement#log_duration############################## ################ ################################### ##################

log_parameter_max_length_on_error (integer)

-1#################### ###########################

########################### ################PostgreSQL#################### ################### ## ##### ################################

log_statement (enum)

##SQL################ ######none#off##ddl#mod####all######## ddl##CREATE#ALTER####DROP######################## mod#####ddl### ##INSERT#UPDATE#DELETE#TRUNCATE####COPY FROM################### PREPARE#EXECUTE###EXPLAIN ANALYZE################################ ## ##########################Execute#############Bind########################## ######################

######none### #############SET######################

> ##
>
> ####################################log_statement = all## ################################ #########################Execute ################################## ############## ##log_min_error_statement#ERROR#############
>
> ##

log_replication_commands (boolean)

#####[69.791667]######################## ##################53.4####### ### ####off### ###########SET########################## ################### #walsender##################/################## ################53.4##### ##### ########### #############SET######################

log_temp_files (integer)

########################### ## #### ## 0##### ######################## ################################## ################## ################ #########-1############### ############SET################ ########### ########################SET############

log_timezone (string)

########################## TimeZone## ############### #############GMT####postgresql.conf#############initdb###

############################ ###8.5.3######### ########postgresql.conf#### ########################

19.8.4. CSV##########

log_destination####csvlog## ###########CSV################## ######timestamp# ##### ######## ####ID# ###### ########## #####ID# ########## ####### ########## ##########ID# ##########ID# ### ##### SQLSTATE#### ######### ########### #### ######################## ######## ############ ####### #################(###log_min_error_statement######### #)# ######### PostgreSQL###############log_error_verbosity#verbose######### #### ####################################ID######ID# ###csvlog################### ########

```
CREATE TABLE postgres_log
(
  log_time timestamp(3) with time zone,
  user_name text,
  database_name text,
  process_id integer,
  connection_from text,
  session_id text,
  session_line_num bigint,
  command_tag text,
  session_start_time timestamp with time zone,
  virtual_transaction_id text,
  transaction_id bigint,
  error_severity text,
  sql_state_code text,
  message text,
  detail text,
  hint text,
  internal_query text,
  internal_query_pos integer,
  context text,
  query text,
  query_pos integer,
  location text,
  application_name text,
  backend_type text,
  leader_pid integer,
  query_id bigint,
  PRIMARY KEY (session_id, session_line_num)
);
```

##################COPY FROM##############

```
COPY postgres_log FROM '/full/path/to/logfile.csv' WITH csv;
```

#####file_fdw#####################################

CSV#######################################

1. ##############################log_filename###log_rotation_age######### ##
2. ####################log_rotation_size#0########################
3. ####################################log_truncate_on_rotation#on####### ###

4. ######################## ##########2######################## COPY######### ##########################1######################## ########################### ##################################### ################################### ######COPY##

19.8.5. JSON#######

log_destination####jsonlog#################################### ####### ##JSON############

NULL######################## ###################### jsonlog################### ######################

#####JSON################################## 19.3#######

#19.3 JSON#########

###	#	##
timestamp	string	#############
user	string	####
dbname	string	#######
pid	number	####ID
remote_host	string	#########
remote_port	number	#########
session_id	string	#####ID
line_num	number	###########
ps	string	###ps##
session_start	string	#########
vxid	string	##########ID
txid	string	##########ID
error_severity	string	######
state_code	string	SQLSTATE###
message	string	########
detail	string	##########
hint	string	###########
internal_query	string	###################
internal_position	number	###################
context	string	#########
statement	string	#################
cursor_position	number	####################
func_name	string	########
file_name	string	###########
file_line_num	number	#############
application_name	string	###############
backend_type	string	#########
leader_pid	number	######################ID
query_id	number	########

19.8.6.

########################## ##############ps#####Windows#####Process Explorer###### ##### #######27.1#########

`cluster_name` (`string`)

> ### ############################### ##### ############################### #synchronous_standby_names####
>
> ###NAMEDATALEN##########64############# #####ASCII#####`cluster_name`### ######## ######C#####16################# ###''############################ ###### #######################

`update_process_title` (`boolean`)

> #######SQL################################ ###################on###### ########Windows############################off############## ########### #SET##########################

19.9.

19.9.1.

################################## ######################pg_stat#pg_statio## ################## ####27##########

`track_activities` (`boolean`)

> ## ################## #### ##############pg_read_all_stats## ############################## ##################### ###########SET####### ###################

`track_activity_query_size` (`integer`)

> pg_stat_activity.query## ########## ############################### #######1024###### ############ #########

`track_counts` (`boolean`)

> ############################ ## ###########SET##########################

`track_io_timing` (`boolean`)

> #########I/O############### ## ## ######################### #pg_test_timing########## I/O#######pg_stat_database#pg_stat_io##### ## BUFFERS###############EXPLAIN ####VERBOSE###############VACUUM# ###log_autovacuum_min_duration##########pg_stat_statements########## ########### #SET##########################

`track_wal_io_timing` (`boolean`)

> #########WAL I/O############### ## ## ############################ pg_test_timing########## I/O#######pg_stat_wal####### ###########SET##### #######################

track_functions (enum)

########################## #################pl######### SQL###C#########
###all######### ####################none### ###########SET##############
############

##

##########################SQL#######################

stats_fetch_consistency (enum)

################################## none##############################
cache##########################pg_stat_clear_snapshot()###########
############################## snapshot###############
##pg_stat_clear_snapshot()##
############ ### ######cache###

##

none################## ######################## cache##############
################ ##################### snapshot###################
##

19.9.2.

compute_query_id (enum)

####################### #########pg_stat_activity####EXPLAIN####
#log_line_prefix####################### pg_stat_statements################## #
########################
######off######on#######pg_stat_statements####################auto#####
##########################EXPLAIN##############auto#######regress### #
#####auto###

##

###
##################

log_statement_stats (boolean)
log_parser_stats (boolean)
log_planner_stats (boolean)
log_executor_stats (boolean)

####Unix#getrusage()##########
######################## log_statement_stats############################
####### log_statement_stats############################## ##############
######### ###########SET##########################

19.10. ##Vacuum##

########################## ###24.1.6######### ################################ ## ################

autovacuum (boolean)

####autovacuum######################## ########### ###autovacuum######## #track_counts############# ########postgresql.conf######################## ###### ############################autovacuum###################

################################ID##############autovacuum################## ### ###24.1.5#########

autovacuum_max_workers (integer)

############autovacuum#####autovacuum################# ######3### ######### ############

autovacuum_naptime (integer)

################autovacuum############### ############################## ######################VACUUM###ANALYZE########## ###################### ######## ######1##1min#### ########postgresql.conf######################## #######

autovacuum_vacuum_threshold (integer)

##########VACUUM################################### ######50###### ####### #postgresql.conf############################## ########################## ############################

autovacuum_vacuum_insert_threshold (integer)

#######VACUUM############################ ######1000###### -1######### VACUUM###########VACUUM################ ########postgresql.conf### ########################## ## ########

autovacuum_analyze_threshold (integer)

##########ANALYZE## ######50###### ## ######postgresql.conf############################## ################### ##############################

autovacuum_vacuum_scale_factor (floating point)

VACUUM################autovacuum_vacuum_threshold################### ## ######0.2########20%#### ########postgresql.conf###################### ##

autovacuum_vacuum_insert_scale_factor (floating point)

VACUUM#############autovacuum_vacuum_insert_threshold############## ####### ######0.2########20%#### ########postgresql.conf################# ##

autovacuum_analyze_scale_factor (floating point)

ANALYZE################autovacuum_analyze_threshold################# #### ######0.1########10%#### ########postgresql.conf################### ########## ##

autovacuum_freeze_max_age (integer)

########ID########VACUUM#################pg_class.relfrozenxid #########
########################## ##
###########

vacuum####pg_xact######################## #########2####################
##
###24.1.5##########

autovacuum_multixact_freeze_max_age (integer)

########ID########VACUUM#################pg_class.relminmxid ############
########################## ##
###########

#############ID#vacuum#pg_multixact#pg_multixact/offsets#############
######### #########4#################### ######################## ############
###################################### ###24.1.5.1##########

autovacuum_vacuum_cost_delay (floating point)

##VACUUM##################### -1########### vacuum_cost_delay########## ######
########################### #######2###### ########postgresql.conf#########
#################### ##

autovacuum_vacuum_cost_limit (integer)

##VACUUM##################### ########-1########### vacuum_cost_limit##########
####################
###################### ########postgresql.conf##########################
##

19.11.

19.11.1.

client_min_messages (enum)

################################
##DEBUG5#DEBUG4#DEBUG3#DEBUG2#DEBUG1#LOG#NOTICE#WARNING###ERROR###
########################## ############################## ######NOTICE### #
###LOG######log_min_messages##################

INFO########################

search_path (string)

###
###########################
##

search_path################################ ##################USAGE#####
#####################

##########################USAGE##################1##########$user##
##CURRENT_USER############################## ###############$user######
###

################pg_catalog########################## ####################
######## pg_catalog####################################

######################pg_temp_*nnn*################ ###pg_temp############
################ #################pg_catalog################ ###########
######################

##search_path##################
############ ####################

##############"$user", public### ################################
#public## ##########################
######################

################5.10######## ##
############

SQL###current_schemas ###########################9.27########## ##
##search_path ############## current_schemas##search_path###########
##############

row_security (boolean)

on###################### off##
################################ ######on### ################off############
####### #####pg_dump#######off####### ################################## #
#######BYPASSRLS###################

######################CREATE POLICY#######

default_table_access_method (string)

################################CREATE################################
############SELECT ... INTO################################## #####
#heap###

default_tablespace (string)

######CREATE##

###################
###PostgreSQL################################ ############################
#######CREATE############## #############

###################### #############temp_tablespaces#######

########################## ##
#####

##default_tablespace###########
###

#################22.6#########

default_toast_compression (enum)

##################TOAST########## ####CREATE TABLE####ALTER
TABLE#COMPRESSION############################ ############
##pglz#lz4#PostgreSQL#--with-lz4################ ######pglz###

temp_tablespaces (string)

######CREATE##
#################### ##

################## ################################PostgreSQL############### #### ### ############### #########PostgreSQL#####################################

temp_tablespaces#################################### ####CREATE####### ################## ################################### ####CREATE####### ########### ##########postgresql.conf####################

############## ##

default_tablespace########

check_function_bodies (boolean)

############### off#######CREATE FUNCTION#CREATE PROCEDURE########### ############### ##(false positive)### ######### ########################off##### pg_dump#############

default_transaction_isolation (enum)

SQL###############read uncommitted###read committed###repeatable read#### ##serializable################## ################################### ##### ##read committed####

#13# ### SET TRANSACTION

default_transaction_read_only (boolean)

#######SQL################################ ################################ ########## #####off#########

########SET TRANSACTION#########

default_transaction_deferrable (boolean)

##################################SQL################################ #### ### ################### ##

######off###

#####SET TRANSACTION##########

transaction_isolation (enum)

################################ ##################default_transaction_isolation## ########### ##################SET TRANSACTION######

transaction_read_only (boolean)

################################### ################## #default_transaction_read_only############# ##################SET TRANSACTION# #####

transaction_deferrable (boolean)

########################(deferrable status)####### ################## #default_transaction_deferrable############# ##################SET TRANSACTION# #####

session_replication_role (enum)

###origin(#####)#replica#####local### ##################################### ##############SET##########################

###replica############ ##ALTER TABLE##ENABLE TRIGGER#ENABLE RULE#######

PostgreSQL#origin#local################### ##############################local##################################2#############

#############################replica##

statement_timeout (integer)

###log_min_error_statement#ERROR###

#####[86.422018]###SQL#####################################SQL############ #13#####PostgreSQL##########SQL##Parse#Bind#Execute#Describe############Execute####Sync## #####SQL#######1######-## #PostgreSQL######13###Describe#################Sync###############

#####################postgresql.conf##statement_timeout#############

transaction_timeout (integer)

#####[51.842105]###BEGIN##

##############_##################idle_in_########_#####_##########

######_#########################

#####[76.223776]######################postgresql.conf#

#statement_timeout############## ##############.conf#PostgreSQL_##

##

##

lock_timeout (integer)

###LOCK TABLE####SELECT FOR UPDATE without NOWAIT###

statement_timeout################################ ####################
#####statement_timeout#######lock_timeout##########################
log_min_error_statement#ERROR##################################

lock_timeout#postgresql.conf#######################################

idle_in_transaction_session_timeout (integer)

###
################################ ############################

####################
################################vacuum########### #########################
########### ######24.1########

idle_session_timeout (integer)

###
#########################

###########
###########idle_in_transaction_session_timeout###########

#######
################################
##############################

vacuum_freeze_table_age (integer)

#####pg_class.relfrozenxid######################VACUUM################
###XID####MXID################
#VACUUM####### ######1.5########### ############20###################
##VACUUM################autovacuum####################VACUUM###########
##autovacuum_freeze_max_age#95%########### ###24.1.5##########

vacuum_freeze_min_age (integer)

VACUUM####XID### ####
##5############# ########0##10#######################VACUUM####
#autovacuum_freeze_max_age############### ########autovacuum################
###24.1.5##########

vacuum_failsafe_age (integer)

VACUUM##################ID############################
#pg_class.relfrozenxid################################# ###VACUUM###
########### ####################ID################################ #######
#VACUUM########################

###
###########################VACUUM#########################

######16########### ############21############
##VACUUM#autovacuum_freeze_max_age#105%######################

vacuum_multixact_freeze_table_age (integer)

pg_class.relminmxid######################VACUUM################### ####
################################XID####MXID#################VACUUM######
######1#5############ ####0##20###
#######VACUUM############VACUUM#autovacuum_multixact_freeze_max_age#95%####
####### ###24.1.5.1########

`vacuum_multixact_freeze_min_age` (integer)

VACUUM## ###### 500############### ####0##10###VACUUM#autovacuum_multixact_freeze_max_age################### ###24.1.5.1#### ####

`vacuum_multixact_failsafe_age` (integer)

VACUUM#################ID############################ #pg_class.relminmxid#################################### ###VACUUM### ########## ####################ID################################## ####### #VACUUM#########################

###

######16############### ############21############ ##VACUUM#autovacuum_multixact_freeze_max_age#105%#######################

`bytea_output` (enum)

bytea############## #####hex##########escape#PostgreSQL########## #### #8.4######### bytea##############################

`xmlbinary` (enum)

########XML############## ######xmlelement#xmlforest###bytea##XML## ########### ######base64#hex### ####XML############### ######base64### XML############9.15##########

##hex######base64################

`xmloption` (enum)

XML##########DOCUMENT####CONTENT########### #########8.13######### #### #DOCUMENT#CONTENT### ######CONTENT###

##SQL################################

```
SET XML OPTION { DOCUMENT | CONTENT };
```

#####PostgreSQL########

`gin_pending_list_limit` (integer)

fastupdate###########GIN############################ ################### ###########GIN######################### ################################## ######4#####(4MB)### #########GIN## ## ##########64.4.4.1#64.4.5#########

`createrole_self_grant` (string)

CREATEROLE####SUPERUSER## ################################## ###set#inherit######################### ####### ############################

###########CREATEROLE##SET ROLE#################### CREATEROLE#####################ADMIN OPTION##############################GRANT########## ####################

########### ######################################SET ROLE##### ########
####CREATEROLE##########################

event_triggers (boolean)

##
############################## ############################## ########### ##
###########SET######################

restrict_nonsystem_relation_kind (string)

################################ ######################### ################
####view#foreign-table###

19.11.2.

DateStyle (string)

################################## ################2################ ######
#ISO#Postgres#SQL#German###/#/##########DMY#MDY#YMD#### ##############
############## Euro###European#####DMY######
##US#NonEuro#NonEuropean#MDY###### ###8.5######### ##########ISO, MDY#
###initdb########lc_time############################

IntervalStyle (enum)

################sql_standard###SQL##################### #######
#postgres##DateStyle#####ISO##############8.4###PostgreSQL#############
#postgres_verbose##DateStyle######ISO################8.4###PostgreSQL#
############ #iso_8601##ISO 8601#4.4.3.2#############format with designators####
##########

##IntervalStyle############################## #######8.5.4#########

TimeZone (string)

######################### #############GMT######postgresql.conf#######
####initdb############################## ###8.5.3#########

timezone_abbreviations (string)

#################################### ######'Default'### ################ #
##Australia#India############################## ###B.4#########

extra_float_digits (integer)

########float4#float8##

####1## 8.1.3######### ###########
############## float8######17##float4#####9###### ######################
###2##################### #############3#############

####################10########### #########################FLT_DIG###
#DBL_DIG######################### ######-1####float4##5###float8###14###
####### ###########2##

##
################PostgreSQL 12######### ##########8.1.3########

client_encoding (string)

PostgreSQL###############
####23.3.1#########

lc_messages (string)

##23.1######### ##################
#######################################

##################
############# #########################

#######################SET############

lc_monetary (string)

################### ####to_char()########### ################# #######23.1#
######## ###

lc_numeric (string)

#################### ####to_char########### ################# #######23.1###
####### ###

lc_time (string)

###to_char############################### ################# #######23.1####
###

icu_validation_level (enum)

ICU## ####
##DISABLED#DEBUG5#DEBUG4#DEBUG3#DEBUG2#DEBUG1#INFO#NOTICE#WARNING#ERROR#
LOG###

DISABLED########################## ############################## #####
#WARNING###

default_text_search_config (string)

####12###########
#pg_catalog.simple####initdb##########################lc_ctype#######
#################

19.11.3.

####'$libdir/mylib'######mylib.so(#
###############mylib.sl)########################### ############################
#########

#####'$libdir/plXXX'#########PostgreSQL#########################
XXX#pgsql#perl#tcl#python###

PostgreSQL################################ ####PostgreSQL########magic block#####
################### ####PostgreSQL######################### LD_PRELOAD####OS#
######################

##

`local_preload_libraries` (`string`)

####################1#################### ##############################
#LOAD########### ############## ################################## #######
################## ############## ##############################

######################## ##
#plugins################ ##
local_preload_libraries#########$libdir/plugins/mylib##############
############################## mylib #$libdir/plugins/mylib#######

##################LOAD###
############## ##############PGOPTIONS###########ALTER ROLE SET#######
##########

###
#session_preload_libraries#######

`session_preload_libraries` (`string`)

####################1#################### ##############################
#LOAD########### ############## ################################## #######
################## ############## ############################## ##########
#############SET#############

###################LOAD## ####ALTER
ROLE SET##############################auto_explain######### ################
###########(##############################)##############################
###############

shared_preload_libraries###
#################### ####################################

`shared_preload_libraries` (`string`)

##############################
#LOAD########### ##################### ###################### #############
##################

##########postmaster########################## ############################
############## ## ##################
#####

#################### ##
################
############################## ########################### ###############
##################### session_preload_libraries#############

> ##
>
> Windows## ###############
> ################### ####postmaster##########################Windows#
> #######shared_preload_libraries######

`jit_provider` (`string`)

##########JIT##########(30.4.2##)###### ######llvmjit### ##################
##

##################JIT#################### ######PostgreSQL#########JIT#######
##############

19.11.4.

dynamic_library_path (string)

##############################CREATE FUNCTION#LOAD#####################
###

dynamic_library_path###################Windows################### ####
#######$libdir###################PostgreSQL#################$libdir#####
#####PostgreSQL############################
##pg_config --pkglibdir ########## #########

```
dynamic_library_path = '/usr/local/lib/postgresql:/home/my_project/lib:
$libdir'
```

Windows##########

```
dynamic_library_path = 'C:\tools\postgresql;H:\my_project\lib;$libdir'
```

##############'$libdir'### ##############################

###################SET###
############ ######################### ############postgresql.conf#######
#########

gin_fuzzy_search_limit (integer)

GIN############################## ###64.4.5#########

19.12.

deadlock_timeout (integer)

#################################### ######################################
######### ##
########
######################### ######1##1s######################### ##############
########## ##################################### ##########################
########################## ############SET##########################

log_lock_waits## #############
#####deadlock_timeout#################

max_locks_per_transaction (integer)

######################################max_locks_per_transaction#########
################# ## ##################
######################## ##
############ ################################## ######64#####################
##
############## ######################

##
####################

max_pred_locks_per_transaction (integer)

##max_pred_locks_per_transaction# ######################### ## ######### ################################## ### #################### ########################### ######64################## ## ## ######################

max_pred_locks_per_relation (integer)

0########## ##########max_pred_locks_per_transaction################## ######-2########### PostgreSQL########### ########postgresql.conf###########################

max_pred_locks_per_page (integer)

######2### ####### #postgresql.conf############################

19.13.

19.13.1. ###PostgreSQL#####

array_nulls (boolean)

##################NULL#NULL###################### ###########on##NULL#### ################ ####8.2##########PostgreSQL#######NULL########## ##NULL##NULL############################## ############################### #off##########

#####off######NULL############################

backslash_quote (enum)

##############\'################ ########SQL##########''#####PostgreSQL #####\'######## ##ASCII#\########## ######\'############################## ##############################SQL# ### #### #backslash_quote####on ### \' ####,off #########safe_encoding ######### ###ASCII#\######################## safe_encoding #########

################\###\######### ######################E'...'################ ###########

escape_string_warning (boolean)

################'...'############\#####standard_conforming_strings## ############## ######on###

##################SQL## ###############(E'...')############### ################################### ###

lo_compat_privileges (boolean)

9.0###PostgreSQL## #### #on##################################### ######off### ############SET##### ######################

##PostgreSQL 9.0############## ########

`quote_all_identifiers` (boolean)

#######SQL## ### `EXPLAIN`# ##############`pg_get_viewdef`################ `pg_dump` ### `pg_dumpall`#`--quote-all-identifiers`##############

`standard_conforming_strings` (boolean)

##SQL####################`'...'`############################ PostgreSQL 9.1## ######`on`##################`off`########## ################################ ################## ######################`E'...'`########################## ## ######### (4.1.2.2)##

`synchronize_seqscans` (boolean)

##################################I/O#################################### ########## ## ########## ######`ORDER BY`################################## ########`off`## ##################################8.3############ ######`on`###

19.13.2.

`transform_null_equals` (boolean)

######*`expr`* `= NULL`#####`NULL = `*`expr`*######*`expr`* `IS NULL`##############*`expr`*# NULL######################### *`expr`* `= NULL`####SQL#########NULL######## ##### #################`off`#######

####Microsoft Access#######NULL########*`expr`* `= NULL`#################### ## *`expr`* `= NULL`#######SQL## ###########NULL## ############### ####### #################NULL################################off###

########`= NULL`######################## ####################`IN`############## ############### ######################################

#######9.2#########

`allow_alter_system` (boolean)

#####[52.884615]#####SSL############################ ###### ##`postgresql.conf`########################## ######`on`### #### ##`allow_alter_`#################`ALTER`######################## ###### ##`PostgreSQL.conf`######################### #######`on`###

################################## ###`ALTER`################## ######### SQL################################ ##################################### ##################`PostgreSQL.auto.conf`#######

###################PostgreSQL################################## ############ ##########`ALTER`##################################### ################### ### ################################## #######

##############`ALTER`############### `PostgreSQL.auto.conf`####### ##`allow_alter_`#######################

19.14.

`exit_on_error` (`boolean`)

on########################## #########off########### FATAL#################
##

`restart_after_crash` (`boolean`)

#######on####PostgreSQL############################## ####################
#################### #### PostgreSQL#####################################
##

########`postgresql.conf`#########################

`data_sync_retry` (`boolean`)

########off#######PostgreSQL##################################PANIC######
######## ############################## #######################

###################
##WAL##############

on###########PostgreSQL## ########
#####################################on##########

`recovery_init_sync_method` (`enum`)

########`fsync`######PostgreSQL##
#########WAL###
##################WAL##
##pg_basebackup##

#####[86.369594]#Linux################################WAL################
###`syncfs`######
###########################`fsync`####################
###Linux######
5.8####################I/O####PostgreSQL##################################
####### #############`syncfs`######################WAL##################
###
##########1################`fsync`###################### ##################
#######
###5.8#############################I/O####PostgreSQL#######################
################

#######`postgresql.conf`############################

19.15.

################### ########`postgresql.conf`############## ################
##############################PostgreSQL################ ########PostgreSQL###
#######################

`block_size` (`integer`)

################## ########`BLCKSZ`##################8192###### #shared_buffers##
##############`block_size`########## ##########19.4##########

`data_checksums` (`boolean`)

################################### ###data checksums########

`data_directory_mode` (`integer`)

#########Unix##########################data_directory################
#Microsoft Windows###########0700######## #######group access########

`debug_assertions` (`boolean`)

PostgreSQL############################## ####USE_ASSERT_CHECKING####
PostgreSQL#################(####configure######--enable-cassert######
#)####### #######PostgreSQL##################

`huge_pages_status` (`enum`)

##############huge pages########:###########POSTGRES-C######## #######
##huge_###=####huge pages############################ ########huge_pages###
#######

`integer_datetimes` (`boolean`)

PostgreSQL#64################################### PostgreSQL 10########on###

`in_hot_standby` (`boolean`)

########################### ###on############################## ###########
######################### ###26.4########

`max_function_args` (`integer`)

############### ##########FUNC_MAX_ARGS########## ########100#####

`max_identifier_length` (`integer`)

############### #######NAMEDATALEN############### ######NAMEDATALEN###
64###########max_identifier_length#63#########################63#######
########

`max_index_keys` (`integer`)

##############################INDEX_MAX_KEYS##################32#####

`segment_size` (`integer`)

################################### #######RELSEG_SIZE######### ########
#############block_size##segment_size###############1GB###

`server_encoding` (`string`)

######################## ##############################client_encoding########
#########

`server_version` (`string`)

################# ########PG_VERSION############

`server_version_num` (`integer`)

#################### ###########PG_VERSION_NUM###########

`shared_memory_size` (`integer`)

################## ####################

`shared_memory_size_in_huge_pages` (integer)

#####huge_page_size################################ ##################-1#### ##

#####Linux############# #############-1######## Linux################18.4.5## ########

`ssl_library` (string)

##########SSL####################PostgreSQL############SSL########### #OpenSSL###############

`wal_block_size` (integer)

WAL################# #######XLOG_BLCKSZ##################8192######

`wal_segment_size` (integer)

######################## #######16MB### ###########28.5########

19.16.

##############################PostgreSQL############################ ########### ##############

###SQL############ ### #plpgsql.variable_conflict#######

###PostgreSQL############## ## ################ ##################################### ### ##########

19.17.

##############postgresql.conf########### ##############################

`allow_in_place_tablespaces` (boolean)

CREATE TABLESPACE##################pg_tblspc######################## ##### ## #### ## ########### #SET##########################

`allow_system_table_mods` (boolean)

############SET###################

`backtrace_functions` (string)

###############C######## #################C############################## ################### ###########################

###

#######################SET############

debug_discard_caches (integer)

1################################## ############################## ####### ################################## 1######################## ############### ######################## #######0########################

#inval.c#pg_config_manual.h##################

##############DISCARD_CACHES_ENABLED###########configure######--enable-cassert########################## ########0################### ########

debug_io_direct (string)

##########WAL########O_DIRECT#####Unix###### ###F_NOCACHE#macOS##FILE_FLAG_NO_BUFFERING #Windows################## ##############

####################I/O##################################I/O############### ### ######################data#WAL#######wal##########WAL####### #wal_init###

##################################I/O## ##########

###

debug_parallel_query (enum)

debug_parallel_query###### ##off############################on######################## ###regress#on########################

###########on##########################Gather########################### #### ### ## ############################## ######################################PARALLEL UNSAFE######PARALLEL RESTRICTED##################

######regress#####on#################################### ############### ##########################regress################################## ###### ######Gather#####EXPLAIN#######off#######################

ignore_system_indexes (boolean)

################################

post_auth_delay (integer)

############################## ## ## ############################## ########################### ############## ###################

pre_auth_delay (integer)

###########fork#################### ## ######################### ############################## ################## ######## #######postgresql.conf############################

trace_notify (boolean)

LISTEN#NOTIFY####################### ################################
##client_min_messages####log_min_messages#DEBUG1############

trace_sort (boolean)

##################################### ####### PostgreSQL########
##TRACE_SORT################### #########TRACE_SORT################

trace_locks (boolean)

########################## ##
############ ##
#################

```
LOG:  LockAcquire: new: lock(0xb7acd844) id(24688,24696,0,0,0,1)
      grantMask(0) req(0,0,0,0,0,0,0)=0 grant(0,0,0,0,0,0,0)=0
      wait(0) type(AccessShareLock)
LOG:  GrantLock: lock(0xb7acd844) id(24688,24696,0,0,0,1)
      grantMask(2) req(1,0,0,0,0,0,0)=1 grant(1,0,0,0,0,0,0)=1
      wait(0) type(AccessShareLock)
LOG:  UnGrantLock: updated: lock(0xb7acd844) id(24688,24696,0,0,0,1)
      grantMask(0) req(0,0,0,0,0,0,0)=0 grant(0,0,0,0,0,0,0)=0
      wait(0) type(AccessShareLock)
LOG:  CleanUpLock: deleting: lock(0xb7acd844) id(24688,24696,0,0,0,1)
      grantMask(0) req(0,0,0,0,0,0,0)=0 grant(0,0,0,0,0,0,0)=0
      wait(0) type(INVALID)
```

############src/include/storage/lock.h ######

########PostgreSQL########LOCK_DEBUG#################

trace_lwlocks (boolean)

############################ ##
#######

########PostgreSQL########LOCK_DEBUG#################

trace_userlocks (boolean)

############################## ###trace_locks##########################

########PostgreSQL########LOCK_DEBUG#################

trace_lock_oidmin (integer)

########OID####################### ############################

########PostgreSQL########LOCK_DEBUG#################

trace_lock_table (integer)

#######OID#################

########PostgreSQL########LOCK_DEBUG#################

debug_deadlocks (boolean)

##

########PostgreSQL########LOCK_DEBUG##################

log_btree_build_stats (boolean)

########B-tree##################CPU######################

########PostgreSQL########BTREE_BUILD_STATS##################

wal_consistency_checking (string)

#########WAL#REDO########################### #######WAL############### ######################### ### ################################## ###################################### ### ##############################

######################## ##################all########## ################## ############################## ######################## ##heap#heap2#btree#hash#gin#gist#sequence#spgist#brin#generic### ## ######################## ###########SET##########################

wal_debug (boolean)

##on#####WAL###########################WAL_DEBUG#### PostgreSQL######### #############

ignore_checksum_failure (boolean)

data checksums##############

#######################PostgreSQL##########################
ignore_checksum_failure####on#######################################
################
################################
#########off### ###########SET##########################

zero_damaged_pages (boolean)

################PostgreSQL############################
zero_damaged_pages#on##
################################
############### ## ##############
##########################on######### ####################################
######off### ###########SET########
##################

ignore_invalid_pages (boolean)

off###########################WAL#####PostgreSQL####PANIC##############
########### ignore_invalid_pages#on######WAL######################
###################### ##
#########PANIC#####################################
############################

jit_debugging_support (boolean)

LLVM####################GDB####### ################## #######off### ##
####################

jit_dump_bitcode (boolean)

#####LLVM IR#data_directory################ ###JIT##########################
#######off### ###########SET##########################

`jit_expressions` (boolean)

JIT############JIT################## #30.2#### ######on###

`jit_profiling_support` (boolean)

LLVM##############JIT#######perf######################### #####~/.debug/jit/############# ############################## ########off### ########## #############

`jit_tuple_deforming` (boolean)

JIT#####################JIT################## #30.2#### ######on###

`remove_temp_files_after_crash` (boolean)

########on#######PostgreSQL################################ ########## ######################### ################################### ####### #postgresql.conf############################

`send_abort_for_crash` (boolean)

######################postmaster#########SIGQUIT############### ########### ######################## ########on##############SIGABRT######## ########## ###################### ############################## ### ## ################ ########postgresql.conf#########################

`send_abort_for_kill` (boolean)

########SIGQUIT################postmaster#5#######SIGKILL############### ## ######on##########SIGKILL#####SIGABRT######## ########################## ##### ############################## ##################################### ##################################### ###################################### ##################################### ########postgresql.conf#########################

`debug_logical_replication_streaming` (enum)

########buffered###immediate### ######buffered### ######################## ## debug_logical_replication_streaming#########################

#####[86.776860]###########debug_logical_replication_streaming######## #################################### immediate#######CREATE SUBSCRIPTION## buffered#######logical_decoding_work_mem############################ ##### ####################_####_########_############################## ################### immediate## ############################## buffered################ _decoding_work_mem#########################

###########streaming######parallel######## ##debug_logical_replication_streaming############################## ############################## buffered################################## ########### immediate## ##############################

19.18.

19.4######### ######### ###############

#19.4

#######	##
-B *x*	shared_buffers = *x*
-d *x*	log_min_messages = DEBUG*x*
-e	datestyle = euro
-fb, -fh, -fi, -fm, -fn, -fo, -fs, -ft	enable_bitmapscan = off, enable_hashjoin = off, enable_indexscan = off, enable_mergejoin = off, enable_nestloop = off, enable_indexonlyscan = off, enable_seqscan = off, enable_tidscan = off
-F	fsync = off
-h *x*	listen_addresses = *x*
-i	listen_addresses = '*'
-k *x*	unix_socket_directories = *x*
-l	ssl = on
-N *x*	max_connections = *x*
-O	allow_system_table_mods = on
-p *x*	port = *x*
-P	ignore_system_indexes = on
-s	log_statement_stats = on
-S *x*	work_mem = *x*
-tpa, -tpl, -te	log_parser_stats = on, log_planner_stats = on, log_executor_stats = on
-W *x*	post_auth_delay = *x*

#20#

##############################Unix##############################PostgreSQL### ############# SQL ## #######21######### ## ##

> ##
>
> #21#############PostgreSQL###################### #######LOGIN########### ########################

###

PostgreSQL####################### ##

PostgreSQL## ## ##### ## ########### ########OS####################

20.1. pg_hba.conf####

#################################### pg_hba.conf#################### #HBA# ##host-based authentication: ############## ######pg_hba.conf################initdb### ################ ############################## hba_file################

pg_hba.conf##################SIGHUP################### ################### ###pg_ctl reload####SQL###pg_reload_conf()########kill -HUP#### ##postmaster##################################

> ##
>
> ##########Windows############ ####pg_hba.conf######################### ####

pg_hba.conf## ###pg_hba_file_rules######## ######error######NULL######################### #######

pg_hba.conf############1####1################## ######################### ## ## ######## ## ##################################### ## ################################all###replication######################### ################################## ################################

###IP################################## ###################### ### ### ########## ################### ### ### ########################

#######include####################### include################################### ### ######include################## include############2############## ##include#include_if_exists###include_dir############################

include_dir######.#####.conf## ##################### ############################(C######################### #########)########

#################

```
local               database  user  auth-method [auth-options]
host                database  user  address     auth-method  [auth-
options]
hostssl             database  user  address     auth-method  [auth-
options]
hostnossl           database  user  address     auth-method  [auth-
options]
hostgssenc          database  user  address     auth-method  [auth-
options]
hostnogssenc        database  user  address     auth-method  [auth-
options]
host                database  user  IP-address  IP-mask      auth-method
 [auth-options]
hostssl             database  user  IP-address  IP-mask      auth-method
 [auth-options]
hostnossl           database  user  IP-address  IP-mask      auth-method
 [auth-options]
hostgssenc          database  user  IP-address  IP-mask      auth-method
 [auth-options]
hostnogssenc        database  user  IP-address  IP-mask      auth-method
 [auth-options]
include             file
include_if_exists   file
include_dir         directory
```

#####################

local

#######Unix##################### #################Unix#####################

host

########TCP/IP############## host######SSL####SSL###GSSAPI#####GSSAPI##### ##########

> ##
>
> ##############################localhost##TCP/IP########## ######### #listen_addresses##################################TCP/IP#########

hostssl

###########SSL##############TCP/IP####################

#####################SSL######################### ### SSL#ssl############### ############################18.9########## ################################ #######hostssl###########

hostnossl

########hostssl########SSL########TCP/IP###########

hostgssenc

########TCP/IP##############GSSAPI######################

####################GSSAPI########################## ####################
###################hostgssenc###########

hostnogssenc

########hostgssenc#########GSSAPI###########TCP/IP############

database

######################## all ########################## sameuser#######
samerole####################
############################## ####samegroup#########samerole#########
##########################samerole#########samerole############## #############
######samerole############## replication################################
############## ########################### ################################
##################################### #############PostgreSQL###############
####### ################################

#############(/)############################## (PostgreSQL##############
##9.7.3.1##########)

################################@##############

user

########################### all######################### #################
#############(/)##################+################### #PostgreSQL##########
#################### +##
###+###########################
################### ################################ #######################
######

##########(/)############################ (PostgreSQL################9.7.3.1###
#######)

##############################@##############

address

############################ #############IP#######################1#######
##

IP##########################(/)#CIDR######################## CIDR#########
###IP################################ ####IP#################0##############
IP#####/####CIDR###################

####IPv4##################172.20.143.89/32###########172.20.143.0/24##
#########10.6.0.0/16######## IPv6################::1/128(#####IPv6#####
#####)###########fe80::7a31:c1ff:0000:0000/96######## 0.0.0.0/0####
IPv4#############::0/0####IPv6############ #############IPv4##32#IPv6##128#
############### ###############0#########

IPv4##############IPv4########IPv6#####################IPv6##IPv4###########
#IPv6###########

##IP############all########### ######IP#################samehost#######
##samenet###########

########IP###
IP############################DNS##############DNS############ #########
############## ##################################IP#######################
#####DNS########################## ######################################

#pg_hba.conf#####################IP########-###########1########### ### ################ 1##IP##################################IP################## ##########1##############

####.################################ ##
##.example.com##foo.example.com###### #example.com###########

#####pg_hba.conf######################## ######### nscd################## ########## ##########IP########################log_hostname# ############## #######

##########local############

> ##
>
> ###############IP###########2## ############### ###############DNS################################## #################### ##############################2##########1######1 ############### ###################################### ########### ########################pg_hba.conf############################## # ############################ ###########1############################ ##
>
> ### ################ ##################
>
> ########Apache HTTP####TCP############################## ########### #######

IP-address
IP-mask

##2########*IP-address/mask-length*############## ######################## ######### ###255.0.0.0#IPv4#CIDR####8#####255.255.255.255#CIDR####32### ######

##########local############

auth-method

################################ ########################20.3########## ##### ################################ldap###########################

trust

############ ######PostgreSQL#########################PostgreSQL######## ######################### ###20.4##########

reject

############ ############################ ####1##reject################ ######## ####################################

scram-sha-256

################SCRAM-SHA-256######### ###20.5########

md5

##################SCRAM-SHA-256####MD5######### ###20.5##########

`password`

############################ ### ################ ###20.5#########

`gss`

#######GSSAPI####### ###TCP/IP################## ###20.6########## GSSAPI# ###############

`sspi`

#######SSPI####### ###Windows############### ###20.7##########

`ident`

####################################ident################################# ###### ident####TCP/IP############################peer############## ### #20.8##########

`peer`

########################20.9##########

`ldap`

LDAP############## ###20.10##########

`radius`

RADIUS############## ###20.11##########

`cert`

SSL#################### ###20.12##########

`pam`

#####################PAM#Pluggable Authentication Modules############### ### #20.13##########

`bsd`

#####################BSD################ ###20.14##########

`auth-options`

`auth-method`######### ################*`name=value`*##################### ### ####################################

`##########################################clientcert####hostssl#########` `## ########verify-ca###verify-full#######  ##########################` `##SSL##########verify full######cn#Common Name##########################` `############ #####cert########20.12##############################hostssl###` `##############################`

`##############################cert########clientcert#############` `##clientname#########################################  ########2######1#####` `####### ########clientname=CN###############Common Name (CN)######## ##` `####clientname=DN###############Distinguished Name (DN)######### ###` `############################## RFC 2253[1]#DN######## ###############DN#########` `###########`

[1] https://datatracker.ietf.org/doc/html/rfc2253

```
openssl x509 -in myclient.crt -noout -subject -nameopt RFC2253 | sed
 "s/^subject=//"
```

###########DN##############################

include

#########################

include_if_exists

################################### #######################################

include_dir

######.######.conf##(C###
################################)########

@## ######pg_hba.conf######
####### ###@################ @##
#####

pg_hba.conf## ####################
########TCP###
#trust########TCP#################### ########################IP#########
################127.0.0.1########trust#####################

###
######################pg_hba.conf############################ #CONNECT############## ##############################pg_hba.conf### ##########CONNECT################

pg_hba.conf########### 20.1###### #######################

#20.1 pg_hba.conf#####

```
# ############################
# ##############Unix######################
# ################
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
local   all             all                                     trust


# ###################TCP/IP#########
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    all             all             127.0.0.1/32            trust


# ########################
#
# TYPE  DATABASE        USER            IP-ADDRESS      IP-MASK
  METHOD
host    all             all             127.0.0.1       255.255.255.255
  trust


# IPv6############
```

```
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    all             all             ::1/128                 trust


# #######################IPv4#IPv6############
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    all             all             localhost               trust


# DATABASE#######################db1234###db12######
# #db######2##4################################
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    "/^db\d{2,4}$"  all             localhost               trust


# IP####192.168.93.x##################
# ident#################################################
# #######postgres############
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    postgres        all             192.168.93.0/24         ident


# #####################
# ###192.168.12.10#####################postgres############
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    postgres        all             192.168.12.10/32        scram-
sha-256


# #######################
# example.com#####################################
#
# Require SCRAM authentication for most users, but make an exception
# for user 'mike', who uses an older client that doesn't support SCRAM
# authentication.
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    all             mike            .example.com            md5
host    all             all             .example.com            scram-
sha-256


# #####host###########3######
# 192.168.54.1##############################
# #######################GSSAPI######
# ##########IP###############
# ##################
# #########GSSAPI####hostgssenc########GSSAPI##
# ###########3###################192.168.12.10#######
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    all             all             192.168.54.1/32         reject
hostgssenc all          all             0.0.0.0/0               gss
host    all             all             192.168.12.10/32        gss


# 192.168.x.x###########ident########
# ########################ident##bryanh#####
# #bryanh##PostgreSQL#####guest1####
```

```
# ###########bryanh###guest1###############
# ####omicron##########pg_ident.conf##########
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    all             all             192.168.0.0/16          ident
 map=omicron


# #################4###################
# ########################################
# ########helpdesk#################support######
# ###############$PGDATA/admins################
# ###############
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
local   sameuser        all                                     md5
local   all             /^.*helpdesk$                           md5
local   all             @admins                                 md5
local   all             +support                                md5


# ######2##1##############
local   all             @admins,+support                        md5


# #########################
local   db1,db2,@demodbs  all                                   md5
```

20.2.

ident#GSSAPI###
########### ##############pg_hba.conf##map=*map-name*############## #######
############################## ##
##########################pg_hba.conf##*map-name*############

########ident##############################pg_ident.conf######################
##########################ident_file################## ident#####################
#########

map-name system-username database-username
include file
include_if_exists file
include_dir directory

###########pg_hba.conf######### *map-name*# pg_hba.conf############## ##2#
##*map-name*##1##
##########################

pg_hba.conf#######################include#################

pg_ident.conf#################### SIGHUP############### ##################
###############pg_ctl reload####SQL##pg_reload_conf()########kill -HUP#
#####postmaster#################

pg_ident.conf###
###pg_ident_file_mappings####### ######error######NULL##################
############

##############
###########
#all#*database-*

username#######*system-username*##
all##########################

database-username#+##+####
#####pg_hba.conf########## ######+###
####+######################### +################+#############

##*system-username*###########/########### ######################### #PostgreSQL
###############9.7.3.1########## ####################*database-username*#####
#\1#########-1################## ######1#######################################
#######

```
mymap   /^(.*)@mydomain\.com$      \1
mymap   /^(.*)@otherdomain\.com$   guest
```

##########@mydomain.com#########################@otherdomain.com#########
#######guest################# \1###*database-username*#########\1#########
#####

database-username##########(/)###########################(PostgreSQL#######
########9.7.3.1########)# *system-username*####################*database-*
username#########\1##############

> ###
>
> #################################### ##################################
> #^#$##########

pg_ident.conf######pg_hba.conf############### # 20.2## 20.1####### ####
##192.168#################### #################bryanh#ann#robert##########
########## Unix###robert#PostgreSQL######bob############# ######## robert####
############ ann#ann##############bryanh#bryanh######guest1### ##########
##

#20.2 pg_ident.conf######

```
# MAPNAME       SYSTEM-USERNAME         PG-USERNAME

omicron         bryanh                  bryanh
omicron         ann                     ann

# bob##########robert###############
omicron         robert                  bob

# bryanh#guest1###########
omicron         bryanh                  guest1
```

20.3.

PostgreSQL#####################

- trust#########################
- ################################
- GSSAPI####GSSAPI##########################Kerberos###Microsoft Active Directory ###########################

- SSPI####GSSAPI###Windows###############
- ident#################Identification Protocol# (RFC 1413[2])########### ######Unix##### ###########peer############
- peer## ####################
- LDAP####LDAP############
- RADIUS####RADIUS############
- #######SSL#############SSL####################
- PAM####PAM(Pluggable Authentication Modules)############
- BSD####BSD#############OpenBSD###############

peer####################trust############## ########################## ########## ############################SSL##

################################

20.4. Trust##

trust########PostgreSQL################## ################################## ########################### #####database#user############# ################ ##

trust######1#################################### ########################## ## ##########################Unix#######################trust################ ######### ######19.3##########unix_socket_permissions### #unix_socket_group######################### ### ##unix_socket_directories##

#########################Unix########## #####TCP/IP######################### ## ##########################pg_hba.conf## host ... 127.0.0.1 ...###### ##trust####################

TCP/IP######trust####trust#####pg_hba.conf################################ ##trust################## localhost#127.0.0.1######TCP/IP###trust################## ##

20.5.

###################### ## ########################

scram-sha-256

scram-sha-256####RFC 7677[3]########SCRAM-SHA-256######## ############## ##

###

md5

md5################################## ##################################### ##################################### ###MD5############################## ################

[2] https://datatracker.ietf.org/doc/html/rfc1413
[3] https://datatracker.ietf.org/doc/html/rfc7677

md5########SCRAM##############pg_hba.conf#md5#################### SCRAM#######################SCRAM################

password

password#################################### ################### ##### #####SSL###########password########## ######SSL#############SSL####### ############

PostgreSQL## ############### #pg_authid#################### **CREATE ROLE foo WITH LOGIN PASSWORD 'secret'**##########SQL####CREATE ROLE#ALTER ROLE########## ### ##psql#\password############ ################################NULL######### ################

########################password_encryption########## ######scram-sha-256### ###############scram-sha-256#password####### ########################### ### ##############md5######scram-sha-256############## ######md5####### ###md5#password######### ############################ ####PostgreSQL###### ############################ ############ ################################ #pg_authid#########

##############md5##scram-sha-256################################SCRAM## ################postgresql.conf#password_encryption = 'scram-sha-256'# ############################pg_hba.conf#####scram-sha-256#########

20.6. GSSAPI##

GSSAPI##RFC 2743[4]############################ PostgreSQL##GSSAPI############ ################### GSSAPI##GSSAPI###############(#########)####### ######### GSSAPI#######SSL###

GSSAPI######PostgreSQL#########################17##########

GSSAPI#Kerberos#########GSSAPI##*servicename/hostname@realm*############## ########### [##############2#########1###################1################ ####################Kerberos#KDC################] ########################## PostgreSQL################## ############################*keytab*########### # #keytab## ####realm############## Kerberos#########################

######*servicename*### #postgres####libpq#krbsrvname############## *hostname*##libpq################## ## realm################Kerberos##############realm###

###GSSAPI######### ###########PostgreSQL######################## pg_ident.conf############### ############# ####pgusername@realm####pgusername######## ##1############# ###username@realm#PostgreSQL#########################

PostgreSQL#############realm## ####### ########################realm## ### ########include_realm#0######### #####realm######(######realm#krb_realm#### #################)krb_realm################### ######pg_ident.conf######## #########################

[4] https://datatracker.ietf.org/doc/html/rfc2743

####keytab########krb_server_keyfile############### ####################keytab###### #############PostgreSQL######keytab################# #########PostgreSQL######## ## #18.1###########

keytab#####Kerberos####################Kerberos################ MIT Kerberos# kadmin##################

```
kadmin% addprinc -randkey postgres/server.my.domain.org
kadmin% ktadd -k krb5.keytab postgres/server.my.domain.org
```

##########GSSAPI###############

include_realm

0############################realm#############20.2######################### krb_realm##################realm## include_realm######(1)##############PostgreSQL###########pg_ident.conf# #######################

map

#################################### ###20.2######### GSSAPI/Kerberos##### #username@EXAMPLE.COM###################username/ hostbased@EXAMPLE.COM########include_realm#0######################## ##username@EXAMPLE.COM#####username/hostbased@EXAMPLE.COM#### 0#### #########username#####username/hostbased###################

krb_realm

realm######################## ######################realm################ #### #################realm##

pg_hba.conf############################krb_caseins_users############ ## krb_realm############################ ##

20.7. SSPI##

SSPI######################Windows###### PostgreSQL##negotiate#######SSPI#### ### ##########Kerberos#################NTLM################ SSPI#GSSAPI#### ######################SSPI#######GSSAPI############## Windows############SSPI ##Windows#############GSSAPI###########

Kerberos############ SSPI##GSSAPI############ ###20.6##########

##########SSPI###############

include_realm

0############################realm#############20.2######################### krb_realm##################realm## include_realm######(1)##############PostgreSQL###########pg_ident.conf# #######################

compat_realm

1#############NetBIOS################SAM####include_realm############## ## ############## 0##############################realm#########

###SSPI################# ## ##############

`upn_username`

`compat_realm`##########################UPN############## ##############
##SAM############## ############2#############################

#################libpq#SAM#################### libpq####libpq################
##

`map`

############################# ###20.2######### SSPI/Kerberos#####
`#username@EXAMPLE.COM`##################`username/hostbased@EXAMPLE.COM`########`include_realm#0`#######################
`##username@EXAMPLE.COM#####username/hostbased@EXAMPLE.COM#### 0`####
########`username#####username/hostbased`##################

`krb_realm`

realm## realm#####################
########### #####realm##

20.8. Ident##

ident##############################ident###
################ ###TCP/IP############

> ##
>
> ident##TCP/IP##################### peer###20.9#####################

#########`ident`##############

`map`

############################## ###20.2#########

######Identification###########RFC 1413[5]######### ######Unix####################
#####TCP###113#####ident########### ident#####################*X*################*Y*#
############################## PostgreSQL###############*X*#*Y*####################
#####ident###

#################################### ##
113## ############################
#######ident
########################## ##############

##

—RFC 1413

#############
####PostgreSQL####################### ###PostgreSQL###########################
##############

20.9. Peer##

[5] https://datatracker.ietf.org/doc/html/rfc1413

peer## ##################################
##################################

#########peer##############

map

#################################20.2#########

peer################getpeereid()###SO_PEERCRED###########################
#################Linux#macOS###BSD#####Solaris########

20.10. LDAP##

#######password###############LDAP############ LDAP#######################
######## #####LDAP##

LDAP###2###########1################################ ####*prefix username suffix*################## #####*prefix*######Active Directory####cn=#*DOMAIN*\#####
######### *suffix*##Active Directory#########DN####################

#####[90.965456]#2###########search+bind#################
#*ldapbinddn*#*ldapbindpasswd*###### ###################LDAP##############
################################## ##
###*ldapbasedn*###########*ldapsearchattribute*#######################
##
######################## ######Apache mod_authnz_ldap###pam_ldap###########
######LDAP################### ## #
###LDAP#####2###############

#######################

ldapserver

####LDAP##########IP##########################

ldapport

LDAP################################## LDAP########################

ldapscheme

ldaps######LDAPS####### ########LDAP#################SSL###LDAP########
#########ldaptls##############

ldaptls

1#######PostgreSQL#LDAP########TLS############ ###RFC 4513[6]#StartTLS#####
#########ldapscheme##############

ldapscheme#ldaptls#######PostgreSQL####LDAP##############################
SSL###############PostgreSQL####PostgreSQL####################

############################

ldapprefix

###############DN####################

[6] https://datatracker.ietf.org/doc/html/rfc4513

ldapsuffix

##############DN##################

#########search+bind############

ldapbasedn

##DN

ldapbinddn

####################################DN

ldapbindpasswd

##

ldapsearchattribute

################################## #############uid#######

ldapsearchfilter

search+bind################### $username################ ####
#ldapsearchattribute###################

ldapurl

RFC 4516[7]LDAP URL# ##########LDAP############################## #########
##########

ldap[s]://*host*[:*port*]/*basedn*[?[*attribute*][?[*scope*][?[*filter*]]]]

scope#base#one#sub######################## ######base##############
############# *attribute*###################ldapsearchattribute######
##*attribute*######ldapsearchfilter#####*filter*#######

URL####ldaps##ldapscheme=ldaps#########SSL##LDAP#####LDAPS########
StartTLS##########LDAP############URL####ldap#####ldapurl##
#ldaptls##################

#######non-anonymous bind#####ldapbinddn###ldapbindpasswd##############
##########

LDAP URL####OpenLDAP############Windows#############

search+bind#####################################

search+bind###########ldapsearchattribute##################
#ldapsearchfilter##############################
ldapsearchattribute=foo#####ldapsearchfilter="(foo=$username)"######
###############ldapsearchattribute=uid########

PostgreSQL##LDAP#############OpenLDAP###################ldapserver######
###################RFC 2782[8] DNS SRV############## _ldap._tcp.DOMAIN#
#########ldapbasedn##DOMAIN#######

[7] https://datatracker.ietf.org/doc/html/rfc4516
[8] https://datatracker.ietf.org/doc/html/rfc2782

#########LDAP##########

```
host ... ldap ldapserver=ldap.example.net ldapprefix="cn=" ldapsuffix=",
 dc=example, dc=net"
```

##########someuser#####################PostgreSQL#DN cn=someuser, dc=example, dc=net#######################LDAP############# ############# ################

###search+bind#######

```
host ... ldap ldapserver=ldap.example.net ldapbasedn="dc=example, dc=net"
 ldapsearchattribute=uid
```

#####[92.181070]##########someuser################PostgreSQL##ldapbinddn##### ##########################DN###(uid=someuser)############################ ##

URL########search+bind#######

```
host ... ldap ldapurl="ldap://ldap.example.net/dc=example,dc=net?uid?sub"
```

LDAP###########################URL############ ###################

ldapsearchattribute#####ldapsearchfilter########ID#################### #search+bind#######

```
host ... ldap ldapserver=ldap.example.net ldapbasedn="dc=example, dc=net"
 ldapsearchfilter="(|(uid=$username)(mail=$username))"
```

DNS SRV###########example.net#LDAP####################search+bind#######

```
host ... ldap ldapbasedn="dc=example,dc=net"
```

> ###
>
> LDAP#DN################################ ##########LDAP############## ################################

20.11. RADIUS##

########RADIUS#####################password######### RADIUS#####/######### ############## #######RADIUS####################################

RADIUS###############RADIUS######################## ########Authenticate Only#########################NAS Identifier######## ##################### ####### RADIUS###############Access Accept####Access Reject###### RADIUS ################

###RADIUS############################## ##################### ########### ################## ## ################ RADIUS## ######################## ###################

RADIUS#######################

`radiusservers`

####RADIUS####DNS######IP##### #############

`radiussecrets`

RADIUS######################### ###PostgreSQL#RADIUS#################### ## #####16################ #############

> ##
>
> ###############PostgreSQL#OpenSSL############################## ###### RADIUS##

`radiusports`

####RADIUS######### #########################RADIUS######1812########

`radiusidentifiers`

RADIUS#######NAS Identifier########### ################################ ###################### ###RADIUS######################### ################### #####postgresql########

RADIUS###2################# ####### RADIUS#########################

```
host ... radius radiusservers="server1,server2" radiussecrets="""secret
 one"","secret two"""
```

20.12.

##############SSL################ #########SSL########SSL###########18.9.2##### ##### ## ###################### #### ####cn#Common Name########################## ######################## ## #########cn#####################################

##########SSL####################

`map`

############################## ###20.2##########

`cert###clientcert###############  cert######clientcert=verify-full## #trust#########`

20.13. PAM##

##############PAM#Pluggable Authentication Modules###########password######### # #####PAM######postgresql### PAM#####/##########################IP########### ############ PAM########Linux-PAM###[9]#########

##########PAM###############

[9] https://www.kernel.org/pub/linux/libs/pam/

pamservice

PAM######

pam_use_hostname

PAM_RHOST#######PAM##################IP################ #######IP#######
####################1########## ############################PAM####
#########PAM##

> ##
>
> PAM#/etc/shadow##################PostgreSQL#########################
> #### ###PAM#LDAP#####################################

20.14. BSD##

####################BSD###########password########### BSD#######/########
########## #############BSD################################## BSD##########
#OpenBSD#########

PostgreSQL##BSD####auth-postgresql#########postgresql#######
#login.conf######################### ######################PostgreSQL#######
############

> ##
>
> BSD##########PostgreSQL#######################################auth####
> ################ auth#####OpenBSD##################

20.15.

##

```
FATAL:  no pg_hba.conf entry for host "123.123.123.123", user "andym",
 database "testdb"
```

#################################### ########################### ###
#pg_hba.conf##############################

```
FATAL:  password authentication failed for user "andym"
```

####################################pg_hba.conf############################
####################Kerberos#ident##################Kerberos####ident#########
######

```
FATAL:  user "andym" does not exist
```

##########################

```
FATAL:  database "testdb" does not exist
```

######################## ##

###

################################## ##################################

#21#

PostgreSQL################################ ################################## ################### ###################################### ################## ### ######################### #### ##

##################################### PostgreSQL#####8.1######################### ########### ################### ##

########################## ####################################5.8##########

21.1.

###########CREATE ROLE SQL#########

```
CREATE ROLE name;
```

name#SQL########### ##################################### ################## #LOGIN####################### ########### ######################DROP ROLE##### ######

```
DROP ROLE name;
```

###########SQL################################createuser######dropuser########## #####

```
createuser name
dropuser name
```

#####################pg_roles################

```
SELECT rolname FROM pg_roles;
```

################################

```
SELECT rolname FROM pg_roles WHERE rolcanlogin;
```

###psql######\du#########################

###1###### ################## ################initdb## ######## #postgres######## ################################

####psql########-U##### ######################### ###########createuser###psql##################### ################

#20### ####### ### ## #######################ID##

21.2.

##

######

LOGIN################################## LOGIN ##########################
########## ###########################

CREATE ROLE *name* LOGIN;
CREATE USER *name*;

#CREATE USER###### LOGIN####CREATE ROLE############CREATE USER#CREATE ROLE######

########

######
######################## ######################CREATE ROLE *name* SUPERUSER########## ##########################

#######

############
###CREATE ROLE *name* CREATEDB#########

######

#
#############CREATE ROLE *name* CREATEROLE#######
CREATEROLE#########ADMIN#######CREATEROLE#######################
####################CREATEROLE######################CREATEROLE###
########################## ##########################ALTER ROLE########
############## ###COMMENT#SECURITY LABEL#######################

####CREATEROLE###SUPERUSER#####################SUPERUSER############
####CREATEROLE###REPLICATION###########REPLICATION#############
############################## ####REPLICATION#####ALTER ROLE ... SET#
##ALTER ROLE ... RENAME######COMMENT ON ROLE#SECURITY LABEL ON ROLE#DROP ROLE########## ####CREATEROLE###BYPASSRLS##################
####

############

#################
############## ######################LOGIN############### ############
##CREATE ROLE *name* REPLICATION LOGIN#########

#####

password#md5#
################ ##################################### #######CREATE ROLE *name* PASSWORD '*string*'###############

#####

############################## ############################CREATE ROLE *name* NOINHERIT###### #######WITH INHERIT TRUE#####WITH INHERIT FALSE##########################

##############

##(RLS)#########################
#######################CREATE ROLE *name* BYPASSRLS##########

####

############################ -1(#####)########### CREATE ROLE *name* CONNECTION LIMIT '*integer*'##################

##################ALTER ROLE########### ###CREATE ROLE#ALTER ROLE#######
###########

######19## ############################
##############:#################

```
ALTER ROLE myname SET enable_indexscan TO off;
```

################################ ##############################SET enable_indexscan TO off;################ ################################
######### ####################ALTER ROLE *rolename* RESET *varname*#########
################LOGIN##

#####[95.462280]##########CREATEROLE##
###########################GRANT created_user TO creating_user WITH ADMIN OPTION, SET FALSE, INHERIT FALSE############
CREATEROLE####################################ADMIN OPTION###############
####CREATEROLE######################### ###INHERIT FALSE, SET FALSE#####
######CREATEROLE##############################SET ROLE###################
###########ADMIN OPTION##################################CREATEROLE######
##########INHERIT###SET################################## ##################
#############SET ROLE#################################### ################
#################CREATEROLE##
##############CREATEROLE############################## ##CREATEROLE#######
##########ADMIN OPTION#########################

21.3.

####################################### ##############################
PostgreSQL########################## ################################

##########################

```
CREATE ROLE name;
```

###################LOGIN######### ######################

#################GRANT###REVOKE###########################

```
GRANT group_role TO role1, ... ;
REVOKE group_role FROM role1, ... ;
```

########################### ################################### ##############
######################## #############PUBLIC#############

############2################## 1###########SET##################SET########
########################### ##

2######INHERIT#########
##
###############

```
CREATE ROLE joe LOGIN;
CREATE ROLE admin;
CREATE ROLE wheel;
CREATE ROLE island;
GRANT admin TO joe WITH INHERIT TRUE;
GRANT wheel TO admin WITH INHERIT FALSE;
GRANT island TO joe WITH INHERIT TRUE, SET FALSE;
```

###joe################joe##############admin#island#######################
#######   ####wheel###################joe#####wheel###################
#admin########WITH INHERIT FALSE##############

```
SET ROLE admin;
```

############admin##############joe#island###############

```
SET ROLE wheel;
```

###########wheel#####################joe#admin################### #########
################

```
SET ROLE joe;
SET ROLE NONE;
RESET ROLE;
```

##

#####################SET TRUE##############SET ROLE#################
############################## ############admin#####wheel###########
###island########### joe#######################

##

##SQL## PostgreSQL####
##########SQL##############INHERIT######SQL#############
#NOINHERIT######## ####8.1####################PostgreSQL##############
#INHERIT######## ######################################

LOGIN#SUPERUSER#CREATEDB####CREATEROLE################################
##################### #######1#####################################SET ROLE#
######### ########admin####CREATEDB###CREATEROLE################## ####
##joe################################## SET ROLE admin################

##############DROP ROLE#########

DROP ROLE *name*;

########################### #######################

21.4.

##DROP ROLE#####
######################## ##
################################

###########ALTER#########1##1############## ##########

```
ALTER TABLE bobs_table OWNER TO alice;
```

######REASSIGN OWNED###
REASSIGN OWNED###
########### #########REASSIGN OWNED#######################################
##

###DROP OWNED command######
########### ##
############## ###DROP OWNED###
###

DROP OWNED## REASSIGN
OWNED###REASSIGN
OWNED#DROP OWNED#######################

##

```
REASSIGN OWNED BY doomed_role TO successor_role;
DROP OWNED BY doomed_role;

-- ############################
DROP ROLE doomed_role;
```

###

####################DROP ROLE###
##

21.5.

PostgreSQL###
(CREATEROLE###########)##############################GRANT(##)#############
#####################

############# 21.1####### ##
################################

#21.1

###	#######
pg_read_all_data	##############SELECT############################## #######USAGE####################(##############)## #### ############BYPASSRLS########### RLS########## #############GRANT##########BYPASSRLS########### ######
pg_write_all_data	##############INSERT#UPDATE###DELETE########### #########################USAGE##################

###	#######
	###(##############)###### ###########BYPASSRLS### ######## RLS#######################GRANT######### #BYPASSRLS################
pg_read_all_settings	##############################
pg_read_all_stats	#####################pg_stat_*##################### ########
pg_stat_scan_tables	############ACCESS SHARE#####################
pg_monitor	###############/##### ##### ##pg_read_all_settings#pg_read_all_stats## #pg_stat_scan_tables#######
pg_database_owner	##################################
pg_signal_backend	#####################################
pg_read_server_files	COPY## ###########
pg_write_server_files	COPY## ##########
pg_execute_server_program	COPY## #####################
pg_checkpoint	CHECKPOINT###########
pg_maintain	Allow executing VACUUM, ANALYZE, CLUSTER, REFRESH MATERIALIZED VIEW, REINDEX, and LOCK TABLE on all relations, as if having MAINTAIN rights on those objects, even without having it explicitly.
pg_use_reserved_connections	reserved_connections#####################
pg_create_subscription	CREATE SUBSCRIPTION##################### #CREATE#######

pg_monitor#pg_read_all_settings#pg_read_all_stats## #pg_stat_scan_tables## #### ##

pg_database_owner##################1########################## ########### ################################## ######pg_database_owner################ ### pg_database_owner############## ################### ##########public## #####

pg_signal_backend## ############# ## ##### ## 9.28.2##########

pg_read_server_files#pg_write_server_files#pg_execute_server_program## ## ########################### ## ## ####

###

####GRANT####################################

```
GRANT pg_signal_backend TO admin_user;
```

21.6.

search_path###

##################################PostgreSQL##################

#22#

######PostgreSQL################1################# #############SQL############# ################################ ##

22.1.

##pg_global################ ##### ### ######################## ######################## ### ##

1######################## ######## ## ######## ###########2#############################20.1#########################5.8###### ## ########postgres_fdw#######1### ##### ##dblink######dblink################ ###

1##PostgreSQL## ##################################### ### ## ##PostgreSQL################ ############################## ### ## ############ ###5.10##########

1## ######WAL##28## ################################## ## #################

#######CREATE DATABASE#####22.2#########DROP DATABASE#####22.5########### # ########################pg_database##################

```
SELECT datname FROM pg_database;
```

###psql######\l#######-l#####################################

> ##
>
> ##SQL################catalog###########################

22.2.

#############PostgreSQL##################18.3###########

CREATE DATABASESQL###################

```
CREATE DATABASE name;
```

####name#SQL############### ############################## ######################## ##

#################### ############21.2##########

CREATE DATABASE################################### ######################
#################### #########initdb########################### #18.2#####
########postgres####### ########################postgres#########

template1#template0###2################################ ################
########template1####### ###template1####################################
######### ################################template1######################
template0#template1##################### ################################
####template1########### #######22.3#########

############createdb#####################

```
createdb dbname
```

createdb########## ###postgres##################CREATE DATABASE########
createdb######################### #####createdb###########################
####

##
##############################20###########

#############
########### SQL#####

```
CREATE DATABASE dbname OWNER rolename;
```

######

```
createdb -O rolename dbname
```

###

22.3.

###CREATE DATABASE######################## #######template1##############
########### ## template1#########
##################################
######## ####PL/Perl######template1###
#################

####CREATE DATABASE#########################GRANT########## ##############
###################

template0########################### #########template1################
################PostgreSQL######################### template0############
################ CREATE DATABASE#template1####template0##############
##template1###
################## ######pg_dump################## #############template1###
###

template1#####template0##################template0#######################
######## ###template1############################## ###template0##########
##template1##

template0###################

```
CREATE DATABASE dbname TEMPLATE template0;
```

#SQL############

```
createdb -T template0 dbname
```

###########

########################### #########CREATE DATABASE#######################
################### ###################COPY DATABASE#######################
CREATE DATABASE######
################# ############################

datistemplate##datallowconn#####################pg_database######
datistemplate##########CREATE DATABASE################################
########## #############CREATEDB########################### ###############
########################### datallowconn#################################
######################### template0##################datallowconn = false##
template0#template1######datistemplate = true##############

> ##
>
> template1#template0###template1#####CREATE DATABASE######
> ############################## ####template1#####template0#########
> ####### ############template1################## #template1######
> ##pg_database.datistemplate = false############
>
> ##################postgres############# ##############################
> ########## ###template1############################

22.4.

#19#########PostgreSQL#################### ################################
###

#######################GEQO##
##############SET geqo TO off;#################### ####################
################

```
ALTER DATABASE mydb SET geqo TO off;
```

############################### ################################SET geqo TO
off;################ ## ##########
#####ALTER DATABASE dbname RESET varname######

22.5.

##############DROP DATABASE##########

```
DROP DATABASE name;
```

################################## ###################################### ####
###############

##########################DROP DATABASE############## ####template1#########
############### ##############################template1##################

##################dropdb##############

```
dropdb dbname
```

```
#createdb#########################################
```

22.6.

PostgreSQL## ####
##

###################PostgreSQL########################### #########2######### 1####
##
##################

##1## ##################
##################################
##

> ##
>
> #PostgreSQL##
> ################ ##
> #################### ##
> ##################### #######RAM#######################################
> #####

############CREATE TABLESPACE############## ##########

```
CREATE TABLESPACE fastspace LOCATION '/ssd1/postgresql/data';
```

##############################PostgreSQL############################## #########
####################################
####################################

> ##
>
> ##1################################
> ####### ####PostgreSQL############### ########################### #########
> #####################

#################################### ##
#####################CREATE##########

#################CREATE#############
############################## #######space1####################

```
CREATE TABLE foo(i int) TABLESPACE space1;
```

##############default_tablespace##############

```
SET default_tablespace = space1;
CREATE TABLE foo(i int);
```

default_tablespace#####################################CREATE TABLE#### #CREATE INDEX######## TABLESPACE##########

temp_tablespaces############## ## ########## ###########1################################## ################## ############################## ######################################

####TABLESPACE######## #######default_tablespace#temp_tablespaces############################## ############################### ##

#####[93.939394]###################2################### pg_global############ ############ pg_default######template1#template0################### ##### ##CREATE DATABASE#TABLESPACE## ######

##

################DROP TABLESPACE##############

#########################pg_tablespace#########

```
SELECT spcname FROM pg_tablespace;
```

###psql######\db############################

$PGDATA/pg_tblspc##############################1####################### # ## 2####### ################## ####### PostgreSQL 9.1##########pg_tablespace################## ######## ##pg_dump####################

#23#

PostgreSQL###2####################

- ## ###23.1#23.2#### ######
- ## ###23.3## ########

23.1.

PostgreSQL############## ###########ISO C#POSIX##############

23.1.1.

##########initdb################################ initdb#################### ##################### ## #### ####################################initdb#--locale################## ###### #########

```
initdb --locale=sv_SE
```

Unix#######################SE################sv######### ###en_US###### #fr_CA#################### ################### ##*language_territory.codeset*############### ####fr_BE.UTF-8######BE##### #########fr##UTF-8###########

```
################################################################################
###   #####Unix######locale       -a##########################   Windows
##German_Germany#Swedish_Sweden.1252############################
```

######################################

LC_COLLATE	########
LC_CTYPE	##############################
LC_MESSAGES	########
LC_MONETARY	####
LC_NUMERIC	#####
LC_TIME	#######

```
########################################initdb#################### ########
#######################################initdb     --locale=fr_CA     --lc-
monetary=en_US#####
```

```
#######################################C########POSIX##########
```

```
#################################################  ###########################
#####################################  LC_COLLATE#LC_CTYPE#################
#########################################  ##############################  ##
##23.2#####################################  initdb###########################
##CREATE DATABASE##########################################
```

####19.11.2#######
initdb#######################################postgresql.conf#############
######postgresql.conf############################

############################
######### ##

##
##LC_ALL#LC_COLLATE######### ###########LANG# ################################C########
##LANGUAGE#### ######### ##gettext####### ######

##############################NLS##########configure --enable-nls###### ## ########################

23.1.2.

###########SQL###########

- #########ORDER BY############################
- upper#lower#initcap##
- #LIKE#SIMILAR TO#POSIX######################### ########################## ##################
- ###to_char##
- LIKE#################

C#POSIX####PostgreSQL#################### ##################LIKE############ ######################################

C###########PostgreSQL#LIKE## # ## ###11.10######## ## #########23.2###########C#######################

23.1.3.

######################### #######initdb#################################### ## ########################### ##

1. #####[84.179104]## ### ############## ##################/############PostgreSQL################## ##### ## # ############ ##################/#####################PostgreSQL############ ####
2. ######initdb## ############ ################################
3. ######################## SQL####CREATE DATABASE##############createdb### ############### ##

4. ############################ #########SQL############# #########23.2#######
   ### ##

5. ########################### #####SQL############### ########################
   ####################

23.1.4.

##

###ICU#####
##########################


```
initdb --locale-provider=icu --icu-locale=en
```

############################# ############################ ###############
#libc#######icu################1###
########

###lc_messages##
###

############################

builtin

  ############################ ##########C###C.UTF-8#################

  ######C#########libc######C######### ######################################
  ######

  ######C.UTF-8####################UTF-8#######Unicode################## ##
  ################### ##########"POSIX Compatible"#######################"###
  #"#######

icu

  ######icu#########ICU########### PostgreSQL####################

  ######ICU### ######
  ############################## LC_COLLATE###LC_CTYPE##ICU##############
  ##

> ##
>
> ######ICU##ICU########
> ###############

libc

  ######libc##################C########### #############
  ##LC_COLLATE#LC_CTYPE############################

> ##
>
> #################libc#######################################

23.1.5. ICU####

23.1.5.1. ICU#####

######ICU##########

```
CREATE COLLATION mycollation1 (provider = icu, locale = 'ja-JP');
CREATE COLLATION mycollation2 (provider = icu, locale = 'fr');
```

23.1.5.2.

ICU################ICU###
("###") ##### ####

```
CREATE COLLATION mycollation3 (provider = icu, locale = 'en-US-u-kn-
true');
NOTICE:  using standard form "en-US-u-kn" for locale "en-US-u-kn-true"
CREATE COLLATION mycollation4 (provider = icu, locale = 'de_DE.utf8');
NOTICE:  using standard form "de-DE" for locale "de_DE.utf8"
```

##############provider#locale################### ICU#########################
#############################

###################root####und("undefined")#############

ICU########libc#######################ICU################## ICU#libc##########
##libc#####################

##########################ICU################################

```
CREATE COLLATION nonsense (provider = icu, locale = 'nonsense');
WARNING:  ICU locale "nonsense" has unknown language "nonsense"
HINT:  To disable ICU locale validation, set parameter
 icu_validation_level to DISABLED.
CREATE COLLATION
```

icu_validation_level######################## ERROR##############################
##################

23.1.5.3.

BCP 47###

##########*language-region*#####*language*### *language*########:fr############
##*region*########:ca############ #:ja-JP#de####fr-CA#

ICU#################################
###

#######################################-u#########1####-*key-value*########
key############*value*############ #########-*key*############-*value*##########
#####true#########

########en-US-u-kn-ks-level2##########################kn#true######ks###
#2########## ##

```
CREATE COLLATION mycollation5 (provider = icu, deterministic = false,
 locale = 'en-US-u-kn-ks-level2');
SELECT 'aB' = 'Ab' COLLATE mycollation5 as result;
```

```
 result
--------
 t
(1 row)

SELECT 'N-45' < 'N-123' COLLATE mycollation5 as result;
 result
--------
 t
(1 row)
```

##23.2.3##########

23.1.6.

########
##locale -a################

PostgreSQL################################## LC_COLLATE#LC_CTYPE############
###################################### LC_MESSAGES#LC_MONETARY#############
###################################### SHOW###########################

#######src/test/locale########PostgreSQL#######################

###
##

####################PostgreSQL## ##
#################55################
###############

23.2.

################
#LC_COLLATE#LC_CTYPE##################

23.2.1.

################################## ###############text#varchar# char### ######
############################
######## ## ##########################
#############

#########default## ####################
######################################

########ORDER BY## <
######## ORDER BY########################## ################################
########### ##############lower#upper#initcap############################
###to_char##############

####################
####################################
###

######################## ##
############COLLATE############### ######################################
##

1. ### ####################
   ##

2. ## ## ##############################
3. ## ## ##############################

###################

```
CREATE TABLE test1 (
    a text COLLATE "de_DE",
    b text COLLATE "es_ES",
    ...
);
```

####

```
SELECT a < 'foo' FROM test1;
```

< ####de_DE############ ######################################

```
SELECT a < ('foo' COLLATE "fr_FR") FROM test1;
```

######################################fr_FR######### #########

```
SELECT a < b FROM test1;
```

############################a#b#################### < ## ## ###############

```
SELECT a < b COLLATE "de_DE" FROM test1;
```

#############

```
SELECT a COLLATE "de_DE" < b FROM test1;
```

######################

```
SELECT a || b FROM test1;
```

################## || ## ##

##

```
SELECT * FROM test1 ORDER BY a || 'foo';
```

###de_DE#######################

```
SELECT * FROM test1 ORDER BY a || b;
```

############## || ## ORDER BY## ## ##

```
SELECT * FROM test1 ORDER BY a || b COLLATE "fr_FR";
```

23.2.2.

######SQL##SQL############# ##(provider)####### ###########libc###########

####C#################### ## ######### #icu######ICU########## ICU######PostgreSQL########ICU#################### ##

libc##########setlocale()##################LC_COLLATE#LC_CTYPE######## ######## (################################LC_COLLATE######### ###### #LC_CTYPE####LC_COLLATE#################### ########LC_CTYPE############ ############################) ###libc#########################(23.3#########)# ### ##############################

icu################ICU################(collator)######## ICU##collate###ctype###### ############################ ###ICU######################### ################## ICU##################

23.2.2.1.

##################################

unicode

#####[72.195122]##############Unicode#############Unicode################ ## ################## #############ICU######### ########ICU root### ########## und-x-icu (#undefined##)########## ########SQL###### ##Unicode################Unicode######################### ################## ## #############ICU##########POSTGRES#ICU################################ ##########ICU############# und-x-icu (#undefined##)########## ####

ucs_basic

########SQL#####################Unicode################ASCII###A####Z#### ########## ######################(stable)### ########UTF8######### ####### ##UTF8########libc######C##########

pg_c_utf8

##############################Unicode############### # #lower#initcap#upper###Unicode################## ##############)### ##POSIX###Unicode##############[1]# ###########(stable)##########POSTGRES#### ################UTF8#########

C (equivalent to POSIX)

######C#POSIX########C############ ########################ASCII###A## ##Z################ ##(stable)####### ############################

default

##

#######################

23.2.2.2.

######################(newlocale######)#########################ICU######### ######################initdb## #pg_collation#######

#################SELECT * FROM pg_collation##########psql##\dOS+######### ##

[1] https://www.unicode.org/reports/tr18/#Compatibility_Properties

23.2.2.2.1. libc####

################de_DE.utf8################### initdb##de_DE.utf8#####
#LC_COLLATE#LC_CTYPE#####UTF8########de_DE.utf8################ #######
#####.utf8################# #####de_DE################### #################
################ ##

libc###locale -a#
LC_COLLATE#LC_CTYPE######libc#################################
#######################CREATE COLLATION########### ####################
##pg_import_system_collations()############

pg_collation#########
###de_DE##
############################ ##
####default#C###POSIX##

PostgreSQL## ####

```
SELECT a COLLATE "C" < b COLLATE "POSIX" FROM test1;
```

##C#POSIX###################### ######################################

23.2.2.2.2. ICU####

ICU################################## ICU######################## ##########
###################### initdb#ICU#API################### #################
ICU##########libc############SQL##############-x-icu#####BCP 47############
####

#####################

de-x-icu

#################

de-AT-x-icu

#######################

####de-DE-x-icu####de-CH-x-icu######### ################de-x-icu####
###

und-x-icu (#undefined##)

ICU#root###### ############################

#####################ICU######### ############################pg_collation#
#ICU########### ################collation "de-x-icu" for encoding "WIN874" does not exist######################

23.2.2.3.

#########################SQL####CREATE COLLATION##################

############################pg_catalog######### ############################
######pg_dump################

23.2.2.3.1. libc####

##########libc###########

```
CREATE COLLATION german (provider = libc, locale = 'de_DE');
```

######locale############################## Unix########locale -a##########
##########

#####libc###
################# ######################################23.2.2.3.3##############
##
#pg_import_system_collations()##########

23.2.2.3.2. ICU####

ICU################

```
CREATE COLLATION german (provider = icu, locale = 'de-DE');
```

ICU#####BCP 47####################libc################# ######libc##############
########

#########ICU#####################################ICU###### ##########23.2.3#####
#####

23.2.2.3.3.

####CREATE COLLATION############################## ######################
####################################ICU################################## #####
##

```
CREATE COLLATION german FROM "de_DE";
CREATE COLLATION french FROM "fr-x-icu";
```

23.2.2.4.

######################## ###################### ###########################
####################
####################Unicode#################### ############################
####### deterministic################################ ###########Unicode Technical Standard 10[2]#########

################CREATE COLLATION#deterministic = false########### #####
#####

```
CREATE COLLATION ndcoll (provider = icu, locale = 'und', deterministic =
 false);
```

################Unicode######### ################################ ############
ICU################ #########

```
CREATE COLLATION case_insensitive (provider = icu, locale = 'und-u-ks-
level2', deterministic = false);
CREATE COLLATION ignore_accents (provider = icu, locale = 'und-u-ks-
level1-kc-true', deterministic = false);
```

##Unicode#############
########################
B-tree## ########################
################### ################################

[2] https://www.unicode.org/reports/tr10

> **###**
>
> ###Unicode##################################normalize###is normalized## ################### #####

23.2.3. ICU########

ICU## ############### ################### ########:

```
-- ignore differences in accents and case
CREATE COLLATION ignore_accent_case (provider = icu, deterministic =
 false, locale = 'und-u-ks-level1');
SELECT 'Å' = 'A' COLLATE ignore_accent_case; -- true
SELECT 'z' = 'Z' COLLATE ignore_accent_case; -- true

-- upper case letters sort before lower case.
CREATE COLLATION upper_first (provider = icu, locale = 'und-u-kf-upper');
SELECT 'B' < 'b' COLLATE upper_first; -- true

-- treat digits numerically and ignore punctuation
CREATE COLLATION num_ignore_punct (provider = icu, deterministic = false,
 locale = 'und-u-ka-shifted-kn');
SELECT 'id-45' < 'id-123' COLLATE num_ignore_punct; -- true
SELECT 'w;x*y-z' = 'wxyz' COLLATE num_ignore_punct; -- true
```

##############23.2.3.2######### ###23.2.3.5#########

23.2.3.1. ICU#####

ICU##2## ################### ##### ##########################

23.1## Unicode##U+2063########### ###############identic###################

#23.1 ICU#######

###	##	'f' = 'f'	'ab' = U&'a\2063b'	'x-y' = 'x_y'	'g' = 'G'	'n' = 'ñ'	'y' = 'z'
level1	####	true	true	true	true	true	false
level2	#####	true	true	true	true	false	false
level3	########	true	true	true	false	false	false
level4	Punctuation[a]	true	true	false	false	false	false
identic	###	true	false	false	false	false	false

[a]only with `ka-shifted`; see # 23.2

################################## #####'á'#######U&'\0061\0301'######## ###U&'\00E1'###############identic############### ######################## ##deterministic#true###################

23.2.3.1.1.

```
CREATE COLLATION level3 (provider = icu, deterministic = false, locale =
 'und-u-ka-shifted-ks-level3');
CREATE COLLATION level4 (provider = icu, deterministic = false, locale =
 'und-u-ka-shifted-ks-level4');
CREATE COLLATION identic (provider = icu, deterministic = false, locale =
 'und-u-ka-shifted-ks-identic');

-- invisible separator ignored at all levels except identic
SELECT 'ab' = U&'a\2063b' COLLATE level4; -- true
SELECT 'ab' = U&'a\2063b' COLLATE identic; -- false

-- punctuation ignored at level3 but not at level 4
SELECT 'x-y' = 'x_y' COLLATE level3; -- true
SELECT 'x-y' = 'x_y' COLLATE level4; -- false
```

23.2.3.2. ICU###########

23.2####################

#23.2 ICU######

##	#	#####	##
co	emoji, phonebk, standard, ...	standard	########################23.2.3.5######## ##
ka	noignore, shifted	noignore	shifted######################### ##################### ########## ##ks#level3############## ########### ############kv#######
kb	true, false	false	###2######## ########und-u-kb#'àe'#'aé'##########
kc	true, false	false	#################3##########2.5####### ### true#########ks#level1############## #####################
kf	upper, lower, false	false	upper############################ lower############################ false############################
kn	true, false	false	true################################## ########### ####'id-45'#'id-123'### ########
kk	true, false	false	############ ################### ##### ##false################## ########### ############################## ############################## ############ ############## #U&'\0065\0323\0302'#U&'\0065\0302\0323'## ##################################e#### ## ################################## ####### ################
kr	space, punct, symbol, currency, digit, *script-id*		1##############BCP 47*script-id##### ## #####latn("###")###grek("####")### ## ###"-"#######

##	#	#####	##
			############### ################## ########################### ##### #digit-currency-space######### #und-u-kr-digit-currency-space### ##################
ks	level1, level2, level3, level4, identic	level3	######################level1###### ########identic################ ## ## 23.1##########
kv	space, punct, symbol, currency	punct	###3################ ############### ##### #####symbol#punct###space## #### ###############ka#shifted##### ##ks#level3###############

################### ##################### ################23.2.3.5##########

> ##
>
> ########################deterministic#false#################### #23.2.2.4##### #####ka#shifted########################### 23.2#####

23.2.3.3.

```
CREATE COLLATION "de-u-co-phonebk-x-icu" (provider = icu, locale =
'de-u-co-phonebk');
```

   ###################

```
CREATE COLLATION "und-u-co-emoji-x-icu" (provider = icu, locale =
'und-u-co-emoji');
```

   ###########root#####Unicode Technical Standard #51####

```
CREATE COLLATION latinlast (provider = icu, locale = 'en-u-kr-grek-
latn');
```

   ##

```
CREATE COLLATION upperfirst (provider = icu, locale = 'en-u-kf-
upper');
```

   ##

```
CREATE COLLATION special (provider = icu, locale = 'en-u-kf-upper-
kr-grek-latn');
```

   ################

23.2.3.4. ICU#####

[3]

########root#############################

[3] https://unicode-org.github.io/icu/userguide/collation/customization/

```
CREATE COLLATION custom (provider = icu, locale = 'und', rules = '&V << w
 <<< W');
```

##########W###V################################## ########################
##

##########EBCDIC########US-ASCII##########################
####ebcdic######

```
CREATE COLLATION ebcdic (provider = icu, locale = 'und',
rules = $$
& ' ' < '.' < '<' < '(' < '+' < \|
< '&' < '!' < '$' < '*' < ')' < ';'
< '-' < '/' < ',' < '%' < '_' < '>' < '?'
< '`' < ':' < '#' < '@' < \' < '=' < '"'
<*a-r < '~' <*s-z < '^' < '[' < ']'
< '{' <*A-I < '}' <*J-R < '\' <*S-Z <*0-9
$$);

SELECT c
FROM (VALUES ('a'), ('b'), ('A'), ('B'), ('1'), ('2'), ('!'), ('^')) AS
 x(c)
ORDER BY c COLLATE ebcdic;
 c
---
 !
 a
 b
 ^
 A
 B
 1
 2
```

23.2.3.5. ICU#####

###(23.2.3)######################## ##########ICU################################

- Unicode Technical Standard #35[4]
- BCP 47[5]
- CLDR repository[6]
- https://unicode-org.github.io/icu/userguide/locale/
- https://unicode-org.github.io/icu/userguide/collation/

23.3.

PostgreSQL##############################ISO 8859################EUC###Unix##
###UTF-8#Mule## ########################
##initdb#####
PostgreSQL###################### ##
###########

##################################LC_CTYPE########LC_COLLATE################
###################################### C####POSIX############################## ####libc#######

[4] https://www.unicode.org/reports/tr35/tr35-collation.html
[5] https://www.rfc-editor.org/info/bcp47
[6] https://github.com/unicode-org/cldr/blob/master/common/bcp47/collation.xml

################################ ####Windows##UTF-8###################### ICU# ##ICU################

23.3.1.

PostgreSQL############ 23.3######

#23.3 PostgreSQL#####

##	##	##	####	ICU?	####/##	##
BIG5	Big Five	###	###	###	1-2	WIN950#Windows950
EUC_CN	Extended UNIX Code-CN	###	##	##	1-3	
EUC_JP	Extended UNIX Code-JP	###	##	##	1-3	
EUC_JIS_2004	Extended UNIX Code-JP, JIS X 0213	###	##	###	1-3	
EUC_KR	Extended UNIX Code-KR	###	##	##	1-3	
EUC_TW	Extended UNIX Code-TW	#######	##	##	1-3	
GB18030	National Standard	###	###	###	1-4	
GBK	Extended National Standard	###	###	###	1-2	WIN936#Windows936
ISO_8859_5	ISO 8859-5#ECMA 113	###/###	##	##	1	
ISO_8859_6	ISO 8859-6#ECMA 114	###/#####	##	##	1	
ISO_8859_7	ISO 8859-7#ECMA 118	###/#####	##	##	1	
ISO_8859_8	ISO 8859-8#ECMA 121	###/#####	##	##	1	
JOHAB	JOHAB	########	###	###	1-3	
KOI8R	KOI8-R	#########	##	##	1	KOI8
KOI8U	KOI8-U	###########	##	##	1	
LATIN1	ISO 8859-1#ECMA 94	######	##	##	1	ISO88591

##	##	##	####	**ICU?**	####/##	##
LATIN2	ISO 8859-2#ECMA 94	#######	##	##	1	ISO88592
LATIN3	ISO 8859-3#ECMA 94	######	##	##	1	ISO88593
LATIN4	ISO 8859-4#ECMA 94	######	##	##	1	ISO88594
LATIN5	ISO 8859-9#ECMA 128	###	##	##	1	ISO88599
LATIN6	ISO 8859-10#ECMA 144	##	##	##	1	ISO885910
LATIN7	ISO 8859-13	#####	##	##	1	ISO885913
LATIN8	ISO 8859-14	###	##	##	1	ISO885914
LATIN9	ISO 8859-15	LATIN1##### #######	##	##	1	ISO885915
LATIN10	ISO 8859-16#ASRO SR 14111	#####	##	###	1	ISO885916
MULE_INTERNAL	Mule#####	###Emacs	##	###	1-4	
SJIS	Shift JIS	###	###	###	1-2	Mskanji#ShiftJIS#WIN932
SHIFT_JIS_2004	Shift JIS, JIS X 0213	###	###	###	1-2	
SQL_ASCII	###########	###	##	###	1	
UHC	########	###	###	###	1-2	WIN949#Windows949
UTF8	Unicode#8###	###	##	##	1-4	Unicode
WIN866	Windows CP866	#####	##	##	1	ALT
WIN874	Windows CP874	###	##	###	1	
WIN1250	Windows CP1250	#######	##	##	1	
WIN1251	Windows CP1251	#####	##	##	1	WIN
WIN1252	Windows CP1252	######	##	##	1	
WIN1253	Windows CP1253	####	##	##	1	
WIN1254	Windows CP1254	###	##	##	1	
WIN1255	Windows CP1255	####	##	##	1	

##	##	##	####	ICU?	####/##	##
WIN1256	Windows CP1256	#####	##	##	1	
WIN1257	Windows CP1257	#####	##	##	1	
WIN1258	Windows CP1258	#####	##	##	1	ABC#TCVN#TCVN5712#VSCII

##########API################################ ###PostgreSQL JDBC#### #MULE_INTERNAL#LATIN6#LATIN8####LATIN10#########

SQL_ASCII##############################SQL_ASCII########0##127######ASCII# #########128##255########## ###SQL_ASCII#################################### ########################### ######ASCII##############SQL_ASCII################ ###########PostgreSQL#ASCII############################

23.3.2.

initdb#PostgreSQL############################## ##########

```
initdb -E EUC_JP
```

###############EUC_JP######Unix########## ##################-E#####--encoding########## -E######--encoding##############initdb################ ########################

###

```
createdb -E EUC_KR -T template0 --lc-collate=ko_KR.euckr --lc-
ctype=ko_KR.euckr korean
```

###EUC_KR######ko_KR#########korean################## SQL################## ####

```
CREATE DATABASE korean WITH ENCODING 'EUC_KR' LC_COLLATE='ko_KR.euckr'
 LC_CTYPE='ko_KR.euckr' TEMPLATE=template0;
```

########template0############################## ############################## ## #######22.3##########

#############pg_database############### psql#-l######\l#################

```
$ psql -l
                                        List of databases
   Name    |  Owner   | Encoding  |  Collation  |    Ctype    |
 Access Privileges
-----------+----------+-----------+-------------+-------------
+-----------------------------------
 clocaledb | hlinnaka | SQL_ASCII | C           | C           |
 englishdb | hlinnaka | UTF8      | en_GB.UTF8  | en_GB.UTF8  |
 japanese  | hlinnaka | UTF8      | ja_JP.UTF8  | ja_JP.UTF8  |
 korean    | hlinnaka | EUC_KR    | ko_KR.euckr | ko_KR.euckr |
 postgres  | hlinnaka | UTF8      | fi_FI.UTF8  | fi_FI.UTF8  |
 template0 | hlinnaka | UTF8      | fi_FI.UTF8  | fi_FI.UTF8  | {=c/
hlinnaka,hlinnaka=CTc/hlinnaka}
 template1 | hlinnaka | UTF8      | fi_FI.UTF8  | fi_FI.UTF8  | {=c/
hlinnaka,hlinnaka=CTc/hlinnaka}
```

```
(7 rows)
```

> **##**
>
> ################PostgreSQL##LC_CTYPE########################### ##### ########################### ## ############## ##
>
> PostgreSQL##LC_CTYPE#C####POSIX################SQL_ASCII########### ################ #######SQL_ASCII#################################### ## ### ####################

23.3.3.

PostgreSQL################23.3.4 ##

##PostgreSQL############ #### ###############

- psql#`\encoding`######### `\encoding`################################ ####### ##SJIS##################

  ```
  \encoding SJIS
  ```

- libpq (32.11)###########################
- `SET client_encoding TO`###### ##SQL#######################

  ```
  SET CLIENT_ENCODING TO 'value';
  ```

  ##SQL###SET NAMES###############

  ```
  SET NAMES 'value';
  ```

  ################################

  ```
  SHOW client_encoding;
  ```

  ########################

  ```
  RESET client_encoding;
  ```

- `PGCLIENTENCODING`###### #########PGCLIENTENCODING#################### ########################### ##############################
- client_encoding####### `client_encoding`#################################### ########## ############################

EUC_JP########LATIN1################ ####################LATIN1######### ####################

################SQL_ASCII### ####### #######SQL_ASCII## ################################ ###########SQL_ASCII###########ASCII######## ##################

23.3.4.

PostgreSQL##pg_conversion#######################2################ PostgreSQL#### 23.4###### 23.5################################## CREATE CONVERSIONSQL###

#23.4

########	################
BIG5	########################
EUC_CN	*EUC_CN*, MULE_INTERNAL, UTF8
EUC_JP	*EUC_JP*, MULE_INTERNAL, SJIS, UTF8
EUC_JIS_2004	*EUC_JIS_2004*, SHIFT_JIS_2004, UTF8
EUC_KR	*EUC_KR*, MULE_INTERNAL, UTF8
EUC_TW	*EUC_TW*, BIG5, MULE_INTERNAL, UTF8
GB18030	########################
GBK	########################
ISO_8859_5	*ISO_8859_5*, KOI8R, MULE_INTERNAL, UTF8, WIN866, WIN1251
ISO_8859_6	*ISO_8859_6*, UTF8
ISO_8859_7	*ISO_8859_7*, UTF8
ISO_8859_8	*ISO_8859_8*, UTF8
JOHAB	########################
KOI8R	*KOI8R*, ISO_8859_5, MULE_INTERNAL, UTF8, WIN866, WIN1251
KOI8U	*KOI8U*, UTF8
LATIN1	*LATIN1*, MULE_INTERNAL, UTF8
LATIN2	*LATIN2*, MULE_INTERNAL, UTF8, WIN1250
LATIN3	*LATIN3*, MULE_INTERNAL, UTF8
LATIN4	*LATIN4*, MULE_INTERNAL, UTF8
LATIN5	*LATIN5*, UTF8
LATIN6	*LATIN6*, UTF8
LATIN7	*LATIN7*, UTF8
LATIN8	*LATIN8*, UTF8
LATIN9	*LATIN9*, UTF8
LATIN10	*LATIN10*, UTF8
MULE_INTERNAL	*MULE_INTERNAL*, BIG5, EUC_CN, EUC_JP, EUC_KR, EUC_TW, ISO_8859_5, KOI8R, LATIN1 ## LATIN4, SJIS, WIN866, WIN1250, WIN1251
SJIS	########################
SHIFT_JIS_2004	########################
SQL_ASCII	##############
UHC	########################
UTF8	#####################
WIN866	*WIN866*, ISO_8859_5, KOI8R, MULE_INTERNAL, UTF8, WIN1251
WIN874	*WIN874*, UTF8

########	################
WIN1250	*WIN1250*, LATIN2, MULE_INTERNAL, UTF8
WIN1251	*WIN1251*, ISO_8859_5, KOI8R, MULE_INTERNAL, UTF8, WIN866
WIN1252	*WIN1252*, UTF8
WIN1253	*WIN1253*, UTF8
WIN1254	*WIN1254*, UTF8
WIN1255	*WIN1255*, UTF8
WIN1256	*WIN1256*, UTF8
WIN1257	*WIN1257*, UTF8
WIN1258	*WIN1258*, UTF8

#23.5

### [a]	########	########
big5_to_euc_tw	BIG5	EUC_TW
big5_to_mic	BIG5	MULE_INTERNAL
big5_to_utf8	BIG5	UTF8
euc_cn_to_mic	EUC_CN	MULE_INTERNAL
euc_cn_to_utf8	EUC_CN	UTF8
euc_jp_to_mic	EUC_JP	MULE_INTERNAL
euc_jp_to_sjis	EUC_JP	SJIS
euc_jp_to_utf8	EUC_JP	UTF8
euc_kr_to_mic	EUC_KR	MULE_INTERNAL
euc_kr_to_utf8	EUC_KR	UTF8
euc_tw_to_big5	EUC_TW	BIG5
euc_tw_to_mic	EUC_TW	MULE_INTERNAL
euc_tw_to_utf8	EUC_TW	UTF8
gb18030_to_utf8	GB18030	UTF8
gbk_to_utf8	GBK	UTF8
iso_8859_10_to_utf8	LATIN6	UTF8
iso_8859_13_to_utf8	LATIN7	UTF8
iso_8859_14_to_utf8	LATIN8	UTF8
iso_8859_15_to_utf8	LATIN9	UTF8
iso_8859_16_to_utf8	LATIN10	UTF8
iso_8859_1_to_mic	LATIN1	MULE_INTERNAL
iso_8859_1_to_utf8	LATIN1	UTF8
iso_8859_2_to_mic	LATIN2	MULE_INTERNAL
iso_8859_2_to_utf8	LATIN2	UTF8
iso_8859_2_to_windows_1250	LATIN2	WIN1250
iso_8859_3_to_mic	LATIN3	MULE_INTERNAL
iso_8859_3_to_utf8	LATIN3	UTF8
iso_8859_4_to_mic	LATIN4	MULE_INTERNAL
iso_8859_4_to_utf8	LATIN4	UTF8

### [a]	########	########
iso_8859_5_to_koi8_r	ISO_8859_5	KOI8R
iso_8859_5_to_mic	ISO_8859_5	MULE_INTERNAL
iso_8859_5_to_utf8	ISO_8859_5	UTF8
iso_8859_5_to_windows_1251	ISO_8859_5	WIN1251
iso_8859_5_to_windows_866	ISO_8859_5	WIN866
iso_8859_6_to_utf8	ISO_8859_6	UTF8
iso_8859_7_to_utf8	ISO_8859_7	UTF8
iso_8859_8_to_utf8	ISO_8859_8	UTF8
iso_8859_9_to_utf8	LATIN5	UTF8
johab_to_utf8	JOHAB	UTF8
koi8_r_to_iso_8859_5	KOI8R	ISO_8859_5
koi8_r_to_mic	KOI8R	MULE_INTERNAL
koi8_r_to_utf8	KOI8R	UTF8
koi8_r_to_windows_1251	KOI8R	WIN1251
koi8_r_to_windows_866	KOI8R	WIN866
koi8_u_to_utf8	KOI8U	UTF8
mic_to_big5	MULE_INTERNAL	BIG5
mic_to_euc_cn	MULE_INTERNAL	EUC_CN
mic_to_euc_jp	MULE_INTERNAL	EUC_JP
mic_to_euc_kr	MULE_INTERNAL	EUC_KR
mic_to_euc_tw	MULE_INTERNAL	EUC_TW
mic_to_iso_8859_1	MULE_INTERNAL	LATIN1
mic_to_iso_8859_2	MULE_INTERNAL	LATIN2
mic_to_iso_8859_3	MULE_INTERNAL	LATIN3
mic_to_iso_8859_4	MULE_INTERNAL	LATIN4
mic_to_iso_8859_5	MULE_INTERNAL	ISO_8859_5
mic_to_koi8_r	MULE_INTERNAL	KOI8R
mic_to_sjis	MULE_INTERNAL	SJIS
mic_to_windows_1250	MULE_INTERNAL	WIN1250
mic_to_windows_1251	MULE_INTERNAL	WIN1251
mic_to_windows_866	MULE_INTERNAL	WIN866
sjis_to_euc_jp	SJIS	EUC_JP
sjis_to_mic	SJIS	MULE_INTERNAL
sjis_to_utf8	SJIS	UTF8
windows_1258_to_utf8	WIN1258	UTF8
uhc_to_utf8	UHC	UTF8
utf8_to_big5	UTF8	BIG5
utf8_to_euc_cn	UTF8	EUC_CN
utf8_to_euc_jp	UTF8	EUC_JP
utf8_to_euc_kr	UTF8	EUC_KR
utf8_to_euc_tw	UTF8	EUC_TW

### [a]	########	########
utf8_to_gb18030	UTF8	GB18030
utf8_to_gbk	UTF8	GBK
utf8_to_iso_8859_1	UTF8	LATIN1
utf8_to_iso_8859_10	UTF8	LATIN6
utf8_to_iso_8859_13	UTF8	LATIN7
utf8_to_iso_8859_14	UTF8	LATIN8
utf8_to_iso_8859_15	UTF8	LATIN9
utf8_to_iso_8859_16	UTF8	LATIN10
utf8_to_iso_8859_2	UTF8	LATIN2
utf8_to_iso_8859_3	UTF8	LATIN3
utf8_to_iso_8859_4	UTF8	LATIN4
utf8_to_iso_8859_5	UTF8	ISO_8859_5
utf8_to_iso_8859_6	UTF8	ISO_8859_6
utf8_to_iso_8859_7	UTF8	ISO_8859_7
utf8_to_iso_8859_8	UTF8	ISO_8859_8
utf8_to_iso_8859_9	UTF8	LATIN5
utf8_to_johab	UTF8	JOHAB
utf8_to_koi8_r	UTF8	KOI8R
utf8_to_koi8_u	UTF8	KOI8U
utf8_to_sjis	UTF8	SJIS
utf8_to_windows_1258	UTF8	WIN1258
utf8_to_uhc	UTF8	UHC
utf8_to_windows_1250	UTF8	WIN1250
utf8_to_windows_1251	UTF8	WIN1251
utf8_to_windows_1252	UTF8	WIN1252
utf8_to_windows_1253	UTF8	WIN1253
utf8_to_windows_1254	UTF8	WIN1254
utf8_to_windows_1255	UTF8	WIN1255
utf8_to_windows_1256	UTF8	WIN1256
utf8_to_windows_1257	UTF8	WIN1257
utf8_to_windows_866	UTF8	WIN866
utf8_to_windows_874	UTF8	WIN874
windows_1250_to_iso_8859_2	WIN1250	LATIN2
windows_1250_to_mic	WIN1250	MULE_INTERNAL
windows_1250_to_utf8	WIN1250	UTF8
windows_1251_to_iso_8859_5	WIN1251	ISO_8859_5
windows_1251_to_koi8_r	WIN1251	KOI8R
windows_1251_to_mic	WIN1251	MULE_INTERNAL
windows_1251_to_utf8	WIN1251	UTF8
windows_1251_to_windows_866	WIN1251	WIN866
windows_1252_to_utf8	WIN1252	UTF8

### [a]	########	########
windows_1256_to_utf8	WIN1256	UTF8
windows_866_to_iso_8859_5	WIN866	ISO_8859_5
windows_866_to_koi8_r	WIN866	KOI8R
windows_866_to_mic	WIN866	MULE_INTERNAL
windows_866_to_utf8	WIN866	UTF8
windows_866_to_windows_1251	WIN866	WIN
windows_874_to_utf8	WIN874	UTF8
euc_jis_2004_to_utf8	EUC_JIS_2004	UTF8
utf8_to_euc_jis_2004	UTF8	EUC_JIS_2004
shift_jis_2004_to_utf8	SHIFT_JIS_2004	UTF8
utf8_to_shift_jis_2004	UTF8	SHIFT_JIS_2004
euc_jis_2004_to_shift_jis_2004	EUC_JIS_2004	SHIFT_JIS_2004
shift_jis_2004_to_euc_jis_2004	SHIFT_JIS_2004	EUC_JIS_2004

[a] ################ ####################################_to_######################## ######## ###### 23.3##########################

23.3.5.

##################################

CJKV########: ########### & #######

EUC_JP#EUC_CN#EUC_KR#EUC_TW############

https://www.unicode.org/

Unicode###Web######

RFC 3629[7]

###UTF-8#8###UCS/Unicode###############

[7] https://datatracker.ietf.org/doc/html/rfc3629

#24#

################PostgreSQL################################## ################ ################## ####cron##############Windows########################## ## ##

###########1############################ ##################################### ######################## PostgreSQL######################25###########

#########24.1####### ################### ###################24.1.3#######

################################### #######24.3#######

check_postgres[1]#################################### check_postgres#Nagios###MRTG ############################

PostgreSQL############################## ################################### ####################

24.1.

PostgreSQL################################### ###############24.1.6########### ############################## ## ####### ##############cron#################################VACUUM######### ################################## ## ########### ## ##

24.1.1.

PostgreSQL#VACUUM##

1. ####################################
2. PostgreSQL###########################
3. ######### ##########################
4. *#######ID################ID#################*

#################### 1#1##VACUUM################################

VACUUM#####VACUUM#VACUUM FULL############# VACUUM FULL### ##################################### ###VACUUM############################## ######## #SELECT#INSERT#UPDATE#DELETE################### ########### #ALTER TABLE################################ VACUUM FULL########## ####ACCESS EXCLUSIVE## ########## #VACUUM######VACUUM FULL###########

VACUUM########I/O############## ################################ ############## ## 19.4.4##########

24.1.2.

PostgreSQL#####UPDATE####DELETE############## ##############MVCC##13######## ################### ## ##########

[1] https://bucardo.org/check_postgres/

################################## ### ################ ###VACUUM##############

#####VACUUM### ########### ####################### ### ############### ####VACUUM FULL### ####### ################################ ##################################### ##

#################VACUUM FULL################VACUUM######## ############### ########### #####VACUUM FULL###### ##################################### ############### ## ## VACUUM FULL## ############################### ###########VACUUM############VACUUM FULL#### ################################

##VACUUM FULL################# ###### ################################## ##################################### ### ### ########1###################### ##################### ##################################VACUUM##############

################################VACUUM#1#1######################### ######## ##################################### ## #########VACUUM########### 1## vacuumdb#################

###
##VACUUM################### ### ######################################VACUUM FULL#########CLUSTER### ########ALTER TABLE###1############## ################################## ############## ##########ACCESS EXCLUSIVE########## ################ ##

###
#######################DELETE###VACUUM########TRUNCATE############## TRUNCATE#################### ##########################VACUUM#VACUUM FULL########### ########MVCC##############

24.1.3.

PostgreSQL### #######ANALYZE## ############ ####################VACUUM####################### ############## ######## ##

##ANALYZE########## ######## ##ANALYZE###################### ###### ######################ANALYZE######### ############################## ##

####################ANALYZE########

##############timestamp############################## ##### #############Web################URL############################## ##URL#####

###########ANALYZE############ ############################### ############## ##################################### ### ## ANALYZE##

> ###
>
> #####ANALYZE############################ANALYZE#################### ################### WHERE## ########### ALTER TABLE SET STATISTICS#######default_statistics_target#### ##########################
>
> ################################### ################################### ######################### ######################################

> ###
>
> ##ANALYZE########### ####### ##ANALYZE####### #########

> ###
>
> ############################ANALYZE########### ######################## ############################## ### ########################ANALYZE############

24.1.4.

##############

#####PostgreSQL## PostgreSQL## ############################## ############################## ############### ############################## ## ### ##

24.1.5. ########ID#########

PostgreSQL#MVCC#########################ID#XID######################## ####### #####XID#########XID## ######## ####ID########32###########40#################################### XID###### ####0## ############### #### ## ########################## #########20##########################

##########################VACUUM##
#################################### PostgreSQL####
XID#FrozenTransactionId####### ##XID####XID##################XID#########
###XID#2######modulo-2^{32}################ #########XID###20#######XID#20#
#######XID############# #########XID######################## #########XID##
############################20################XID################## ########
#########20##
#####XID#FrozenTransactionId###
###

> ##
>
> 9.4#########PostgreSQL######XID####FrozenTransactionId#############
> #########xmin############### ##################################xmin#
> ################# ####9.4#########pg_upgrade##################xmin#
> FrozenTransactionId (2)################
>
> ############xmin#BootstrapTransactionId (1)#####################
> initdb################### FrozenTransactionId########XID#######XID#
> ##############

vacuum_freeze_min_age##################XID#################### ##############
###
############################

VACUUM## ##############################
############XID############################## ########VACUUM##############
######################## ###################VACUUM###########################
##XID##MXID########## ################
################# vacuum_freeze_table_age#VACUUM############### ###############
###################vacuum_freeze_table_age##vacuum_freeze_min_age####
################################## vacuum_freeze_table_age#0######VACUUM##
###################

###########################20############################
#vacuum_freeze_min_age############ ##################################
######################autovacuum_freeze_max_age##################XID########
############################## ########################

#######################################autovacuum_freeze_max_age -
vacuum_freeze_min_age######################## ############################
############ ##
###########
##autovacuum_freeze_max_age#######vacuum_freeze_min_age##############
#######

vacuum_freeze_table_age##########0.95 * autovacuum_freeze_max_age### #
###################### autovacuum_freeze_max_age#######################
##########0.95####################VACUUM################### ###############
#VACUUM###
##vacuum_freeze_table_age#autovacuum_freeze_max_age################ #
##########
##################

autovacuum_freeze_max_age##########vacuum_freeze_table_age###########
##################pg_xact#pg_commit_ts###############
autovacuum_freeze_max_age#######################

(track_commit_timestamp########)######################## #######1##########
2############autovacuum_freeze_max_age##########20##########pg_xact####
0.5#######pg_commit_ts##20GB############## ############################
##autovacuum_freeze_max_age################## #####
##pg_xact#pg_commit_ts########################## #######2###############
#pg_xact####50MB#pg_commit_ts####2GB#########

vacuum_freeze_min_age ########1######## #####VACUUM################### ####
###############XID################################## ######################
##################

#####################XID###########VACUUM########pg_class#pg_database#
XID########## ###########pg_class##relfrozenxid####relfrozenxid########
#####VACUUM############VACUUM#######################XID####### ###########
###pg_database##datfrozenxid####################XID####### #############
########relfrozenxid######## ################################

```
SELECT c.oid::regclass as table_name,
       greatest(age(c.relfrozenxid),age(t.relfrozenxid)) as age
FROM pg_class c
LEFT JOIN pg_class t ON c.reltoastrelid = t.oid
WHERE c.relkind IN ('r', 'm');

SELECT datname, age(datfrozenxid) FROM pg_database;
```

age######XID############XID#################

> ###
>
> VACUUM#####VERBOSE###############VACUUM####################### ###
> #relfrozenxid###relminmxid################################
> (log_autovacuum_min_duration######)##########################VACUUM######
> #######################

VACUUM##############################relfrozenxid#############XID########
####################### ####relfrozenxid#vacuum_freeze_table_age########
########VACUUM#FREEZE###
################## VACUUM##############################age(relfrozenxid)##
#####vacuum_freeze_min_age############## #VACUUM#########################
VACUUM#relfrozenxid##############XID##############################
############# relfrozenxid######VACUUM#autovacuum_freeze_max_age######
##

#######################XID#########################XID########4000#########
##########################

```
WARNING:  database "mydb" must be vacuumed within 39985967 transactions
HINT:  To avoid XID assignment failures, execute a database-wide VACUUM in
 that database.
```

###############VACUUM########### ####VACUUM############################ ####
###########################datfrozenxid############### ####################
########300#################XID###########

```
ERROR:  database is not accepting commands that assign new XIDs to avoid
 wraparound data loss in database "mydb"
HINT:  Execute a database-wide VACUUM in that database.
```

#####[85.146444]## ################################## VACUUM################## ####################
#########postmaster## ############
##########
############################## ##################### ######################
##postmaster## ###############

1. ##################### pg_prepared_xacts#age(transactionid)################
   ##### ##################################

2. ############################ pg_stat_activity#age(backend_xid)##
   #age(backend_xmin)######################### ##########################
   ###pg_terminate_backend################

3. #################### pg_stat_replication#####age(xmin)###age(catalog_xmin)#
   ############# ##
   ##

4. ##########VACUUM###### ########VACUUM####### #############relminxid##
   ############VACUUM################ #######VACUUM FULL###############
   XID########################## ######XID############ID####################
   ###VACUUM FREEZE########### ##################################

5. ##

> **##**
>
> ###########postmaster#####VACUUM########################## ############
> ## ########################
> #ID############################## ##################################
> #TRUNCATE###DROP#####VACUUM############## ######################300
> ##################### ####################postgres##################

24.1.5.1.

###########ID################################ ################################
##:############ID################
ID########### ############ID##########ID##############pg_multixact########
####################ID#########xmax########### ########ID################ID#32
##################
#####################32################################

##############VACUUM#######################vacuum_multixact_freeze_min_age#######
#########ID################## #######0####################ID####################
####ID######## ########pg_class.relminmxid##########################ID#####
############### ####vacuum_multixact_freeze_table_age######################## #####
pg_class.relminmxid############
#mxid_age()######

####VACUUM#####################relminmxid################### ###############
##

########autovacuum_multixact_freeze_max_age#####################################
############# ############################2GB################################
#################################### ######################################

#####[93.005181]#XID#######################MXID######################MXID####
#####4000################################## ####XID#########################
#3000##############MXID##########

MXID##############XID#################### ###########################

1. ##
2. pg_stat_activity###########MXID################XID######MXID######## ########################
3. XID##############################MXID####### MXID########################### ################

24.1.6.

PostgreSQL################################## ###VACUUM#ANALYZE############## ########## ## ################### ## ####track_counts#true############################## ########################### ################

################################### ## ################### ######1################autovacuum_naptime################### ############## ################N####################autovacuum_naptime/N### ######## #####autovacuum_max_workers############## ########## #autovacuum_max_workers## ########## ##########################VACUUM###ANALYZE##########
log_autovacuum_min_duration############################

###
################### ## # ### ######## #max_connections####superuser_reserved_connections####################

#####relfrozenxid##autovacuum_freeze_max_age################################
####freeze max age## ##########VACUUM### ##

####### = ######### + ######### * ####

##############autovacuum_vacuum_threshold###########autovacuum_vacuum_scale_factor## ####pg_class.reltuples###

###

######### = ########### + ########### * ####

#####[85.266458]################autovacuum_vacuum_insert_threshold###########
#autovacuum_vacuum_insert_scale_factor### ##
############################ ########################INSERT########UPDATE/
DELETE################################autovacuum_freeze_min_age##############
####UPDATE#DELETE###INSERT#####
############## (##) ####
#relfrozenxid##vacuum_freeze_table_age#########################
##relfrozenxid######################### ################################
############################autovacuum_vacuum_insert_threshold##################
##autovacuum_vacuum_insert_scale_factor### ##
INSERT##########/###############
####################autovacuum_freeze_min_age#################################
########## ## ##########INSERT###
################################ #####relfrozenxid########_####_####_age#

##relfrozenxid# ##########
###########################

################## ###########

= ###### + ###### *

###ANALYZE#########################

(################################
###) ############################ANALYZE#################################
############################## ######################################ANALYZE#
##

############################## ###########SQL################################
##

##############postgresql.conf########(######################)#############
##
###############19.10#########

#####################################(19.4.4#########)#################### ###
#######################I/O############## ########
#autovacuum_vacuum_cost_delay###autovacuum_vacuum_cost_limit########
##

############################## ###########SHARE UPDATE EXCLUSIVE#####
13.2########
#######################ID##############(#####pg_stat_activity############
#####(to prevent wraparound)#######)#########################

##
SHARE UPDATE EXCLUSIVE###########################(###ANALYZE)##### #########################

24.2.

REINDEX###

########B-tree######################## ######################## ##################
##################################### ##
###################### ############################

B-tree############################## B-tree####################################
#############

###B-tree### ##################
######################### ####B-tree#################### ######################
#########################

REINDEX######################## ############ACCESS EXCLUSIVE####
########CONCURRENTLY####################################SHARE UPDATE
EXCLUSIVE###########

24.3.

##############/dev/null################################## ##################
###

> ##
>
> ## ########DDL##
> ################################ ERROR#####################SQL#########
> ###################### ##
> ## ####################################

########################
##

##postgres#stderr###
######### #####PostgreSQL##

####stderr################################## ##################
##postgresql.conf#logging_collector#######true###################### ###
##################19.8.1####### ######################CSV(######)###############
######

####Apache###
#####rotatelogs#PostgreSQL############ ##################stderr##################
pg_ctl################stderr###stdout#######################################
####

```
pg_ctl start | rotatelogs /var/log/pgsql_log 86400
```

PostgreSQL##############################logrotate########################### ###
############################logrotate#################### ##############
##logrotate## ##############SIGHUP#
######postrotate#################################### PostgreSQL#######
#pg_ctl#logrotate############### ##
##################(19.8.1########)#

> ##
>
> ##
> ############ ## logrotate#######
> ### ##################
> ######################prerotate######################

######################syslog#####syslog###################### #####
##postgresql.conf#log_destination#######syslog#syslog################## #
#########################syslog#####SIGHUP############## ##################
##logrotate##############syslog##################

############syslog########################## ################################
###Linux###syslog############################## ############syslog######
##########-############

#
####################################
###########

pgBadger[2]################################ check_postgres[3]#########################
##################Nagios##############

[2] https://pgbadger.darold.net/
[3] https://bucardo.org/check_postgres/

#25#

####################PostgreSQL########################### ################## ####################################

PostgreSQL###############3############

- SQL######
- #################
- #######

############## ####################

25.1. SQL######

###########SQL## #### #######PostgreSQL#pg_dump################### #######################

```
pg_dump dbname > dumpfile
```

########pg_dump############### ######################### ####################### ##pg_dump###

pg_dump##PostgreSQL## ############ ##################################### ####pg_dump########################### ## ## ################## (#####################################-n schema#####-t table##)

pg_dump##########################-h host######-p port########### ########## ##########PGHOST############## ##############PGPORT########################## ##

##PostgreSQL####################pg_dump## ###### ##########-U##########PGUSER########### pg_dump######20################ ####################

##################pg_dump########pg_dump###############PostgreSQL########### #### ## pg_dump##32#####64######### ##

pg_dump######################## ########pg_dump############################## pg_dump#########################ALTER TABLE##############################

25.1.1.

pg_dump##############psql###################### ###########################

```
psql dbname < dumpfile
```

###dumpfile#pg_dump################## dbname###################### ### #createdb -T template0 dbname ######psql#########template0########## psql #pg_dump## ########psql############## ###### #################pg_restore #################

SQL###
######################
#########

######psql#######SQL################# ON_ERROR_STOP#######psql###############
##SQL##########psql#########3####################

```
psql --set ON_ERROR_STOP=on dbname < dumpfile
```

################################ ########1##############################
#######psql#############-1###--
single-transaction############ ##
############## ##

pg_dump#psql## ##########

```
pg_dump -h host1 dbname | psql -h host2 dbname
```

> ##
>
> pg_dump#########template0########## ###template1##################
> #######pg_dump########## ########################template1###########
> ########template0######################

###ANALYZE############# ##
#####24.1.3 # 24.1.6######### ###########PostgreSQL#######################14.4###
#######

25.1.2. pg_dumpall###

pg_dump###################### ########################### ####################
########## ##############################pg_dumpall############### pg_dumpall###
###############
#

```
pg_dumpall > dumpfile
```

#######psql#########

```
psql -f dumpfile postgres
```

postgres
############################pg_dumpall#####################################
######### ##

pg_dumpall###pg_dump#######
###

############pg_dumpall #--globals-only ###################### ################
pg_dump ############################

25.1.3.

#######################################pg_dump########################## #####
##pg_dump###################Unix######################### ################

########. ##############gzip################

```
pg_dump dbname | gzip > filename.gz
```

###############

```
gunzip -c filename.gz | psql dbname
```

###############

```
cat filename.gz | gunzip | psql dbname
```

split###. split### ###2############ ###########

```
pg_dump dbname | split -b 2G - filename
```

###############

```
cat filename* | psql dbname
```

GNU split###############gzip#####################

```
pg_dump dbname | split -b 2G --filter='gzip > $FILE.gz'
```

##########zcat############

pg_dump############. ##PostgreSQL#zlib#################################### ######################### gzip## ##### ################################

```
pg_dump -Fc dbname > filename
```

###########psql####################pg_restore############### #############

```
pg_restore -d dbname filename
```

###pg_dump#pg_restore###################

##################################split###################

pg_dump#####. pg_dump## ################ ####### ####-j################### ##########################

```
pg_dump -j num -F d -f out.dir dbname
```

pg_restore -j############################ ###pg_dump -j################## ##

25.2.

###############PostgreSQL## 18.2##### ################### ##

```
tar -cf backup.tar /usr/local/pgsql/data
```

#########2##############################pg_dump###############

1. ##################################### ############################ #tar#### ## ##### ######18.5########## ##

2. ## ######################## ## #pg_xact/*################################ ########################### ### #############pg_xact## ##### ##

##WAL###### ##### ############ ########################WAL############# CHECKPOINT#####

############WAL### ### ###

########################25.3.2######### ############################ ################### ##25.3.5##########

###########################rsync########### ########################rsync####### #rsync --checksum############################ (rsync##############1####### #######--checksum######) ##rsync## rsync######## ##

###################SQL######################## #pg_dump######################### ######################### ################################

25.3. ##########################PITR#

PostgreSQL#####################pg_wal/################WAL######### ######## ############################ ########################## ######################## ## ################### ###################3############ ##################WAL######################### ### ##WAL############################ ##

- ## ################################### ### ################################### ################################ ##tar###### ###########

- #####WAL################################WAL################################## ## ##

- WAL################################## ### ########################### ## ####################################

- #######WAL###
  ###########2###

> **##**
>
> pg_dump#pg_dumpall###
> ###############WAL######################

#########
############## ##WAL############
######################################

###
###############WAL######### ################################WAL###########
################## ########WAL#####################

25.3.1. WAL########

############PostgreSQL############WAL########## ################1#16#######
#########initdb#############WAL############### ##################WAL#######
################## WAL###
################################## ##
########

WAL###
####NFS################
###
##CD########################## ######################PostgreSQL##############
######################## ######PostgreSQL#######################################
###################### ##########cp#########################C############### ###
###########

WAL##############wal_level########replica####archive_mode#on###
##archive_command#########################archive_library######################## #
############postgresql.conf###########

archive_command###%p##########################%f################## #########
################################### ###%####################%%########## #######
################

```
archive_command = 'test ! -f /mnt/server/archivedir/%f && cp %p /mnt/
server/archivedir/%f'  # Unix
archive_command = 'copy "%p" "C:\\server\\archivedir\\%f"'  # Windows
```

############WAL######/mnt/server/archivedir############ ######### #########
################################ %p###%f#################################

```
test ! -f /mnt/server/archivedir/00000001000000A900000065
 && cp pg_wal/00000001000000A900000065 /mnt/server/
archivedir/00000001000000A900000065
```

##################################

#############PostgreSQL########################## ###########WAL###############
###########################
####################

PostgreSQL########################
################################### ###############PostgreSQL##################
#######################

###############archive_library######################### ##########C######
####################
###############49##########

#################################SIGTERM#########125################
#command not found###############ERROR###FATAL###########################
##postmaster########### ##########pg_stat_archiver#########

##############2##
######
##

####PostgreSQL###########WAL######################## ######################
###########
####################WAL##
#######0#######true########## ##################WAL######################
############0#######false############

##Unix###########test################################## #####Unix########
#cp######-i ## ######
GNU#cp#####-i ##

###
########### ## ###################
###################### ###
###########WAL#########pg_wal/################ #pg_wal/################
#####PostgreSQL###
###

####WAL## ##########
################### ## ######pg_wal/##
##############
####################

###64######ASCII#######################
#######%p#####################%f#############

WAL########PostgreSQL#################################SQL##############
##########postgresql.conf#pg_hba.conf####pg_ident.conf################
########### ## ##############
####19.2#########

##################WAL################## ############WAL###################
#######################
#######archive_timeout################WAL############################## #####
#########
#archive_timeout############### ############### ########archive_timeout#
#######

#################################pg_switch_wal###########################
####WAL########### 9.95#######

wal_level#minimal####14.4.7###############SQL#####WAL################## ##
###WAL###########
################ #######wal_level################ #####
##archive_command#archive_library###################### ###########

##########archive_command#####''########## ###########
#archive_command########WAL#####pg_wal/######

25.3.2.

#####################pg_basebackup ######### #######TAR#################### ######### ###pg_basebackup##################API######################## 25.3.4# ####

################################ #######full_page_writes################### #######full_page_writes################################

##WAL########################## ########################WAL########################## ##########WAL############### #######WAL##### 000000010000123400005 5CD### ###############000000010000123400005 5CD.007C9330.backup############ ######2######WAL################################ ########################## ##########WAL##WAL######## ################################ ## ##

############################## ####pg_basebackup############################## #####WAL########### ## ##############

###############WAL##WAL############## ########################### ##— ### #####WAL## ##

25.3.3. Making an Incremental Backup

######pg_basebackup##########################--incremental########### ####### ##########################--incremental############## ##################### ## ##################################

###pg_wal/summaries### #####WAL######### ## ################## ###########LSN##############LSN#######LSN############## ################LSN #######WAL## # #######WAL################## ##########################WAL################## ###########################

############pg_combinebackup######### ###########################pg_combinebackup########### pg_combinebackup limitations#########

########################WAL##############WAL########### ###25.3.5####### #########.#############.########################## ##################### ##pg_combinebackup################################ PostgreSQL############### ## ################### ################

###################make## #### ## ############### ##

################################

25.3.4. ####API##################

#####[62.469734]######API##################pg_basebackup ##################### ####### ## #### ##pg_basebackup##API################ ## ########pg_basebackup############################## ########################### ##########################

############################API################pg_basebackup################## ###

1. WAL##############################

2. pg_backup_start## #######EXECUTE##########################

   ```
   SELECT pg_backup_start(label => 'label', fast => false);
   ```

   ###label################################## pg_backup_start################ ################### ########################

   ################################### ########pg_backup_start################ ##################### ##########################checkpoint_timeout## #checkpoint_completion_target########### ################################### ## #####################pg_backup_start#2###########true################ ######## ####################I/O##################

3. #pg_dump#pg_dumpall#####tar#cpio## #### ## ################## #25.3.4.1#########

4. ########################

   ```
   SELECT * FROM pg_backup_stop(wait_for_archive => true);
   ```

   ##########################WAL################# ########WAL############### ######## #######################pg_switch_wal############## ############# ###############WAL######################

   pg_backup_stop#3######1####### 2#######################backup_label#### ######################### 3############tablespace_map################## ######### ##############################1################################ #############

5. ############WAL######################## pg_backup_stop#####1########### ##################################### ########archive_mode###### #wait_for_archive#####true#####pg_backup_stop ##################### ### ########pg_backup_stop###############archive_mode#always######## ## ###archive_command####archive_library############################## ###### ################### ## ##### ## ####### pg_backup_stop####################statement_timeout########### #########pg_backup_stop##

##############WAL##
##wait_for_archive#####(######true##)#false###############WAL#########
#pg_backup_stop############### #######pg_backup_stop#####WAL########
###################### ###################### WAL#########################
######WAL################### ################

25.3.4.1.

################
################################### ################################ ####rsync#
###
###GNU tar###########tar##
########GNU tar######1.16############################1############2######
GNU# tar#1.23####################--warning=no-file-changed --
warning=no-file-removed##

##########################/usr/local/pgsql/data#########################
######################
######################### #########################

########pg_wal/################################ ##########################
################## pg_wal/## #
#################### ################postmaster###########postmaster########
#postmaster.pid#postmaster.opts######## (#########pg_ctl################)

##pg_replslot/##############
################ ####################################WAL###################
####################################WAL######## ############################
##################################
################################### ##
######################################

#####
#pg_dynshmem/#pg_notify/#pg_serial/#pg_snapshots/#pg_stat_tmp/
#pg_subtrans/## ######postmaster#######
######

pgsql_tmp################################## ########postmaster################
################

pg_internal.init## ####################
#######################

################pg_backup_start###########pg_backup_start###########WAL
############## ##
######### ####################pg_tblspc/#################################
######################################

############################## ##############
##pg_backup_start#pg_backup_stop############# ################WAL######
############################## ############################

25.3.5.

#################################### #############

1. ########################

2. ## ########
   ########2#################### ####################pg_wal##############
   ##### ##########################WAL####################

3. ##

4. #####[75.260417]################################## ############root########
   ################################## ################pg_tblspc/#############
   ######################### ##
   ############ ##
   #### ################pg_tblspc/####################################

5. ##
   ################################## ##
   ################### ########pg_combinebackup##############################
   ##################################### ################################

6. pg_wal/################### ##
   ####### pg_wal/################################## ########################
   ########################

7. ##2############WAL#################pg_wal/########## ##################
   ##

8. postgresql.conf#19.5.5################################
   #recovery.signal########## #######pg_hba.conf##########################
   ####################

9. ############ ######################WAL############### ##################
   ########################## ##################################
   #recovery.signal###### ####################

10. ## #############1######## ############
    ########pg_hba.conf#############

######################
#WAL##################PostgreSQL#####restore_command### archive_command#
################# ########WAL############%f#WAL###################%p######
######%###############%%#
######### ##########################

```
restore_command = 'cp /mnt/server/archivedir/%f %p'
```

##############WAL######/mnt/server/archivedir############## ##############
############ ##

################################## ##
############## ############### #######################(###############SIGTERM#
#)###########(##############)######################

##########WAL################## .history############################## ##
##%p###########%f############### ###########################

################WAL######pg_wal/######## ################################ #
######################pg_wal/###################

###########WAL#################### ###########################WAL########
#################### ###########file not found############# ##############
#restore_command######## ###########00000001.history####################
####### ############################## #######25.3.6#########

#
######recovery target###ID##
########## #########################ID#######################################
#############

##

####################pg_backup_stop#################### ##############
################################## ######################################
###############

####WAL################################## ################################
#################################### ############################WAL######
################################## ######################### ###############
######################### ##WAL#####
#pg_control##########

25.3.6.

##SF#################### ##########
###########5:15PM#################################### ######################
#5:14PM######################## ############################### ###############
###########################
##WAL##################################### ################################
#######WAL########################WAL################

###########PostgreSQL################## ##############################WAL###
######################### #####ID###WAL#################################
##########WAL########### #####WAL#####00000001000001234000055CD#####
#00000001#16#########ID### ####################################ID###10######
#############

############################## ################################## #############
###################### ##
######### ################################### ##############################
#######################

#################PostgreSQL## ######
######################################WAL########################### #######
#####WAL###########WAL################ ################################
############################## ##
################### ##

#################################### ######################################
##current#recovery_target_timeline######
###ID############## ######################################

25.3.7.

##############################

25.3.7.1.

#######################PostgreSQL###################### #######################
################pg_dump################################ ####pg_dump##########
################################

pg_basebackup############# ####-X####

25.3.7.2.

##gzip#############

```
archive_command = 'gzip < %p > /mnt/server/archivedir/%f.gz'
```

gunzip##########

```
restore_command = 'gunzip < /mnt/server/archivedir/%f.gz > %p'
```

25.3.7.3. archive_command#####

postgresql.conf#######################archive_command##################

```
archive_command = 'local_backup_script.sh "%p" "%f"'
```

######bash###perl#####################

##########################

- ##########################
- ######################WAL########
- ##############################
- ######################

> ###
>
> archive_command##########logging_collector#################### ######## stderr#################################### ################################ #####

25.3.8.

##################################### ##################

- ######################CREATE DATABASE################################ #CREATE DATABASE#######template## ########## ##################### ##############################template########## #########
- CREATE TABLESPACE###############WAL#################################### ### ######WAL########################### WAL############################## ######### ################################ ################################### ########################

#########WAL## #### ##################################### ################################### ################################### ## ##### ####full_page_writes#######################################WAL##############

############28################## ################# PITR#####WAL############## #### ######full_page_writes#################################WAL####### ######### ##WAL#################### ########

#26#

################### ##
##################### ############################ ##########################
################################### ################################### #######
############################
#1######## ##
##############

##################################
################ ##

#########1################################### ##############################
############ ### ######################
######################### ##

################### ### ###
################################### ##
########### ##
############# ################################### ############################
###

###
#######

########################### ######################### ##########################
##

##

26.1.

##################

########### 1## ############
################## ##
########### ################################

################################
POSIX################## # 18.2.2.1######### ################################
############################## ##
####

##########################

############################
################################ ##
###
Linux####DRBD################################

#############

##################################WAL######################## ##############
############
############################

#########################26.2######################26.2.5#####################
26.4

##########

PostgreSQL###########
##WAL######################## ## #
#######
################29######### ####################(#47#)######################
#####

#########################

#################
#################################### ####################################
######################## ##
#####

#################Slony-I################################### ################
##

SQL###################

SQL################################SQL###########1################## ######
############# ## ##########
############################1####################

###################random()########CURRENT_TIMESTAMP#################
################### ######################################SQL############
##
####2######PREPARE
TRANSACTION###COMMIT PREPARED############### Pgpool-II#Continuent
Tungsten#################

################

###
##
Bucardo###################

###############

####################################
########### ## ################
##
######################### ##1######
######################random()###################

PostgreSQL ######################## ####PostgreSQL # 2######PREPARE
TRANSACTION###COMMIT PREPARED###
###

26.1######################

#26.1

##	######	######### ######### ###	######### ######	######### ###	###### ######### ###	**SQL### ######### ######### ###**	######### ######### ###	###### ######### ###
#####	NAS	DRBD	###### ###### #######	######Londiste#Slony ####### ##pglogical		pgpool-II	Bucardo	

##	######	######### ######### ###	######### ######	######### ###	###### ######### ###	**SQL### ######### ######### ###**	######### ######### ###	###### ######### ###
####	######	#### ####	WAL	##### ######	#####	SQL	#####	###### ######
######## ####		#	#	#	#	#	#	#
######## ######				#		#	#	#
######## ######## ###	#		#	#		#		
######## ###	#		#### ####	#### ####	#		#	
######## ######## ###	#	#	#### ####	#### ####		#		#
######## ######## #####			##### ######	#	#	#	#	#
######## #				#	#		#	#
######## ####	#	#	#		#	#		#

##################

######

> ################################## #################1########## ################ ### ################################## #################### ## ##

##################

> ##################################### ## ## ################################ ## ### ###PL/Proxy##################

###PostgreSQL##########################PostgreSQL################################ ##################################### ####################

26.2.

###1##################(HA)######## ########## ####################################

#######################WAL########################## ######################### ######## ## #########################

################WAL############################ PostgreSQL################## ##### ###WAL#########1######(WAL#####)################ WAL####(16MB)####

####################
################### ##WAL###########
#26.2.5####

############################ ####WAL########################### #########
##############
################archive_timeout################### #######################
############################### ################26.2.5###########################
######

####################
############# ##
######################### ################################ ##################
#######26.4##########

26.2.1.

##############
##
####### CREATE TABLESPACE##
################################## ##
####2##################2################## ##################################
###32########64####################

###################PostgreSQL############### ############################
PostgreSQL###
########
######################WAL#########
####################

26.2.2.

###################standby.signal#######################

################################WAL###### ########WAL#####(restore_command#
#)#######TCP##(##############)##############WAL############ ################
#####pg_wal############WAL############# ##################################
#WAL############ ########################pg_wal#############

############restore_command#########################WAL############### ##
######WAL######restore_command######pg_wal################WAL########
##pg_wal######
###########WAL############## ##
#################################### #######pg_wal##########################
##########################

#########pg_ctl promote###########pg_promote()##########################
###################pg_wal#########WAL##############################

26.2.3.

25.3## ############
################################### ##
##################

################################(###)###
##replication######pg_hba.conf#######

######################max_wal_senders#########################
##############max_replication_slots####################

25.3.2##

26.2.4.

###(25.3.5##)# #################
###standby.signal############# WAL#######################restore_command##
######## ################################recovery_target_timeline#latest##
###

> ##
>
> ##############restore_command##################################

##############################(###IP####)#########################libpq#####
#primary_conninfo######### ######################primary_conninfo#################
###

###
WAL######################

WAL###############archive_cleanup_command#####################################
################ ##pg_archivecleanup######################pg_archivecleanup######
#archive_cleanup_command################### ##############################
###
####

##############

```
primary_conninfo = 'host=192.168.1.50 port=5432 user=foo password=foopass
 options=''-c wal_sender_timeout=5000'''
restore_command = 'cp /path/to/archive/%f %p'
archive_cleanup_command = 'pg_archivecleanup /path/to/archive %r'
```

###
#max_wal_senders##############

26.2.5.

#######WAL#
##########WAL#################WAL#########################

##########################(26.2.8##) ##
######################## ##
#########1###### ##############################archive_timeout########

###WAL################
wal_keep_size##############
#######################WAL##################################WAL##############
#########################WAL##################################

#########################26.2## ########
###primary_conninfo###########
############### ##################replication##############(26.2.5.1##)########
##listen_addresses########(pg_hba.conf##)#########

##############################tcp_keepalives_idle#tcp_keepalives_interval##
#tcp_keepalives_count#############################

################################max_wal_senders#####

#########primary_conninfo##########################WAL################
######## #################walreceiver##############walsender#######

26.2.5.1.

##########WAL## WAL######
############### #######################REPLICATION#######################
########## ###########REPLICATION## # LOGIN#######################
REPLICATION##############SUPERUSER############################

##################pg_hba.conf####*database*######replication##############
#########IP####192.168.1.100##########################foo##############
##pg_hba.conf################

```
# ### foo #### 192.168.1.100 ################################
# #########################
#
# TYPE  DATABASE        USER            ADDRESS                 METHOD
host    replication     foo             192.168.1.100/32        md5
```

#####################################primary_conninfo####### ##############
#~/.pgpass############*database*######replication####### #################
IP #####192.168.1.50######5432##################foo#########foopass##
################postgresql.conf##############

```
# ######### 192.168.1.50 ##### 5432#######
# ##### foo ####### foopass ###
primary_conninfo = 'host=192.168.1.50 port=5432 user=foo password=foopass'
```

26.2.5.2.

###WAL######## ####
#######WAL##################WAL####################### ################
#pg_current_wal_lsn###########pg_last_wal_receive_lsn##############
9.95#### 9.96##### ##############ps######## WAL#####################27.1##
###

pg_stat_replication######WAL#######################
pg_current_wal_lsn####sent_lsn##
############sent_lsn#pg_last_wal_receive_lsn#####################
#######################

###########WAL##########pg_stat_wal_receiver##################
pg_last_wal_replay_lsn####flushed_lsn##########WAL###########WAL###
##############

26.2.6.

#####[91.746032]#################################### ########WAL###########
########WAL###
###

###########################wal_keep_size#######archive_command#archive_library#####
#######################WAL################## ##############################
#######################WAL##################

#####[73.046875]#####hot_standby_feedback#######################################
#################################### ######################### ##############
############hot_standby_feedback##
######################

> ##
>
> ############################WAL#########pg_wal#####################
> ################ max_slot_wal_keep_size##################WAL############
> #########

26.2.6.1.

###

#################pg_replication_slots ###########

################################(53.4##)####SQL##(9.28.6##)###############

26.2.6.2.

############################

```
postgres=# SELECT * FROM
 pg_create_physical_replication_slot('node_a_slot');
  slot_name  | lsn
-------------+-----
 node_a_slot |

postgres=# SELECT slot_name, slot_type, active FROM pg_replication_slots;
  slot_name  | slot_type | active
-------------+-----------+--------
 node_a_slot | physical  | f
(1 row)
```

####################################primary_slot_name############# ########
####

```
primary_conninfo = 'host=192.168.1.50 port=5432 user=foo password=foopass'
primary_slot_name = 'node_a_slot'
```

26.2.7.

###WAL##############
##

#######################
#################################### ######################################
##############1##############1##############

######################WAL######################WAL########### ##############
#######################################WAL ##############

#####################################26.2.8################################

##

################################recovery_target_timeline#'latest'########
##

###
(max_wal_senders#hot_standby#### #################) ################################
########primary_conninfo############

26.2.8.

PostgreSQL############################ ######################################
################################ ##################################

######
################################ ####################2-safe###########
#synchronous_commit#remote_write###########group-1-safe (group-safe#1-safe) ####
###

##
############################## ##
########
##
##############################

####################
#################################### ##
####### ##

####################
##########################WAL########################## ####################
###########pg_receivewal#pg_recvlogical################################
##

26.2.8.1.

###
##synchronous_standby_names################ ##synchronous_commit#on###########
#####################################19.5.1 ###19.6.2########### ################
synchronous_commit####
################### ##
#####################################

############################WAL################# ######
#wal_receiver_status_interval#######################WAL################
############## synchronous_commit#remote_apply##########################
################################ #############synchronous_standby_names###
##
############################## ##
####
##

synchronous_commit#remote_write##
####on#########
############### ##PostgreSQL#
#################### ## #############
###

synchronous_commit#remote_apply##
######################### ############################

################################## ##############################WAL########
##################################

26.2.8.2.

################################## ##
##############################synchronous_standby_names#######
##FIRST#ANY#######

##FIRST##WAL######################
############## ## ##################
######################## ##

##################synchronous_standby_names#######

```
synchronous_standby_names = 'FIRST 2 (s1, s2, s3)'
```

########4#########s1#s2#s3#s4######s1#s2############## ###################
######### s3#############s1####s2######################### ###############
##s4###########

##ANY##WAL##################
#################

###################synchronous_standby_names#######

```
synchronous_standby_names = 'ANY 2 (s1, s2, s3)'
```

########4#########s1#s2#s3#s4####################s1#s2#s3################
########## #################s4###########

pg_stat_replication##############################

26.2.8.3.

###
###
###

PostgreSQL##
###

###########################10%##
############90% ############

###
############## ##
#####

#####WAL##

26.2.8.4.

synchronous_commit##on#remote_apply#remote_write############
##synchronous_standby_names######################################
###

##
##synchronous_standby_names############################## ##############

############################ ##
####

##########################
############################

#######################
#########################

#################################### ###catchup######### #################
#########streaming####### ###catchup############################ ##########
############################### #######streaming########################### #
#####pg_stat_replication#############

###
###########################WAL##############################
PostgreSQL##WAL####
###

###
##synchronous_standby_names##################### ########################
#######

##

#####[80.758017]#################################pg_backup_start()###
pg_backup_stop()#########synchronous_commit = off#################### #####
########################## ##make####pg_
######_####()#pg_######_stop()####_####=################## ############
#################### #####

26.2.9.

#########WAL##################2############# WAL###########################
########WAL############# ########WAL#############archive_mode#always##
#######################WAL############################WAL##########WAL#
################################   ######################archive_command##
#archive_library###
archive_command#archive_library#####################################2#
############################## ###2###
#######

archive_mode#on### ###################
################### #############WAL######################### #####WAL####
################WAL#################WAL######################### ########
###################### ##################################### ################
################ ##################on#always################

26.3.

##

############################## ##
############# ##

###
################ ###STONITH (Shoot the Other Node In The Head)############# ########
###

###########witness#################################### #####

PostgreSQL## ######### #####IP###

################################1######### ############# ###################### ################################ ################################ ############ ## pg_rewind###################### ####### ##################################### ################################ ##

#############

##########################47.2.3## ## ####29.3#######################

################################pg_ctl promote#######pg_promote()####### ## ######### ##

26.4.

###
#################### ##
##
####################

###

26.4.1.

#########hot_standby### ###### ################# #######################

#################################### ################################ ######## ## ############################ ####### ## ############# ## ## #####13.2##########

##

- ###########: SELECT###COPY TO
- ########: DECLARE#FETCH#CLOSE
- #####: SHOW#SET#RESET
- ##############:
  - BEGIN#END#ABORT#START TRANSACTION
  - SAVEPOINT#RELEASE#ROLLBACK TO SAVEPOINT
  - EXCEPTION######################

- LOCK TABLE# ###################### ACCESS SHARE###ROW SHARE###ROW EXCLUSIVE
- #####: PREPARE#EXECUTE#DEALLOCATE#DISCARD
- ########: LOAD
- UNLISTEN

################################ID########## ############################## ### ######################

- ########DML#: INSERT#UPDATE#DELETE#MERGE#COPY FROM###TRUNCATE# ######## ##################################### ################################ID##### ##
- ########DDL#: CREATE#DROP#ALTER###COMMENT# ###################### ######## ###########################
- SELECT ... FOR SHARE | UPDATE# #####################################
- ################SELECT#####
- ROW EXCLUSIVE MODE###############LOCK
- #########LOCK# ###ACCESS EXCLUSIVE MODE#########
- ################################
  - BEGIN READ WRITE#START TRANSACTION READ WRITE
  - SET TRANSACTION READ WRITE#SET SESSION CHARACTERISTICS AS TRANSACTION READ WRITE
  - SET transaction_read_only = off
- #########: PREPARE TRANSACTION#COMMIT PREPARED###ROLLBACK PREPARED# ##############################1######WAL##############
- #########: nextval()#setval()
- LISTEN#NOTIFY

#########################LISTEN#NOTIFY############ ######################### ###################### ############################

###########transaction_read_only######################## ################ ## ############################## ################### ########################## ############################## ######################################

####SHOW in_hot_standby#################################### #########14## ###in_hot_standby################################SHOW transaction_read_only#### ################################(# 9.96)###### ##### ################################ ## ##########################

26.4.2.

################################ ########################## ################## ########### ################### ##WAL ################################I/O################

#################################### ## ### ################################ # ################

- ### ####LOCK######## DDL#######
- ##
- ##
- WAL##1############################## ################
- WAL###
  ##

############################ ################################## ############## ######## WAL## ########WAL# ### ################################### #### #########WAL##

##DROP TABLE############## ##### #DROP TABLE########################## ##################################DROP TABLE######### ####DROP TABLE## ################################## ####################WAL################### ########## ########WAL##################################DROP TABLE######## ####################################

#######################WAL################################## ####WAL######## ############### ############max_standby_archive_delay#max_standby_streaming_delay## ########### ####WAL################## ########################WAL######## ################################## #######WAL## ##2##############

###
#################################### ##
########## ###############WAL##
##############################

max_standby_archive_delay###max_standby_streaming_delay##############
################### ##########################DROP DATABASE################## ############## ##
###################

################WAL#####
##

################WAL################################ ########################
################################### ##
######################################

#############WAL#################################### ##PostgreSQL#MVCC#####
####
######################################
##############################

################################
#################MVCC#################### ######VACUUM####################
1## ####################
#################VACUUM####################

##
######max_standby_archive_delay###max_standby_streaming_delay####
#statement_timeout#################

###
##hot_standby_feedback############## ###VACUUM#######################
####################### ##
################## ##
################ ############################ ######################
##hot_standby_feedback##################################### ##
##max_standby_archive_delay############WAL#############################
###
#max_standby_streaming_delay##################

############################pg_stat_database_conflicts###################
##pg_stat_database######################

#########WAL###deadlock_timeout#####################################
log_recovery_conflict_waits###########

26.4.3.

postgresql.conf####hot_standby#on###############standby.signal########
#################### ##
##################################
##############################

```
LOG:  entering standby mode


... ######### ...

LOG:  consistent recovery state reached
LOG:  database system is ready to accept read-only connections
```

################################## ######wal_level#replica####logical######
#############WAL################################ ##########################
############

- ###########64################
- #####################

###############(###########)###############WAL################
#archive_timeout################

#########################ID## ###########
############
###
############### ################

- max_connections
- max_prepared_transactions
- max_locks_per_transaction
- max_wal_senders
- max_worker_processes

###################
################################### ##
######
###

######WAL################### ######################################WAL#######
################## #######


```
WARNING:  hot standby is not possible because of insufficient parameter
 settings
DETAIL:  max_connections = 80 is a lower setting than on the primary
 server, where its value was 100.
LOG:  recovery has paused
DETAIL:  If recovery is unpaused, the server will shut down.
HINT:  You can then restart the server after making the necessary
 configuration changes.
```

############################
##

max_standby_archive_delay###max_standby_streaming_delay########################## ##
###################### ## #####
###################### ##
###########################-1#######################

####################################WAL######### ##############################
######### ## #####
####################relcache############## ####################################
######## ############################dblink##############################PL###
################################

##############################

- #######: ###CREATE INDEX
- ########: GRANT#REVOKE#REASSIGN
- ######: ANALYZE#VACUUM#CLUSTER#REINDEX

###

####################
################## ####################

pg_cancel_backend()#pg_terminate_backend()############################
##################   pg_stat_activity##############################   ##
#########pg_prepared_xacts#########   ####################################
#pg_prepared_xacts###

pg_locks##########################   pg_locks##############################
#AccessExclusiveLocks#################################### ################
################ ##############AccessExclusiveLocks#######pg_locks########
#######################

################Nagios#####check_pgsql#######   ########################
##check_postgres############### ## ######
################################

######WAL############## ####pg_backup_start#pg_switch_wal#####

pg_stat_statements########################

##########WAL########## ######WAL### ### ###

############## ##
####

- ############################AccessExclusiveLocks######2################
  ##### ##AccessExclusiveLocks################################
  #AccessExclusiveLocks###############1##############
  ##max_locks_per_transaction######################################
  max_prepared_transactions####0######################

- ## #13.2.3###13.4.1### ################
  ##

#27#

################################## ####################

################################## #######PostgreSQL#################
##ps#top#iostat#vmstat#####Unix################### ####################
##########PostgreSQL#EXPLAIN#################### 14.1####################
##EXPLAIN##################

27.1. ####Unix###

#####UNIX##########PostgreSQL#######################ps####################
###########

```
$ ps auxww | grep ^postgres
postgres  15551  0.0  0.1  57536  7132 pts/0    S    18:02   0:00 postgres
 -i
postgres  15554  0.0  0.0  57536  1184 ?        Ss   18:02   0:00
 postgres: background writer
postgres  15555  0.0  0.0  57536   916 ?        Ss   18:02   0:00
 postgres: checkpointer
postgres  15556  0.0  0.0  57536   916 ?        Ss   18:02   0:00
 postgres: walwriter
postgres  15557  0.0  0.0  58504  2244 ?        Ss   18:02   0:00
 postgres: autovacuum launcher
postgres  15582  0.0  0.0  58772  3080 ?        Ss   18:04   0:00
 postgres: joe runbug 127.0.0.1 idle
postgres  15606  0.0  0.0  58772  3052 ?        Ss   18:07   0:00
 postgres: tgl regression [local] SELECT waiting
postgres  15610  0.0  0.0  58772  3056 ?        Ss   18:07   0:00
 postgres: tgl regression [local] idle in transaction
```

#ps########################## ###################### #######Linux########### ##
######################### ################################ ##4################
############################## ##
###########1########################

postgres: *user database host activity*

###########(######)## ####
##idle#######################idle in transaction#BEGIN################
########SELECT#################### ##
##waiting######## #########15606#####15610##################################
########################15610################# ##########pg_locks#######
##

cluster_name##########ps################

```
$ psql -c 'SHOW cluster_name'
 cluster_name
--------------
 server1
(1 row)

$ ps aux|grep server1
postgres   27093  0.0  0.0  30096  2752 ?        Ss   11:34   0:00
 postgres: server1: background writer
...
```

update_process_title########################## ################################ #
##

###

Solaris############## /bin/ps#####/usr/ucb/ps############# ###w####1### ##2############# ######postgres########ps############################ ############# ##3####################ps#######postgres################ ####

27.2.

PostgreSQL#################################### ############################## ######################### ## ##### ######################################

###PostgreSQL## ############################### ######################

27.2.1.

#postgresql.conf######################### ####################19############

track_activities##

track_counts##

track_functions##################################

track_io_timing##########################fsync##################

track_wal_io_timing#######WAL########fsync##################

############################postgresql.conf######## ####SET############## ################ ##SET################## ####

################### ####PostgreSQL################################### ######## ################################pg_stat############### ################### ###### ## ########################

27.2.2.

#################################### 27.1######### ############################ ######## 27.2######### #######27.2.26#####################################

##########################PGSTAT_MIN_INTERVAL####1########################## ##########1### ## #### ####track_activities########################

##1## ################# ## ################ ## ########## #### ## ####### ## # #########stats_fetch_consistency#snapshot####### ########################

##stats_fetch_consistency#none############## You can invoke pg_stat_clear_snapshot() to discard the current transaction's statistics snapshot or cached values (if any). The next use of statistical information will (when in snapshot mode) cause a new snapshot to be built or (when in cache mode) accessed statistics to be cached.

############## #pg_stat_xact_all_tables#pg_stat_xact_sys_tables#pg_stat_xact_user_tables# ###pg_stat_xact_user_functions## ###### ##

27.1## ################################## ################## ##################NULL######## ############################ ################################## ##############pg_read_all_stats######## #21.5##############################

#27.1

####	##
pg_stat_activity	#########1################### ######################## ###### #pg_stat_activity#########
pg_stat_replication	WAL#######1################### ########################## ###### #pg_stat_replication#########
pg_stat_wal_receiver	1####################WAL ################### ###### #pg_stat_wal_receiver#########
pg_stat_recovery_prefetch	1#################### ################# ###### #pg_stat_recovery_prefetch ###### ###
pg_stat_subscription	1##################1######## ###################### ###### #pg_stat_subscription#########
pg_stat_ssl	###################1############SSL# ######## #######pg_stat_ssl####### ##
pg_stat_gssapi	###################1########### #GSSAPI################## ###### #pg_stat_gssapi#########
pg_stat_progress_analyze	ANALYZE############################ ######1################# 27.4.1###### ###
pg_stat_progress_create_index	CREATE INDEX###REINDEX############ ####1################# 27.4.4#########
pg_stat_progress_vacuum	VACUUM############################ #####1################# 27.4.5####### ##
pg_stat_progress_cluster	CLUSTER###VACUUM FULL############# ###1################# 27.4.2#########
pg_stat_progress_basebackup	###################WAL########1#### ############# 27.4.6#########

####	##
pg_stat_progress_copy	COPY################1################ ## 27.4.3#########

#27.2

####	##
pg_stat_archiver	WAL######################1######### #######pg_stat_archiver#########
pg_stat_bgwriter	############################1######## ## #######pg_stat_bgwriter#########
pg_stat_checkpointer	One row only, showing statistics about the checkpointer process's activity. See pg_stat_checkpointer for details.
pg_stat_database	########1######################### ### ####pg_stat_database#########
pg_stat_database_conflicts	########1############################## ################################### ## #####pg_stat_database_conflicts### #######
pg_stat_io	################################ ####1#########I/O######### ###### #pg_stat_io#########
pg_stat_replication_slots	##############1############## ###################### ####### pg_stat_replication_slots######## ##
pg_stat_slru	SLRU###1################### ###### #pg_stat_slru#########
pg_stat_subscription_stats	###########1##################### ### ####pg_stat_subscription_stats### #######
pg_stat_wal	WAL#############1######### ###### #pg_stat_wal#########
pg_stat_all_tables	##################1########### ##################### ###### #pg_stat_all_tables#########
pg_stat_sys_tables	################## ##pg_stat_all_tables######
pg_stat_user_tables	################## ##pg_stat_all_tables######
pg_stat_xact_all_tables	pg_stat_all_tables############### ######################(############### ###pg_stat_all_tables############# ####) ################################ ##########
pg_stat_xact_sys_tables	################## ##pg_stat_xact_all_tables######
pg_stat_xact_user_tables	################## ##pg_stat_xact_all_tables######

####	##
pg_stat_all_indexes	##################1########## ######################## ###### #pg_stat_all_indexes#########
pg_stat_sys_indexes	######################## ##pg_stat_all_indexes######
pg_stat_user_indexes	######################## ##pg_stat_all_indexes######
pg_stat_user_functions	#########1########################## ###### ######pg_stat_user_functions###### ####
pg_stat_xact_user_functions	pg_stat_user_functions########### ########################### (######## ########pg_stat_user_functions### ######)
pg_statio_all_tables	#################1########## ######I/O############# ###### #pg_statio_all_tables#########
pg_statio_sys_tables	################## ##pg_statio_all_tables######
pg_statio_user_tables	################## ##pg_statio_all_tables######
pg_statio_all_indexes	###################1########## #######I/O############## ###### #pg_statio_all_indexes#########
pg_statio_sys_indexes	######################## ##pg_statio_all_indexes ######
pg_statio_user_indexes	######################## ##pg_statio_all_indexes######
pg_statio_all_sequences	##################1########## ######I/O############## ###### #pg_statio_all_sequences#########
pg_statio_sys_sequences	################### ##pg_statio_all_sequences##### #### ###################################
pg_statio_user_sequences	################## ##pg_statio_all_sequences######

###

pg_stat_io#pg_statio_################################## ################## ######## PostgreSQL#I/O####I/O## ## ############I/O################### ###PostgreSQL##

27.2.3. pg_stat_activity

pg_stat_activity##################################1#######

#27.3 pg_stat_activity###

# #	##
datid oid	#################OID###
datname name	######################
pid integer	##########ID###
leader_pid integer	######################################ID### ######################### ##############ID### NULL## #########################
usesysid oid	################OID###
usename name	######################
application_name text	#########################
client_addr inet	##################IP####### ########NULL######################Unix### #####################################
client_hostname text	client_addr#DNS############################## IP#####log_hostname##### ##############NULL######
client_port integer	######################TCP#########Unix###########-1### #######NULL ###########################
backend_start timestamp with time zone	############# #####################################
xact_start timestamp with time zone	##NULL### ################## ###########query_start######
query_start timestamp with time zone	#######################state#active#######################
state_change timestamp with time zone	state#########
wait_event_type text	##############################NULL###### # 27.4#########
wait_event text	################################NULL###### # 27.5### 27.13###########
state text	################## ################### • active: ################## • idle: ############################## • idle in transaction: ###

##
• idle in transaction (aborted): #####idle in transaction###### ################################ • fastpath function call: ################# • disabled: ###############track_activities################
backend_xid xid ################################ 66.1#########
backend_xmin xid #########xmin###
query_id bigint #################### state#active########################### ###### ################################## ######################## ##compute_query_id## #######null######
query text ######################## state#active########################### ### ############################ ##################1024################### ###track_activity_query_size#########
backend_type text ################### ######autovacuum launcher#autovacuum worker#logical replication launcher#logical replication worker#parallel worker#background writer#client backend#checkpointer#archiver#standalone backend#startup#walreceiver#walsender#walwriter#walsummarizer# ### ##

##

wait_event#state######### #######active##################waiting### ################## ###active####wait_event#NULL#################### ########################

#27.4

#######	##
Activity	################ ################### ################### wait_event###### ######### # 27.5#########
BufferPin	##################################### ##################################### ################################ # 27.6# ########
Client	##################################### ####### ############################ ########## wait_event############# ### 27.7#########
Extension	################################# # 27.8#########

#######	##
`InjectionPoint`	################################### ############# ####36.10.13######### ### #######################
`IO`	######################### `wait_event`################ 27.9###### ####
`IPC`	################################ `wait_event`################ 27.10##### #####
`Lock`	##################### ############# ##############################SQL ################ ################# ################################## ## `wait_event`################## # 27.11#########
`LWLock`	##################### ############### ##################### `wait_event`## ##################### ############### ################################### # 27.12#########
`Timeout`	############################ `wait_event`################ 27.13##### #####

#27.5 Wait Events of Type `Activity`

`Activity` Wait Event	**Description**
`ArchiverMain`	Waiting in main loop of archiver process.
`AutovacuumMain`	Waiting in main loop of autovacuum launcher process.
`BgwriterHibernate`	Waiting in background writer process, hibernating.
`BgwriterMain`	Waiting in main loop of background writer process.
`CheckpointerMain`	Waiting in main loop of checkpointer process.
`LogicalApplyMain`	Waiting in main loop of logical replication apply process.
`LogicalLauncherMain`	Waiting in main loop of logical replication launcher process.
`LogicalParallelApplyMain`	Waiting in main loop of logical replication parallel apply process.
`RecoveryWalStream`	Waiting in main loop of startup process for WAL to arrive, during streaming recovery.
`ReplicationSlotsyncMain`	Waiting in main loop of slot sync worker.
`ReplicationSlotsyncShutdown`	Waiting for slot sync worker to shut down.
`SysloggerMain`	Waiting in main loop of syslogger process.
`WalReceiverMain`	Waiting in main loop of WAL receiver process.
`WalSenderMain`	Waiting in main loop of WAL sender process.

Activity Wait Event	Description
WalSummarizerWal	Waiting in WAL summarizer for more WAL to be generated.
WalWriterMain	Waiting in main loop of WAL writer process.

#27.6 Wait Events of Type Bufferpin

BufferPin Wait Event	Description
BufferPin	Waiting to acquire an exclusive pin on a buffer.

#27.7 Wait Events of Type Client

Client Wait Event	Description
ClientRead	Waiting to read data from the client.
ClientWrite	Waiting to write data to the client.
GssOpenServer	Waiting to read data from the client while establishing a GSSAPI session.
LibpqwalreceiverConnect	Waiting in WAL receiver to establish connection to remote server.
LibpqwalreceiverReceive	Waiting in WAL receiver to receive data from remote server.
SslOpenServer	Waiting for SSL while attempting connection.
WaitForStandbyConfirmation	Waiting for WAL to be received and flushed by the physical standby.
WalSenderWaitForWal	Waiting for WAL to be flushed in WAL sender process.
WalSenderWriteData	Waiting for any activity when processing replies from WAL receiver in WAL sender process.

#27.8 Wait Events of Type Extension

Extension Wait Event	Description
Extension	Waiting in an extension.

#27.9 Wait Events of Type Io

IO Wait Event	Description
BasebackupRead	Waiting for base backup to read from a file.
BasebackupSync	Waiting for data written by a base backup to reach durable storage.
BasebackupWrite	Waiting for base backup to write to a file.
BuffileRead	Waiting for a read from a buffered file.
BuffileTruncate	Waiting for a buffered file to be truncated.
BuffileWrite	Waiting for a write to a buffered file.
ControlFileRead	Waiting for a read from the pg_control file.
ControlFileSync	Waiting for the pg_control file to reach durable storage.
ControlFileSyncUpdate	Waiting for an update to the pg_control file to reach durable storage.
ControlFileWrite	Waiting for a write to the pg_control file.

IO Wait Event	Description
ControlFileWriteUpdate	Waiting for a write to update the pg_control file.
CopyFileRead	Waiting for a read during a file copy operation.
CopyFileWrite	Waiting for a write during a file copy operation.
DataFileExtend	Waiting for a relation data file to be extended.
DataFileFlush	Waiting for a relation data file to reach durable storage.
DataFileImmediateSync	Waiting for an immediate synchronization of a relation data file to durable storage.
DataFilePrefetch	Waiting for an asynchronous prefetch from a relation data file.
DataFileRead	Waiting for a read from a relation data file.
DataFileSync	Waiting for changes to a relation data file to reach durable storage.
DataFileTruncate	Waiting for a relation data file to be truncated.
DataFileWrite	Waiting for a write to a relation data file.
DsmAllocate	Waiting for a dynamic shared memory segment to be allocated.
DsmFillZeroWrite	Waiting to fill a dynamic shared memory backing file with zeroes.
LockFileAddtodatadirRead	Waiting for a read while adding a line to the data directory lock file.
LockFileAddtodatadirSync	Waiting for data to reach durable storage while adding a line to the data directory lock file.
LockFileAddtodatadirWrite	Waiting for a write while adding a line to the data directory lock file.
LockFileCreateRead	Waiting to read while creating the data directory lock file.
LockFileCreateSync	Waiting for data to reach durable storage while creating the data directory lock file.
LockFileCreateWrite	Waiting for a write while creating the data directory lock file.
LockFileRecheckdatadirRead	Waiting for a read during recheck of the data directory lock file.
LogicalRewriteCheckpointSync	Waiting for logical rewrite mappings to reach durable storage during a checkpoint.
LogicalRewriteMappingSync	Waiting for mapping data to reach durable storage during a logical rewrite.
LogicalRewriteMappingWrite	Waiting for a write of mapping data during a logical rewrite.
LogicalRewriteSync	Waiting for logical rewrite mappings to reach durable storage.
LogicalRewriteTruncate	Waiting for truncate of mapping data during a logical rewrite.
LogicalRewriteWrite	Waiting for a write of logical rewrite mappings.
RelationMapRead	Waiting for a read of the relation map file.

IO Wait Event	Description
RelationMapReplace	Waiting for durable replacement of a relation map file.
RelationMapWrite	Waiting for a write to the relation map file.
ReorderBufferRead	Waiting for a read during reorder buffer management.
ReorderBufferWrite	Waiting for a write during reorder buffer management.
ReorderLogicalMappingRead	Waiting for a read of a logical mapping during reorder buffer management.
ReplicationSlotRead	Waiting for a read from a replication slot control file.
ReplicationSlotRestoreSync	Waiting for a replication slot control file to reach durable storage while restoring it to memory.
ReplicationSlotSync	Waiting for a replication slot control file to reach durable storage.
ReplicationSlotWrite	Waiting for a write to a replication slot control file.
SlruFlushSync	Waiting for SLRU data to reach durable storage during a checkpoint or database shutdown.
SlruRead	Waiting for a read of an SLRU page.
SlruSync	Waiting for SLRU data to reach durable storage following a page write.
SlruWrite	Waiting for a write of an SLRU page.
SnapbuildRead	Waiting for a read of a serialized historical catalog snapshot.
SnapbuildSync	Waiting for a serialized historical catalog snapshot to reach durable storage.
SnapbuildWrite	Waiting for a write of a serialized historical catalog snapshot.
TimelineHistoryFileSync	Waiting for a timeline history file received via streaming replication to reach durable storage.
TimelineHistoryFileWrite	Waiting for a write of a timeline history file received via streaming replication.
TimelineHistoryRead	Waiting for a read of a timeline history file.
TimelineHistorySync	Waiting for a newly created timeline history file to reach durable storage.
TimelineHistoryWrite	Waiting for a write of a newly created timeline history file.
TwophaseFileRead	Waiting for a read of a two phase state file.
TwophaseFileSync	Waiting for a two phase state file to reach durable storage.
TwophaseFileWrite	Waiting for a write of a two phase state file.
VersionFileSync	Waiting for the version file to reach durable storage while creating a database.
VersionFileWrite	Waiting for the version file to be written while creating a database.

IO Wait Event	Description
WalsenderTimelineHistoryRead	Waiting for a read from a timeline history file during a walsender timeline command.
WalBootstrapSync	Waiting for WAL to reach durable storage during bootstrapping.
WalBootstrapWrite	Waiting for a write of a WAL page during bootstrapping.
WalCopyRead	Waiting for a read when creating a new WAL segment by copying an existing one.
WalCopySync	Waiting for a new WAL segment created by copying an existing one to reach durable storage.
WalCopyWrite	Waiting for a write when creating a new WAL segment by copying an existing one.
WalInitSync	Waiting for a newly initialized WAL file to reach durable storage.
WalInitWrite	Waiting for a write while initializing a new WAL file.
WalRead	Waiting for a read from a WAL file.
WalSummaryRead	Waiting for a read from a WAL summary file.
WalSummaryWrite	Waiting for a write to a WAL summary file.
WalSync	Waiting for a WAL file to reach durable storage.
WalSyncMethodAssign	Waiting for data to reach durable storage while assigning a new WAL sync method.
WalWrite	Waiting for a write to a WAL file.

#27.10 Wait Events of Type Ipc

IPC Wait Event	Description
AppendReady	Waiting for subplan nodes of an Append plan node to be ready.
ArchiveCleanupCommand	Waiting for archive_cleanup_command to complete.
ArchiveCommand	Waiting for archive_command to complete.
BackendTermination	Waiting for the termination of another backend.
BackupWaitWalArchive	Waiting for WAL files required for a backup to be successfully archived.
BgworkerShutdown	Waiting for background worker to shut down.
BgworkerStartup	Waiting for background worker to start up.
BtreePage	Waiting for the page number needed to continue a parallel B-tree scan to become available.
BufferIo	Waiting for buffer I/O to complete.
CheckpointDelayComplete	Waiting for a backend that blocks a checkpoint from completing.
CheckpointDelayStart	Waiting for a backend that blocks a checkpoint from starting.
CheckpointDone	Waiting for a checkpoint to complete.
CheckpointStart	Waiting for a checkpoint to start.

IPC Wait Event	Description
ExecuteGather	Waiting for activity from a child process while executing a Gather plan node.
HashBatchAllocate	Waiting for an elected Parallel Hash participant to allocate a hash table.
HashBatchElect	Waiting to elect a Parallel Hash participant to allocate a hash table.
HashBatchLoad	Waiting for other Parallel Hash participants to finish loading a hash table.
HashBuildAllocate	Waiting for an elected Parallel Hash participant to allocate the initial hash table.
HashBuildElect	Waiting to elect a Parallel Hash participant to allocate the initial hash table.
HashBuildHashInner	Waiting for other Parallel Hash participants to finish hashing the inner relation.
HashBuildHashOuter	Waiting for other Parallel Hash participants to finish partitioning the outer relation.
HashGrowBatchesDecide	Waiting to elect a Parallel Hash participant to decide on future batch growth.
HashGrowBatchesElect	Waiting to elect a Parallel Hash participant to allocate more batches.
HashGrowBatchesFinish	Waiting for an elected Parallel Hash participant to decide on future batch growth.
HashGrowBatchesReallocate	Waiting for an elected Parallel Hash participant to allocate more batches.
HashGrowBatchesRepartition	Waiting for other Parallel Hash participants to finish repartitioning.
HashGrowBucketsElect	Waiting to elect a Parallel Hash participant to allocate more buckets.
HashGrowBucketsReallocate	Waiting for an elected Parallel Hash participant to finish allocating more buckets.
HashGrowBucketsReinsert	Waiting for other Parallel Hash participants to finish inserting tuples into new buckets.
LogicalApplySendData	Waiting for a logical replication leader apply process to send data to a parallel apply process.
LogicalParallelApplyStateChange	Waiting for a logical replication parallel apply process to change state.
LogicalSyncData	Waiting for a logical replication remote server to send data for initial table synchronization.
LogicalSyncStateChange	Waiting for a logical replication remote server to change state.
MessageQueueInternal	Waiting for another process to be attached to a shared message queue.
MessageQueuePutMessage	Waiting to write a protocol message to a shared message queue.
MessageQueueReceive	Waiting to receive bytes from a shared message queue.

IPC Wait Event	Description
MessageQueueSend	Waiting to send bytes to a shared message queue.
MultixactCreation	Waiting for a multixact creation to complete.
ParallelBitmapScan	Waiting for parallel bitmap scan to become initialized.
ParallelCreateIndexScan	Waiting for parallel CREATE INDEX workers to finish heap scan.
ParallelFinish	Waiting for parallel workers to finish computing.
ProcarrayGroupUpdate	Waiting for the group leader to clear the transaction ID at end of a parallel operation.
ProcSignalBarrier	Waiting for a barrier event to be processed by all backends.
Promote	Waiting for standby promotion.
RecoveryConflictSnapshot	Waiting for recovery conflict resolution for a vacuum cleanup.
RecoveryConflictTablespace	Waiting for recovery conflict resolution for dropping a tablespace.
RecoveryEndCommand	Waiting for recovery_end_command to complete.
RecoveryPause	Waiting for recovery to be resumed.
ReplicationOriginDrop	Waiting for a replication origin to become inactive so it can be dropped.
ReplicationSlotDrop	Waiting for a replication slot to become inactive so it can be dropped.
RestoreCommand	Waiting for restore_command to complete.
SafeSnapshot	Waiting to obtain a valid snapshot for a READ ONLY DEFERRABLE transaction.
SyncRep	Waiting for confirmation from a remote server during synchronous replication.
WalReceiverExit	Waiting for the WAL receiver to exit.
WalReceiverWaitStart	Waiting for startup process to send initial data for streaming replication.
WalSummaryReady	Waiting for a new WAL summary to be generated.
XactGroupUpdate	Waiting for the group leader to update transaction status at end of a parallel operation.

#27.11 Wait Events of Type Lock

Lock Wait Event	Description
advisory	Waiting to acquire an advisory user lock.
applytransaction	Waiting to acquire a lock on a remote transaction being applied by a logical replication subscriber.
extend	Waiting to extend a relation.

Lock Wait Event	Description
frozenid	Waiting to update pg_database.datfrozenxid and pg_database.datminmxid.
object	Waiting to acquire a lock on a non-relation database object.
page	Waiting to acquire a lock on a page of a relation.
relation	Waiting to acquire a lock on a relation.
spectoken	Waiting to acquire a speculative insertion lock.
transactionid	Waiting for a transaction to finish.
tuple	Waiting to acquire a lock on a tuple.
userlock	Waiting to acquire a user lock.
virtualxid	Waiting to acquire a virtual transaction ID lock; see 66.1.

#27.12 Wait Events of Type Lwlock

LWLock Wait Event	Description
AddinShmemInit	Waiting to manage an extension's space allocation in shared memory.
AutoFile	Waiting to update the postgresql.auto.conf file.
Autovacuum	Waiting to read or update the current state of autovacuum workers.
AutovacuumSchedule	Waiting to ensure that a table selected for autovacuum still needs vacuuming.
BackgroundWorker	Waiting to read or update background worker state.
BtreeVacuum	Waiting to read or update vacuum-related information for a B-tree index.
BufferContent	Waiting to access a data page in memory.
BufferMapping	Waiting to associate a data block with a buffer in the buffer pool.
CheckpointerComm	Waiting to manage fsync requests.
CommitTs	Waiting to read or update the last value set for a transaction commit timestamp.
CommitTsBuffer	Waiting for I/O on a commit timestamp SLRU buffer.
CommitTsSLRU	Waiting to access the commit timestamp SLRU cache.
ControlFile	Waiting to read or update the pg_control file or create a new WAL file.
DSMRegistry	Waiting to read or update the dynamic shared memory registry.
DSMRegistryDSA	Waiting to access dynamic shared memory registry's dynamic shared memory allocator.
DSMRegistryHash	Waiting to access dynamic shared memory registry's shared hash table.

LWLock Wait Event	Description
DynamicSharedMemoryControl	Waiting to read or update dynamic shared memory allocation information.
InjectionPoint	Waiting to read or update information related to injection points.
LockFastPath	Waiting to read or update a process' fast-path lock information.
LockManager	Waiting to read or update information about #heavyweight# locks.
LogicalRepLauncherDSA	Waiting to access logical replication launcher's dynamic shared memory allocator.
LogicalRepLauncherHash	Waiting to access logical replication launcher's shared hash table.
LogicalRepWorker	Waiting to read or update the state of logical replication workers.
MultiXactGen	Waiting to read or update shared multixact state.
MultiXactMemberBuffer	Waiting for I/O on a multixact member SLRU buffer.
MultiXactMemberSLRU	Waiting to access the multixact member SLRU cache.
MultiXactOffsetBuffer	Waiting for I/O on a multixact offset SLRU buffer.
MultiXactOffsetSLRU	Waiting to access the multixact offset SLRU cache.
MultiXactTruncation	Waiting to read or truncate multixact information.
NotifyBuffer	Waiting for I/O on a NOTIFY message SLRU buffer.
NotifyQueue	Waiting to read or update NOTIFY messages.
NotifyQueueTail	Waiting to update limit on NOTIFY message storage.
NotifySLRU	Waiting to access the NOTIFY message SLRU cache.
OidGen	Waiting to allocate a new OID.
ParallelAppend	Waiting to choose the next subplan during Parallel Append plan execution.
ParallelHashJoin	Waiting to synchronize workers during Parallel Hash Join plan execution.
ParallelQueryDSA	Waiting for parallel query dynamic shared memory allocation.
ParallelVacuumDSA	Waiting for parallel vacuum dynamic shared memory allocation.
PerSessionDSA	Waiting for parallel query dynamic shared memory allocation.
PerSessionRecordType	Waiting to access a parallel query's information about composite types.

LWLock Wait Event	Description
PerSessionRecordTypmod	Waiting to access a parallel query's information about type modifiers that identify anonymous record types.
PerXactPredicateList	Waiting to access the list of predicate locks held by the current serializable transaction during a parallel query.
PgStatsData	Waiting for shared memory stats data access.
PgStatsDSA	Waiting for stats dynamic shared memory allocator access.
PgStatsHash	Waiting for stats shared memory hash table access.
PredicateLockManager	Waiting to access predicate lock information used by serializable transactions.
ProcArray	Waiting to access the shared per-process data structures (typically, to get a snapshot or report a session's transaction ID).
RelationMapping	Waiting to read or update a pg_filenode.map file (used to track the filenode assignments of certain system catalogs).
RelCacheInit	Waiting to read or update a pg_internal.init relation cache initialization file.
ReplicationOrigin	Waiting to create, drop or use a replication origin.
ReplicationOriginState	Waiting to read or update the progress of one replication origin.
ReplicationSlotAllocation	Waiting to allocate or free a replication slot.
ReplicationSlotControl	Waiting to read or update replication slot state.
ReplicationSlotIO	Waiting for I/O on a replication slot.
SerialBuffer	Waiting for I/O on a serializable transaction conflict SLRU buffer.
SerialControl	Waiting to read or update shared pg_serial state.
SerializableFinishedList	Waiting to access the list of finished serializable transactions.
SerializablePredicateList	Waiting to access the list of predicate locks held by serializable transactions.
SerializableXactHash	Waiting to read or update information about serializable transactions.
SerialSLRU	Waiting to access the serializable transaction conflict SLRU cache.
SharedTidBitmap	Waiting to access a shared TID bitmap during a parallel bitmap index scan.
SharedTupleStore	Waiting to access a shared tuple store during parallel query.
ShmemIndex	Waiting to find or allocate space in shared memory.

LWLock Wait Event	Description
SInvalRead	Waiting to retrieve messages from the shared catalog invalidation queue.
SInvalWrite	Waiting to add a message to the shared catalog invalidation queue.
SubtransBuffer	Waiting for I/O on a sub-transaction SLRU buffer.
SubtransSLRU	Waiting to access the sub-transaction SLRU cache.
SyncRep	Waiting to read or update information about the state of synchronous replication.
SyncScan	Waiting to select the starting location of a synchronized table scan.
TablespaceCreate	Waiting to create or drop a tablespace.
TwoPhaseState	Waiting to read or update the state of prepared transactions.
WaitEventCustom	Waiting to read or update custom wait events information.
WALBufMapping	Waiting to replace a page in WAL buffers.
WALInsert	Waiting to insert WAL data into a memory buffer.
WALSummarizer	Waiting to read or update WAL summarization state.
WALWrite	Waiting for WAL buffers to be written to disk.
WrapLimitsVacuum	Waiting to update limits on transaction id and multixact consumption.
XactBuffer	Waiting for I/O on a transaction status SLRU buffer.
XactSLRU	Waiting to access the transaction status SLRU cache.
XactTruncation	Waiting to execute pg_xact_status or update the oldest transaction ID available to it.
XidGen	Waiting to allocate a new transaction ID.

#27.13 Wait Events of Type Timeout

Timeout Wait Event	Description
BaseBackupThrottle	Waiting during base backup when throttling activity.
CheckpointWriteDelay	Waiting between writes while performing a checkpoint.
PgSleep	Waiting due to a call to pg_sleep or a sibling function.
RecoveryApplyDelay	Waiting to apply WAL during recovery because of a delay setting.
RecoveryRetrieveRetryInterval	Waiting during recovery when WAL data is not available from any source (pg_wal, archive or stream).

Timeout Wait Event	Description
RegisterSyncRequest	Waiting while sending synchronization requests to the checkpointer, because the request queue is full.
SpinDelay	Waiting while acquiring a contended spinlock.
VacuumDelay	Waiting in a cost-based vacuum delay point.
VacuumTruncate	Waiting to acquire an exclusive lock to truncate off any empty pages at the end of a table vacuumed.
WalSummarizerError	Waiting after a WAL summarizer error.

#####[78.431373]######################### ###########################

```
SELECT pid, wait_event_type, wait_event FROM pg_stat_activity WHERE
 wait_event is NOT NULL;
 pid  | wait_event_type | wait_event
------+-----------------+------------
 2540 | Lock            | relation
 6644 | LWLock          | ProcArray
(2 rows)
```

```
SELECT a.pid, a.wait_event, w.description
  FROM pg_stat_activity a JOIN
       pg_wait_events w ON (a.wait_event_type = w.type AND
                            a.wait_event = w.name)
  WHERE a.wait_event is NOT NULL and a.state = 'active';
-[ RECORD 1 ]-------------------------------------------------------
------------
pid         | 686674
wait_event  | WALInitSync
description | Waiting for a newly initialized WAL file to reach durable
 storage
```

> **##**
>
> #####[55.882353]##### 27.12######LWLock######### ##########################
> ####################LWLock#########################extension##########
> ##### ############extension###########LWLock###### 27.8#### 27.12#####
> ########### ########extension###LWLock##################################
> extension-assigned##########extension###############

27.2.4. pg_stat_replication

pg_stat_replication######WAL########1#####################################
######### ########################## ########################

#27.14 pg_stat_replication###

##	
pid	integer WAL###########ID###
usesysid	oid WAL################OID###

##
usename name   WAL####################
application_name text   WAL#####################
client_addr inet   WAL################IP####### ########NULL##################Unix#####   #############
client_hostname text   client_addr#DNS############################ IP#####log_hostname#####   ################NULL######
client_port integer   ######WAL############TCP########Unix############-1###
backend_start timestamp with time zone   #################WAL##############
backend_xmin xid   hot_standby_feedback################xmin###
state text   WAL############# ##############   • startup: ##WAL################   • catchup: ##WAL###################################   • streaming: ##WAL##   • backup: ##WAL####################   • stopping: ##WAL################
sent_lsn pg_lsn   ##########################
write_lsn pg_lsn   ##
flush_lsn pg_lsn   ##
replay_lsn pg_lsn   ##
write_lag interval   ###WAL##   ############## ################################synchronous_commit#   ###remote_write##################################
flush_lag interval   ###WAL##   ########### ################################synchronous_commit###   #on##################################
replay_lag interval   ###WAL##   ######## ################################synchronous_commit###   #remote_apply##################################
sync_priority integer

##
#########
sync_state text ################# ############# • async: ################ • potential: ## • sync: ############### • quorum: ##############################
reply_time timestamp with time zone ################################

pg_stat_replication#################WAL################################ ######################## ## ############################## ############replay_lag#################### ################### ######################WAL################################ #############NULL######

############################ ## #######NULL###########

> ##
>
> ## ############## #WAL## ############### ####pg_stat_replication############################WAL############## ################## ## ########### ## #######NULL###### ## ######

27.2.5. pg_stat_replication_slots

pg_stat_replication_slots######################1######################### ##

#27.15 pg_stat_replication_slots###

##
slot_name text ###########################
spill_txns bigint WAL################################logical_decoding_work_mem### ######################## ###
spill_count bigint #######WAL##################################### ################### ##################################
spill_bytes bigint

##
##
######WAL## ########## ##################I/O####logical_decoding_work_mem#######
stream_txns bigint
######WAL########################## #logical_decoding_work_mem################################### #### ############################## (##########################)###### ##################
stream_countbigint
######WAL## ## ##
stream_bytesbigint
######WAL## ############### ####################logical_decoding_work_mem### ###########
total_txns bigint
##
total_bytesbigint
######WAL## ##### ###
stats_reset timestamp with time zone
##################

27.2.6. pg_stat_wal_receiver

pg_stat_wal_receiver#####1####################WAL#################

#27.16 pg_stat_wal_receiver###

##
##
pid integer
WAL############ID###
status text
WAL###############
receive_start_lsn pg_lsn
WAL##############################
receive_start_tli integer
WAL##########################
written_lsn pg_lsn
########
flushed_lsn pg_lsn
################################## ########WAL################### ######
received_tli integer
########WAL##############
last_msg_send_time timestamp with time zone
####WAL##########################

##	
	##
last_msg_receipt_time	timestamp with time zone
	####WAL##########################
latest_end_lsn	pg_lsn
	####WAL##########################
latest_end_time	timestamp with time zone
	####WAL##############################
slot_name	text
	WAL############################
sender_host	text
	WAL##########PostgreSQL############ ########IP#########UNIX######### ################ #######/##########################
sender_port	integer
	WAL##########PostgreSQL##############
conninfo	text
	########################WAL####################

27.2.7. pg_stat_recovery_prefetch

pg_stat_recovery_prefetch####1########
wal_distance#block_distance#io_depth#############
#pg_stat_reset_shared####################

#27.17 pg_stat_recovery_prefetch###

##	
	##
stats_reset	timestamp with time zone
	##################
prefetch	bigint
	################################
hit	bigint
	###################################
skip_init	bigint
	################################
skip_new	bigint
	################################
skip_fpw	bigint
	#########WAL##############################
skip_rep	bigint
	##
wal_distance	int
	#####################
block_distance	int
	#########################
io_depth	int
	#########################

27.2.8. pg_stat_subscription

#27.18 pg_stat_subscription###

# #	##
subid oid	#########OID###
subname name	##############
worker_type text	######################### #####apply#parallel apply#table synchronization###
pid integer	######################ID###
leader_pid integer	##################################ID# ############################### ###NULL#
relid oid	#################OID##########################NULL###
received_lsn pg_lsn	################################0### ###########NULL###
last_msg_send_time timestamp with time zone	####WAL##NULL###
last_msg_receipt_time timestamp with time zone	####WAL##NULL###
latest_end_lsn pg_lsn	####WAL##NULL###
latest_end_time timestamp with time zone	####WAL##NULL###

27.2.9. pg_stat_subscription_stats

pg_stat_subscription_stats################1########

#27.19 pg_stat_subscription_stats###

# #	##
subid oid	#########OID###
subname name	##############
apply_error_count bigint	###################
sync_error_count bigint	######################
stats_reset timestamp with time zone	##################

27.2.10. pg_stat_ssl

pg_stat_ssl#################WAL########1##########SSL################
pg_stat_activity###pg_stat_replication#pid##############################

#27.20 **pg_stat_ssl###**

##	##
pid integer	############WAL##########ID###
ssl boolean	#####SSL###############
version text	#######SSL#############SSL##########NULL######
cipher text	#######SSL############SSL##########NULL######
bits integer	##########################SSL##########NULL######
client_dn text	###################(DN)##################################SSL##### ######NULL###### #########DN######NAMEDATALEN########64########### #####
client_serial numeric	##SSL##########NULL###### ## ##
issuer_dn text	#############DN#########################SSL##########NULL###### # #######client_dn############

27.2.11. pg_stat_gssapi

pg_stat_gssapi##########1##########GSSAPI############## ################
###pg_stat_activity#pg_stat_replication#pid########

#27.21 **pg_stat_gssapi###**

##	##
pid integer	##########ID###
gss_authenticated boolean	#####GSSAPI##############
principal text	###########################GSSAPI###########NULL### ############## #NAMEDATALEN########64###################
encrypted boolean	#####GSSAPI###############
credentials_delegated boolean	#####GSSAPI##################

27.2.12. pg_stat_archiver

pg_stat_archiver##

#27.22 pg_stat_archiver###

##	##
archived_count bigint	#########WAL########
last_archived_wal text	############WAL#########
last_archived_time timestamp with time zone	###################
failed_count bigint	WAL##################
last_failed_wal text	##############WAL#########
last_failed_time timestamp with time zone	##################
stats_reset timestamp with time zone	##################

###WAL###
############# ######last_archived_wal#################################
##

27.2.13. pg_stat_io

pg_stat_io#####################I/O##########I/O################1#############
I/O######## #################

#########################I/O######## #######################I/O#################
##############################

#27.23 pg_stat_io View

##	##
backend_type text	### backend_type####### #pg_stat_activity######### ###backend_type#I/O################## ########
object text	I/O################ ############ • relation: ########### • temp relation: ##########
context text	I/O########### ############ • normal: I/O####################context### ################################ ############ ##################################context normal####### • ######vacuum: ### ################# ##########################context normal####### • bulkread: #######I/O##

##
• bulkwrite: COPY#######################I/O#####
reads bigint op_bytes####################
read_time double precision #########################track_io_timing###############
writes bigint op_bytes####################
write_time double precision #########################track_io_timing###############
writebacks bigint ############################op_bytes#########
writeback_time double precision ##########################track_io_timing################### ############ ##############################
extends bigint op_bytes#########################
extend_time double precision ######################track_io_timing###############
op_bytes bigint I/O############################ ##############################BLCKSZ#######block_size######### ### ####8192###
hits bigint #########################
evictions bigint ## context#normal## context#bulkwrite#bulkread####vacuum############I/O############### ##
reuses bigint ################################bulkread#bulkwrite####vacuum context##I/O##################
fsyncs bigint fsync######## ####context normal#########
fsync_time double precision fsync####################track_io_timing###############
stats_reset timestamp with time zone ########################

#################I/O#########/######I/O#######I/O######### ################# #
##############################backend_type checkpointer###object temp
relation############

###################I/O##########I/O###########I/O################ #######NULL#
############fsync######fsyncs#object temp relation####NULL######
####################backend_type#temp relation####reads#NULL######

pg_stat_io########################## #######

• evictions####################################

- ## client backend#####fsync################################## ###############28.5## #######
- ## ##### ## ############# ######28.5##########

> ##
>
> ######### I/O####track_io_timing######################## ########## ##track_io_timing#######################I/O########################

27.2.14. pg_stat_bgwriter

#####[82.269504]#pg_stat_archiver## #### ##pg_stat_bgwriter######1##############################

#27.24 pg_stat_bgwriter###

##	##
buffers_clean bigint	##########################
maxwritten_clean bigint	##
buffers_alloc bigint	#############
stats_reset timestamp with time zone	###################

27.2.15. pg_stat_checkpointer

#####[85.135135]#pg_stat_archiver## #### ##pg_stat_checkpointer######1################################

#27.25 pg_stat_checkpointer View

##	##
num_timed bigint	##################################
num_requested bigint	##############################
restartpoints_timed bigint	##
restartpoints_req bigint	####################
restartpoints_done bigint	#####[62.500000]############################## ####################
write_time double precision	

##	##
	#####[78.378378]## ###### ###
sync_time double precision	#####[74.509804]## ###### ###
buffers_written bigint	#####[59.677419]######################### ############################## #######
stats_reset timestamp with time zone	##################

27.2.16. pg_stat_wal

pg_stat_wal###########WAL###########1########

#27.26 pg_stat_wal###

##	##
wal_records bigint	#####WAL#########
wal_fpi bigint	#####WAL##############
wal_bytes numeric	#####WAL###########
wal_buffers_full bigint	WAL#############WAL###################
wal_write bigint	WAL#####XLogWrite################### ##WAL##XLogWrite###### ##28.5#########
wal_sync bigint	WAL#####issue_xlog_fsync#################fsync#on# #wal_sync_method#fdatasync###fsync###fsync_writethrough######## ############## ##WAL##issue_xlog_fsync########28.5#########
wal_write_time double precision	XLogWrite######WAL##############################track_wal_io_timing ##################### ####wal_sync_method#open_datasync## #open_sync##############
wal_sync_time double precision	issue_xlog_fsync#####WAL################### #########track_wal_io_timing#####fsync#on## #wal_sync_method#fdatasync###fsync###fsync_writethrough###### #########0####
stats_reset timestamp with time zone	##################

27.2.17. pg_stat_database

pg_stat_database##################1################1#################### ###

#27.27 pg_stat_database###

##	##
datid oid	#######OID#########################0######
datname name	######################NULL######
numbackends integer	######################################NULL###### ################### ############# ##############################
xact_commit bigint	##########################
xact_rollback bigint	############################
blks_read bigint	#########################
blks_hit bigint	##PostgreSQL##### ###
tup_returned bigint	##
tup_fetched bigint	################################
tup_inserted bigint	#######################
tup_updated bigint	#######################
tup_deleted bigint	#######################
conflicts bigint	################################## (################### ###### #pg_stat_database_conflicts#########)
temp_files bigint	################################## ########################### #log_temp_files##########################
temp_bytes bigint	################################### ###############log_temp_files####### ###################
deadlocks bigint	######################
checksum_failures bigint	##NULL# ##
checksum_last_failure timestamp with time zone	## NULL###
blk_read_time double precision	##(track_io_timing######## #########)#

##	##
blk_write_time double precision	##(track_io_timing######## #########)#
session_time double precision	## ##############################
active_time double precision	########SQL#################### #pg_stat_activity#active#fastpath function call#########
idle_in_transaction_time double precision	##pg_stat_activity#idle in transaction#idle in transaction (aborted)#########
sessions bigint	##########################
sessions_abandoned bigint	##
sessions_fatal bigint	##
sessions_killed bigint	###
stats_reset timestamp with time zone	##################

27.2.18. pg_stat_database_conflicts

pg_stat_database_conflicts#############1################################ ############################## ##

#27.28 pg_stat_database_conflicts###

##	##
datid oid	#######OID###
datname name	###########
confl_tablespace bigint	##
confl_lock bigint	##
confl_snapshot bigint	##
confl_bufferpin bigint	##
confl_deadlock bigint	##
confl_active_logicalslot bigint	#####################wal_level###

27.2.19. pg_stat_all_tables

pg_stat_all_tables##################TOAST#########1#################### ############ pg_stat_user_tables###pg_stat_sys_tables################## ##############################

#27.29 pg_stat_all_tables###

# #	##
relid oid	#####OID###
schemaname name	##################
relname name	#########
seq_scan bigint	###########################
last_seq_scan timestamp with time zone	##
seq_tup_read bigint	###########################
idx_scan bigint	###########################
last_idx_scan timestamp with time zone	##
idx_tup_fetch bigint	##########################
n_tup_ins bigint	##########
n_tup_upd bigint	########## ######n_tup_hot_upd#n_tup_newpage_upd############HOT ###############
n_tup_del bigint	##########
n_tup_hot_upd bigint	HOT######### ##############################
n_tup_newpage_upd bigint	##############################t_ctid############################### # ######HOT#####
n_live_tup bigint	###########
n_dead_tup bigint	###########
n_mod_since_analyze bigint	##############################
n_ins_since_vacuum bigint	##################################
last_vacuum timestamp with time zone	#######################VACUUM FULL########

##
last_autovacuum timestamp with time zone ##################################
last_analyze timestamp with time zone ###################
last_autoanalyze timestamp with time zone ##############################
vacuum_count bigint ######################VACUUM FULL########
autovacuum_count bigint ##############################
analyze_count bigint ##################
autoanalyze_count bigint ############################

27.2.20. pg_stat_all_indexes

pg_stat_all_indexes#######################1############################## ##### pg_stat_user_indexes#pg_stat_sys_indexes########################## ######################################

#27.30 pg_stat_all_indexes###

##
relid oid ################OID###
indexrelid oid ######OID###
schemaname name ###################
relname name ######################
indexrelname name ###########
idx_scan bigint ##############################
last_idx_scan timestamp with time zone ##
idx_tup_read bigint ##################################
idx_tup_fetch bigint ##

############AND#OR################ ## ################ ########################## #pg_stat_all_indexes.idx_tup_read########### #pg_stat_all_tables.idx_tup_fetch#######

##pg_stat_all_indexes.idx_tup_fetch####### ############################## ################################### ###########################

##

idx_tup_read#idx_tup_fetch################################## idx_tup_read#########################idx_tup_fetch################ ####### ## ###############

##

##SQL############9.25####### ####################################1################### ############## #####pg_stat_all_indexes.idx_scan################################ ########################

27.2.21. pg_statio_all_tables

pg_statio_all_tables###################TOAST##########1##############I/O# ############# pg_statio_user_tables#pg_statio_sys_tables############## ##

#27.31 pg_statio_all_tables###

# #	##
relid oid	#####OID###
schemaname name	##################
relname name	#########
heap_blks_read bigint	########################
heap_blks_hit bigint	################
idx_blks_read bigint	###################################
idx_blks_hit bigint	############################
toast_blks_read bigint	#####TOAST##############################
toast_blks_hit bigint	#####TOAST#########################
tidx_blks_read bigint	#####TOAST#####################################
tidx_blks_hit bigint	

##
#####TOAST################################

27.2.22. pg_statio_all_indexes

pg_statio_all_indexes#######################1#################I/O#########
pg_statio_user_indexes#pg_statio_sys_indexes####################
##

#27.32 **pg_statio_all_indexes###**

##
relid oid #################OID###
indexrelid oid #######OID###
schemaname name ####################
relname name #######################
indexrelname name ############
idx_blks_read bigint #########################
idx_blks_hit bigint ####################

27.2.23. pg_statio_all_sequences

pg_statio_all_sequences######################1#################I/O#########
######

#27.33 **pg_statio_all_sequences###**

##
relid oid ######OID###
schemaname name ####################
relname name ###########
blks_read bigint #########################
blks_hit bigint ####################

27.2.24. pg_stat_user_functions

pg_stat_user_functions#############1##########################
track_functions##########################

#27.34 pg_stat_user_functions###

# #	##
funcid oid	###OID###
schemaname name	################
funcname name	########
calls bigint	#############
total_time double precision	######################################
self_time double precision	###

27.2.25. pg_stat_slru

#####[85.064935]#PostgreSQL#*SLRU*(simple least-recently-used)#########################
pg_stat_slru#########SLRU########1#################################
######PostgreSQL#SLRU## ##############
SLRU########1##################################

#################SLRU################################_buffers########

#27.35 pg_stat_slru###

# #	##
name text	SLRU######
blks_zeroed bigint	###################
blks_hit bigint	SLRU##SLRU################ ################################
blks_read bigint	SLRU###################
blks_written bigint	SLRU##################
blks_exists bigint	SLRU################
flushes bigint	SLRU##################
truncates bigint	SLRU###########
stats_reset timestamp with time zone	

##
##
##################

27.2.26.

##
####psql##\d+ pg_stat_activity###
####### ##OID###### ###
################################OID###### ################OID###### #########
##

#################### 27.36######

#27.36

##
##
pg_backend_pid () → integer
############################ID######
pg_stat_get_activity (integer) → setof record
########ID####################NULL################################
######## ########pg_stat_activity########
pg_stat_get_snapshot_timestamp () → timestamp with time zone
######################### ######################NULL######
stats_fetch_consistency#snapshot################################
####################
pg_stat_get_xact_blocks_fetched (oid) → bigint
####
#pg_stat_get_xact_blocks_hit#########read()########### ########
################################
pg_stat_get_xact_blocks_hit (oid) → bigint
##read()##
############
pg_stat_clear_snapshot () → void
################################
pg_stat_reset () → void
##################################
##EXECUTE##########
pg_stat_reset_shared ([*target* text DEFAULT NULL]) → void
##################################### *target*###########
• ######archiver: pg_stat_archiver########################
• ######bgwriter: pg_stat_bgwriter########################
• ######checkpointer: pg_stat_checkpointer########################
• ######io: pg_stat_io########################
• ######recovery_prefetch: pg_stat_recovery_prefetch###########
##############

• ######slru: pg_stat_slru####################### • ######wal: pg_stat_wal####################### • ######NULL########: ####################### ##EXECUTE#########
pg_stat_reset_single_table_counters (oid) → void ### ##### ##EXECUTE#########
pg_stat_reset_single_function_counters (oid) → void ################################## ##EXECUTE#########
pg_stat_reset_slru ([*target* text DEFAULT NULL]) → void #####[69.385027]#####SLRU##################SLRU##### ########### ###NULL#####pg_stat_slru############# SLRU##################### ########################### #CommitTs#MultiXactMember#MultiXactOffset#Notify#Serial#Subtrans#Xact# 1######## ###other(#######################)############## ###########SLRU##################### ########SLRU###### ##########SLRU################ *target*#NULL############### ##pg_stat_slru###########SLRU################### ############### ##commit_timestamp#multixact_member#multixact_offset#notify#serializable#s ###### ###other ############## #######################SLRU######## ########## ##EXECUTE#########
pg_stat_reset_replication_slot (text) → void ################################ ###NULL########################### ##### ##EXECUTE#########
pg_stat_reset_subscription_stats (oid) → void pg_stat_subscription_stats################################## ## #NULL############################ ##EXECUTE#########

##
pg_stat_reset()#####################ANALYZE####################### ########### ## ################ #######################ANALYZE##############

######pg_stat_get_activity##pg_stat_activity########################## ######################## ################################### ################ ################### ##### 27.37######## #######################ID####### ###### ##############ID#########(> = 0)### #########ID################# ######ID##### ## ##pg_stat_get_backend_idset###### ####################ID################## ################PID############## ###########

```
SELECT pg_stat_get_backend_pid(backendid) AS pid,
       pg_stat_get_backend_activity(backendid) AS query
FROM pg_stat_get_backend_idset() AS backendid;
```

#27.37 ###############

##
pg_stat_get_backend_activity(integer) → text ##########################
pg_stat_get_backend_activity_start(integer) → timestamp with time zone ###########################
pg_stat_get_backend_client_addr(integer) → inet ##################IP#########
pg_stat_get_backend_client_port(integer) → integer ###############TCP##########
pg_stat_get_backend_dbid(integer) → oid #################OID######
pg_stat_get_backend_idset() → setof integer #############ID##########
pg_stat_get_backend_pid(integer) → integer ##########ID######
pg_stat_get_backend_start(integer) → timestamp with time zone #################
pg_stat_get_backend_subxact(integer) → record #####ID################################## ########################### ####################*subxact_count*################################ ########*subxact_overflow*###
pg_stat_get_backend_userid(integer) → oid ################OID######
pg_stat_get_backend_wait_event(integer) → text ##############################NULL###### #### 27.5### 27.13###########
pg_stat_get_backend_wait_event_type(integer) → text ##############################NULL###### ####### 27.4##########
pg_stat_get_backend_xact_start(integer) → timestamp with time zone ##############################

27.3.

##########################pg_locks############# ########################### ######################## ########################

- ##PostgreSQL################ #####
- ##

- ##

pg_locks#######52.12###### PostgreSQL##################13##########

27.4.

PostgreSQL################################ ##################### ##ANALYZE#CLUSTER#CREATE INDEX#VACUUM#COPY#####BASE_BACKUP(### ##pg_basebackup##############################)##### ##############################

27.4.1. ANALYZE##########

ANALYZE################pg_stat_progress_analyze######################### ######### ####################################

#27.38 pg_stat_progress_analyze###

##	##
pid integer	##########ID###
datid oid	####################OID###
datname name	#######################
relid oid	###########OID###
phase text	############ # 27.39#########
sample_blks_total bigint	###################
sample_blks_scanned bigint	##################
ext_stats_total bigint	###########
ext_stats_computed bigint	################ ###########computing extended statistics###### ####
child_tables_total bigint	#########
child_tables_done bigint	################ ###########acquiring inherited sample rows#### ######
current_child_table_relid oid	################OID### ############acquiring inherited sample rows########

#27.39 ANALYZE#####

####	##
initializing	######################### ########################
acquiring sample rows	###############relid#####################

####	##
acquiring inherited sample rows	################################ #child_tables_total#child_tables_done#current_child_table ################
computing statistics	#####################################
computing extended statistics	##
finalizing analyze	#####pg_class######## ############ANALYZE#######

##

ANALYZE### ### ##ANALYZE###

27.4.2. CLUSTER###########

CLUSTER#VACUUM FULL################pg_stat_progress_cluster############ ############################ ####################################

#27.40 pg_stat_progress_cluster###

# #	##
pid integer	###########ID###
datid oid	####################OID###
datname name	#########################
relid oid	##############OID###
command text	############ CLUSTER#VACUUM FULL#######
phase text	############## # 27.41#########
cluster_index_relid oid	##OID############
heap_tuples_scanned bigint	################## ############seq scanning heap#index scanning heap#####writing new heap############
heap_tuples_written bigint	############## ############seq scanning heap#index scanning heap#####writing new heap############
heap_blks_total bigint	################## #####seq scanning heap#############
heap_blks_scanned bigint	################## ############seq scanning heap############
index_rebuild_count bigint	############# ###########rebuilding index############

#27.41 CLUSTER#VACUUM FULL#####

####	##
initializing	########################## ######################
seq scanning heap	###################################
index scanning heap	CLUSTER###############################
sorting tuples	CLUSTER################
writing new heap	CLUSTER################
swapping relation files	###############################
rebuilding index	#######################
performing final cleanup	###################### ########## ##CLUSTER#VACUUM FULL######

27.4.3. COPY###########

COPY###############pg_stat_progress_copy#######COPY#################### ####### ####################################

#27.42 pg_stat_progress_copy###

##
pid integer ##########ID###
datid oid ####################OID###
datname name #########################
relid oid COPY##############OID### SELECT##############0########
command text ########### COPY FROM###COPY TO###
type text #####[86.196319]########################## FILE# PROGRAM# PIPE#COPY FROM STDIN###COPY TO STDOUT### ###CALLBACK### ################################ ########################## ### FILE#PROGRAM#PIPE#COPY FROM STDIN#COPY TO STDOUT## ###CALLBACK##################################
bytes_processed bigint COPY##################
bytes_total bigint COPY FROM######################### ########0########
tuples_processed bigint COPY##################
tuples_excluded bigint COPY#####WHERE########################
tuples_skipped bigint ################################ ########ON_ERROR#########stop#### ###############

27.4.4. CREATE INDEX##########

CREATE INDEX#REINDEX###############pg_stat_progress_create_index#####
#######################1######## ####################################

#27.43 pg_stat_progress_create_index###

##
pid integer ####################ID###
datid oid ###################OID###
datname name #########################
relid oid #################OID###
index_relid oid ##################OID### #######CREATE INDEX#######0###
command text ##########CREATE INDEX#CREATE INDEX CONCURRENTLY#REINDEX### #REINDEX CONCURRENTLY###
phase text ##################### # 27.44#########
lockers_total bigint #######################
lockers_done bigint ################
current_locker_pid bigint ##################ID###
blocks_total bigint ##############################
blocks_done bigint ######################
tuples_total bigint ##############################
tuples_done bigint ######################
partitions_total bigint ### 0##REINDEX# ###########################
partitions_done bigint ### 0##REINDEX##############################

#27.44 CREATE INDEX#####

####	##
initializing	CREATE INDEX#REINDEX################# ######### ##############

####	##
waiting for writers before build	CREATE INDEX CONCURRENTLY#REINDEX CONCURRENTLY########################### ######################### ################ ####### #lockers_total#lockers_done### ##current_locker_pid#################
building index	################################ ############ ## ### ######blocks_total#blocks_done######## #tuples_total#tuples_done############
waiting for writers before validation	CREATE INDEX CONCURRENTLY#REINDEX CONCURRENTLY########################### ########################## ################ ######## #lockers_total#lockers_done### ##current_locker_pid#################
index validation: scanning index	CREATE INDEX CONCURRENTLY############# ################## ####################### #blocks_total##################blocks_done## ##############
index validation: sorting tuples	CREATE INDEX CONCURRENTLY################ (scanning index)############
index validation: scanning table	CREATE INDEX CONCURRENTLY####2################ ######################### ######################## #blocks_total################blocks_done###### ##########
waiting for old snapshots	CREATE INDEX CONCURRENTLY#REINDEX CONCURRENTLY########################### ########################## ################ ######## #lockers_total#lockers_done### ##current_locker_pid#################
waiting for readers before marking dead	REINDEX CONCURRENTLY###################### ################################### ########## ############ #lockers_total#lockers_done### ##current_locker_pid#################
waiting for readers before dropping	REINDEX CONCURRENTLY###################### ################################## ########### ########### #lockers_total#lockers_done### ##current_locker_pid#################

27.4.5. VACUUM##########

VACUUM############pg_stat_progress_vacuum############################## ####################1####### ################################### VACUUM FULL# #######pg_stat_progress_cluster################VACUUM################## ##VACUUM FULL#CLUSTER################## 27.4.2#########

#27.45 pg_stat_progress_vacuum###

##
pid integer ##########ID###
datid oid

##	##
	####################OID###
datname name	#########################
relid oid	##################OID###
phase text	#################### # 27.46##########
heap_blks_total bigint	################## ########################### ###############VACUUM# ####################
heap_blks_scanned bigint	################### ## ## ##heap_blks_total## ####### #############scanning heap###########
heap_blks_vacuumed bigint	##################### ##################################vacuuming heap########### ##
index_vacuum_count bigint	#########################
max_dead_tuple_bytes bigint	#####[78.620690]################################maintenance_work_mem### ############## ######maintenance_work_mem################################ ###########
dead_tuple_bytes bigint	#####[75.714286]################################### #################### ##############################
num_dead_item_ids bigint	##
indexes_total bigint	########################## ####vacuuming indexes#######cleaning up indexes################
indexes_processed bigint	#####[58.169935]############### ############rebuilding index########## ### #################### ## ############

#27.46 VACUUM#####

####	##
initializing	VACUUM###################### #################### ####
scanning heap	VACUUM################ ############################ ################# #############heap_blks_scanned# ########
vacuuming indexes	VACUUM##################### ############### ## maintenance_work_mem################################ #############autovacuum_work_mem################### ####################

####	##
vacuuming heap	VACUUM################## ############### ### heap_blks_scanned#heap_blks_total############ ######################### ######################### ########
cleaning up indexes	VACUUM################## ######################### ###########################
truncating heap	VACUUM### ###### ######################
performing final cleanup	VACUUM################## ########VACUUM######## #######pg_class######################### ######## #####VACUUM#######

27.4.6.

pg_basebackup##############################pg_stat_progress_basebackup### ####BASE_BACKUP################################WAL################ ####### ##############################

#27.47 pg_stat_progress_basebackup###

##
pid integer WAL###########ID###
phase text ############ # 27.48##########
backup_total bigint ################ #######streaming database files############ ## ######streaming database files###################WAL###### ##################################### ############################ #backup_streamed####### pg_basebackup##########(#####--no-estimate-size###########)####NULL###
backup_streamed bigint ############### ###########streaming database files## #transferring wal files##########
tablespaces_total bigint ###################
tablespaces_streamed bigint ################## ###########streaming database files###########

#27.48

####	##
initializing	WAL########################## ################### ###
waiting for checkpoint to finish	WAL########################## #pg_backup_start############################### ##
estimating backup size	WAL## ####

####	##
streaming database files	WAL##
waiting for wal archiving to finish	WAL########pg_backup_stop################ ##################WAL###################### ### pg_basebackup#--wal-method=none###--wal-method=stream##################################
transferring wal files	WAL###################WAL################ pg_basebackup#--wal-method=fetch################ #waiting for wal archiving to finish####### ## ########################

27.5.

PostgreSQL################################ ##################################### ##################

################################### ################################### ###### ##########PostgreSQL###################configure################################

##############Solaris#macOS#FreeBSD#NetBSD#Oracle Linux#####DTrace[1]########### ####### SystemTap[2]#######DTrace#####Linux###################### ############## #####src/include/utils/probes.h#######################

27.5.1.

##################configure##############PostgreSQL######################## DTrace##########configure#--enable-dtrace####### ###17.3.3.6#########

27.5.2.

27.49#################################### 27.50#################### # ##PostgreSQL#########################

#27.49 #######DTrace####

##	#####	##
transaction-start	(LocalTransactionId)	######################## ##arg0########ID###
transaction-commit	(LocalTransactionId)	########################arg0 ########ID###
transaction-abort	(LocalTransactionId)	########################arg0 ########ID###
query-start	(const char *)	######################arg0### ########
query-done	(const char *)	########################arg0# ##########
query-parse-start	(const char *)	######################## ##arg0###########
query-parse-done	(const char *)	######################### ##arg0###########

[1] https://en.wikipedia.org/wiki/DTrace
[2] https://sourceware.org/systemtap/

##	#####	##
query-rewrite-start	(const char *)	######################## ##arg0###########
query-rewrite-done	(const char *)	######################### ##arg0###########
query-plan-start	()	##########################
query-plan-done	()	###########################
query-execute-start	()	########(#######)############ ####
query-execute-done	()	########(#######)########### #######
statement-status	(const char *)	######## #pg_stat_activity.status# ################ arg0######## #####
checkpoint-start	(int)	################### ## arg0############## (shutdown#immediate#force)##### ################
checkpoint-done	(int, int, int, int, int)	######################## (## ############################ ##) arg0############arg1##### ####arg2#3#4################ WAL########
clog-checkpoint-start	(bool)	CLOG##################### #### arg0#true############### ##false######################### ###
clog-checkpoint-done	(bool)	CLOG################ ########## arg0#clog-checkpoint-start######## ##
subtrans-checkpoint-start	(bool)	############################ ####### arg0#true########### #####false################### ######
subtrans-checkpoint-done	(bool)	########################## ########## arg0#subtrans-checkpoint-start######## ##
multixact-checkpoint-start	(bool)	############################ ######## arg0#true########### ######false################## #######
multixact-checkpoint-done	(bool)	########################## ########## arg0#multixact-checkpoint-start######## ##
buffer-checkpoint-start	(int)	########################## ####### arg0##############

##	#####	##
		(shutdown#immediate#force)##### ################
`buffer-sync-start`	`(int, int)`	############################ #######(################### #####)# arg0#########arg1##### #####################
`buffer-sync-written`	`(int)`	############################ ######## arg0######ID######
`buffer-sync-done`	`(int, int, int)`	############################ arg0########## arg1######### ################### arg2#### ##################(`buffer-sync-start`#arg1##)### ###### ############################ ######
`buffer-checkpoint-sync-start`	`()`	############################ #########################
`buffer-checkpoint-done`	`()`	############################ ######
`twophase-checkpoint-start`	`()`	########################### ####
`twophase-checkpoint-done`	`()`	########################### ######
`buffer-extend-start`	`(ForkNumber, BlockNumber, Oid, Oid, Oid, int, unsigned int)`	Probe that fires when a relation extension starts. arg0 contains the fork to be extended. arg1, arg2, and arg3 contain the tablespace, database, and relation OIDs identifying the relation. arg4 is the ID of the backend which created the temporary relation for a local buffer, or `INVALID_PROC_NUMBER` (-1) for a shared buffer. arg5 is the number of blocks the caller would like to extend by.
`buffer-extend-done`	`(ForkNumber, BlockNumber, Oid, Oid, Oid, int, unsigned int, BlockNumber)`	Probe that fires when a relation extension is complete. arg0 contains the fork to be extended. arg1, arg2, and arg3 contain the tablespace, database, and relation OIDs identifying the relation. arg4 is the ID of the backend which created the temporary relation for a local buffer, or `INVALID_PROC_NUMBER` (-1) for a shared buffer. arg5 is the number of blocks the relation was extended by, this can be less than the number in the `buffer-extend-start` due to resource constraints. arg6 contains the BlockNumber of the first new block.

##	#####	##
`buffer-read-start`	`(ForkNumber, BlockNumber, Oid, Oid, Oid, int)`	Probe that fires when a buffer read is started. arg0 and arg1 contain the fork and block numbers of the page. arg2, arg3, and arg4 contain the tablespace, database, and relation OIDs identifying the relation. arg5 is the ID of the backend which created the temporary relation for a local buffer, or `INVALID_PROC_NUMBER` (-1) for a shared buffer.
`buffer-read-done`	`(ForkNumber, BlockNumber, Oid, Oid, Oid, int, bool)`	Probe that fires when a buffer read is complete. arg0 and arg1 contain the fork and block numbers of the page. arg2, arg3, and arg4 contain the tablespace, database, and relation OIDs identifying the relation. arg5 is the ID of the backend which created the temporary relation for a local buffer, or `INVALID_PROC_NUMBER` (-1) for a shared buffer. arg6 is true if the buffer was found in the pool, false if not.
`buffer-flush-start`	`(ForkNumber, BlockNumber, Oid, Oid, Oid)`	########################## arg0#arg1################### ### arg2#arg3#arg4########### ########################OID ###
`buffer-flush-done`	`(ForkNumber, BlockNumber, Oid, Oid, Oid)`	################### (###### ########################## ########################## ########) ###`buffer-flush-start`######
`wal-buffer-write-dirty-start`	`()`	WAL######################## #WAL################### (# ##############wal_buffers#### ###########)
`wal-buffer-write-dirty-done`	`()`	#####WAL################### ####
`wal-insert`	`(unsigned char, unsigned char)`	WAL################## arg0# ##############(rmid)### arg1## #######
`wal-switch`	`()`	WAL########################
`smgr-md-read-start`	`(ForkNumber, BlockNumber, Oid, Oid, Oid, int)`	Probe that fires when beginning to read a block from a relation. arg0 and arg1 contain the fork and block numbers of the page. arg2, arg3, and arg4 contain the tablespace, database, and relation OIDs identifying the relation. arg5 is the ID of the backend which created the temporary relation for a local buffer,

##	#####	##
		or INVALID_PROC_NUMBER (-1) for a shared buffer.
smgr-md-read-done	(ForkNumber, BlockNumber, Oid, Oid, Oid, int, int, int)	Probe that fires when a block read is complete. arg0 and arg1 contain the fork and block numbers of the page. arg2, arg3, and arg4 contain the tablespace, database, and relation OIDs identifying the relation. arg5 is the ID of the backend which created the temporary relation for a local buffer, or INVALID_PROC_NUMBER (-1) for a shared buffer. arg6 is the number of bytes actually read, while arg7 is the number requested (if these are different it indicates a short read).
smgr-md-write-start	(ForkNumber, BlockNumber, Oid, Oid, Oid, int)	Probe that fires when beginning to write a block to a relation. arg0 and arg1 contain the fork and block numbers of the page. arg2, arg3, and arg4 contain the tablespace, database, and relation OIDs identifying the relation. arg5 is the ID of the backend which created the temporary relation for a local buffer, or INVALID_PROC_NUMBER (-1) for a shared buffer.
smgr-md-write-done	(ForkNumber, BlockNumber, Oid, Oid, Oid, int, int, int)	Probe that fires when a block write is complete. arg0 and arg1 contain the fork and block numbers of the page. arg2, arg3, and arg4 contain the tablespace, database, and relation OIDs identifying the relation. arg5 is the ID of the backend which created the temporary relation for a local buffer, or INVALID_PROC_NUMBER (-1) for a shared buffer. arg6 is the number of bytes actually written, while arg7 is the number requested (if these are different it indicates a short write).
sort-start	(int, bool, int, int, bool, int)	################### arg0#####################datum######## arg1#true################## arg2########### arg3######(work_mem)######### arg4#true#################### ############### arg5##0###### ##1############2########### ##########
sort-done	(bool, long)	################### arg0#true# #######false########### arg1#

##	#####	##
		########################### ######################
lwlock-acquire	(char *, LWLockMode)	LWLock############## arg0# LWLock########### arg1###### ###################
lwlock-release	(char *)	LWLock################### ########################### ##### arg0#LWLock###########
lwlock-wait-start	(char *, LWLockMode)	LWLock################### ########################### #### arg0#LWLock########### arg1######################## ##
lwlock-wait-done	(char *, LWLockMode)	########LWLock########### ########################### ### arg0#LWLock########### arg1######################## ##
lwlock-condacquire	(char *, LWLockMode)	####################LWLock# ############### arg0#LWLock# ########## arg1############# ############
lwlock-condacquire-fail	(char *, LWLockMode)	####################LWLock# ############### arg0#LWLock# ########## arg1############# ############
lock-wait-start	(unsigned int, unsigned int, unsigned int, unsigned int, unsigned int, LOCKMODE)	#####(lmgr lock)############## ############################ ################ arg0##arg3# ###################### arg4# ###################### arg5### ##############
lock-wait-done	(unsigned int, unsigned int, unsigned int, unsigned int, unsigned int, LOCKMODE)	#####(lmgr lock)########### #####(##########)# ###lock-wait-start#####
deadlock-found	()	########################### ####

#27.50

#	##
LocalTransactionId	unsigned int
LWLockMode	int
LOCKMODE	int
BlockNumber	unsigned int
Oid	unsigned int
ForkNumber	int
bool	unsigned char

27.5.3.

#############pg_stat_database#####################################DTrace#######

```
#!/usr/sbin/dtrace -qs

postgresql$1:::transaction-start
{
      @start["Start"] = count();
      self->ts  = timestamp;
}

postgresql$1:::transaction-abort
{
      @abort["Abort"] = count();
}

postgresql$1:::transaction-commit
/self->ts/
{
      @commit["Commit"] = count();
      @time["Total time (ns)"] = sum(timestamp - self->ts);
      self->ts=0;
}
```

#######D##################

```
# ./txn_count.d `pgrep -n postgres` or ./txn_count.d <PID>
^C

Start                                           71
Commit                                          70
Total time (ns)                        2312105013
```

##

#####################SystemTap#DTrace################### ################ ##SystemTap## #######SystemTap#######################

DTrace## ################################ ## ##################

27.5.4.

###################

1. ################################
2. src/backend/utils/probes.d#############
3. ###################pg_trace.h## #TRACE_POSTGRESQL########
4. ###########################

#: ###########ID################################

1. ######transaction-start###LocalTransactionId###################
2. src/backend/utils/probes.d#############:

   ```
   probe transaction__start(LocalTransactionId);
   ```

   ################################ DTrace#################################### #############transaction-start#####################

3. #######transaction__start#TRACE_POSTGRESQL_TRANSACTION_START##### ##########(###################)########pg_trace.h###################### ########################### ##############

   ```
   TRACE_POSTGRESQL_TRANSACTION_START(vxid.localTransactionId);
   ```

4. ############################DTrace################################## ###### ###########:

   ```
   # dtrace -ln transaction-start
      ID    PROVIDER          MODULE           FUNCTION NAME
   18705 postgresql49878     postgres     StartTransactionCommand
    transaction-start
   18755 postgresql49877     postgres     StartTransactionCommand
    transaction-start
   18805 postgresql49876     postgres     StartTransactionCommand
    transaction-start
   18855 postgresql49875     postgres     StartTransactionCommand
    transaction-start
   18986 postgresql49873     postgres     StartTransactionCommand
    transaction-start
   ```

C##################################:

- ### #######################
- #################PostgreSQL#--enable-dtrace############################## ############################## ################################## ########## ################### ##:

  ```
  if (TRACE_POSTGRESQL_TRANSACTION_START_ENABLED())
      TRACE_POSTGRESQL_TRANSACTION_START(some_function(...));
  ```

  ##########ENABLED##########

27.6. Monitoring Disk Usage

###########PostgreSQL##################################

27.6.1. Determining Disk Usage

######TOAST##65.2########## TOAST ##############################1####### ################################## #### ############################## #####1################################## ######### ########65.1#######

####################3######### # 9.100###SQL######### oid2name################ ################## SQL######################## ############################## #########

##################################psql###

```
SELECT pg_relation_filepath(oid), relpages FROM pg_class WHERE relname =
 'customer';

 pg_relation_filepath | relpages
----------------------+----------
 base/16384/16806     |       60
(1 row)
```

######1######8#######relpages#VACUUM#ANALYZE####CREATE INDEX########DDL
######################### ##

######TOAST#####################################

```
SELECT relname, relpages
FROM pg_class,
     (SELECT reltoastrelid
      FROM pg_class
      WHERE relname = 'customer') AS ss
WHERE oid = ss.reltoastrelid OR
      oid = (SELECT indexrelid
             FROM pg_index
             WHERE indrelid = ss.reltoastrelid)
ORDER BY relname;

       relname        | relpages
----------------------+----------
 pg_toast_16806       |        0
 pg_toast_16806_index |        1
```

##########################

```
SELECT c2.relname, c2.relpages
FROM pg_class c, pg_class c2, pg_index i
WHERE c.relname = 'customer' AND
      c.oid = i.indrelid AND
      c2.oid = i.indexrelid
ORDER BY c2.relname;

      relname      | relpages
-------------------+----------
 customer_id_index |       26
```

##

```
SELECT relname, relpages
FROM pg_class
ORDER BY relpages DESC;

       relname        | relpages
----------------------+----------
 bigtable             |     3290
 customer             |     3144
```

27.6.2. Disk Full Failure

####################
############ ########################## ##WAL################################
#####################

#################################### ################################## ######## ############### ### 22.6##########

###

############

##

#28#

##########################PostgreSQL##################

28.1.

################################ PostgreSQL############################## ######## ## ##################################### ## ######## ## ##########################

############CPU########### ## ############################### ####### ## ################################### ######################### PostgreSQL########### ##############wal_sync_method##

################################### ###RAID#################### ######*write-through*## ###*write-back*############### #################### ## ################## ##############################(BBUs)######################### ######## ##############################

################################write-through#######write-back### ################# ####write-back################################ ########IDE###SATA############### ####write-back######################### ##############(SSD)########write-back###### ######

###

- Linux##`hdparm -I`########IDE###SATA######################### `Write cache#` ## *###################### `hdparm -W 0`#################### SCSI###### ##sdparm[1]############# `sdparm --get=WCE`#####################`sdparm --clear=WCE`###############
- FreeBSD###IDE#######`camcontrol identify`##########################/`boot/loader.conf`#hw.ata.wc=0######SCSI########`camcontrol identify`## ###########`sdparm`####################################
- Solaris#################`format -e`####### (Solaris#ZFS######################## ######################################)
- #####[80.000000]#Windows#####`wal_sync_method #open_datasync`(#####)####My `Computer\Open\`*`disk drive`*`\Properties\Hardware\Properties\Policies\Enable write caching on the disk`############ ################ ###########`wal_sync_method`#fdatasync#NTFS####fsync## #fsync_writethrough######################## ######Windows###`wal_sync_`# ########_datasync##########My######\####\#######\Properties\##### #\Properties\Policies\## ## ####`wal_sync_`#####fdatasync#NTFS#####fsync####################
- macOS###`wal_sync_method`#fsync_writethrough########################## ##

###SATA####(ATAPI-6#######)##################(`FLUSH CACHE EXT`)#########SCSI ############`SYNCHRONIZE CACHE`############### ###########PostgreSQL#######

[1] http://sg.danny.cz/sg/sdparm.html

###############(###ZFS#ext4)#######write-back######################## ############ write##############################(BBU)################################ ##BBU################# pg_test_fsync################################### ######################write#### #######(#########)###################BBU######################## ##write### ### ######################### ##########################

##write-back############### ##SSD## #### `diskchecker.pl`[2]#######I/O##########################

###########512################# ####################################### ######################512############(PostgreSQL#### ###8192#######16##########)##512#### ## ############PostgreSQL######## ##############################WAL########### ######PostgreSQL############### ##########WAL############## ####################################ZFS######## ##full_page_writes##BBU################## ##8kB##BBU#####################################

###PostgreSQL## ########

- WAL##################################CRC-32 (32-bit)################ CRC#### ####WAL###
- ####################################WAL############################## ####### ############initdb#########
- `pg_xact#pg_subtrans#pg_multixact#pg_serial#pg_notify#pg_stat#pg_snapshots#` ## ############################## ######################WAL##############WAL#################
- `pg_twophase`############CRC-32#########
- ##WAL ##############

PostgreSQL####################################Error Correcting Codes -ECC-######## #####RAM############

28.2.

######################

##########initdb################################ ################################## ##

#########################`SHOW data_checksums`###################data_checksums# ################

#ignore_checksum_failure#######

[2] https://brad.livejournal.com/2116715.html

28.2.1.

pg_checksums###

28.3. ########(WAL)

########WAL########################## ##########################(###########)# ############### #######WAL#########(##############)########################### ######WAL## ######################### ### ##############################WAL##################REDO################### ######

> ###
>
> WAL##WAL############## ################## ### ################## ### Linux#ext3###########`data=writeback`###### ###########################

WAL############################## ## ##################WAL############################ WAL######################## ########################WAL################ ################################## ################ ##############################WAL#######`fsync`############ ########################

###WAL####25.3## WAL################ WAL######################## ##############################WAL############## ######### ### ################ ##################WAL######################

28.4.

##############################

################################ ############WAL############################## ## ############## ## ############ #########WAL## ### ################################

######################## ## ################################ ## ######################## ########ATM### # ##

################################WAL### ############# ## ####### ## ############################## ############ ###########B################A################B##############A##

################################## ## ###### ## ########################

#synchronous_commit####### ### ######## ##################################synchronous_commit########

###DROP TABLE##################synchronous_commit#################### ### ## PREPARE TRANSACTION###### #########################

####################WAL### ####### #############WAL###########WAL#####wal_writer_delay################# ################ WAL## #wal_writer_delay#####

##

########fsync = off################# fsync################################## ## ####PostgreSQL## ############## #PostgreSQL## ###### ####################fsync##############################

##commit_delay## ########## #commit_delay######## ########WAL## ##commit_delay######## # ## #################

28.5. WAL###

################WAL################### ################ ################### ###19##########

##WAL########## ## (############WAL################) ## ##WAL########(###redo###########)##REDO################# ################### ################################ ##############redo##################WAL###### ######################## (WAL##############WAL############################### ###)

##I/O######### ################I/O#### ## ################################

#################################### checkpoint_timeout##########max_wal_size## ################################### ##############5##1GB####### ############ ####WAL#####checkpoint_timeout########################## (WAL######### ## ##archive_timeout##############) ###CHECKPOINT SQL############################

checkpoint_timeout###max_wal_size################################### #### ## ## #################### full_page_writes## ##### ## ############ ################WAL############################## ###############I/O#######

################### 1#######################################2################## ##WAL############## ### ### ###########################checkpoint_warning################# ############

#checkpoint_warning#######max_wal_size########################## ######## ### max_wal_size######## #########COPY################################

#############I/O## #### #checkpoint_completion_target########checkpoint_timeout######################## ####### I/O##################checkpoint_timeout###########max_wal_size# ## ######0.9#######PostgreSQL######## ##########################90%################################## ########### ####I/O##########################I/O############ ############################## ######### ###################WAL################### ################## ##checkpoint_timeout###I/O############# ###### ####checkpoint_completion_target##########################I/O###### ####################################I/O######################## checkpoint_completion_target####1.0#################################### ####################0.9################# 1.0#################################### ## ######WAL######################################

#####[92.294221]#Linux###POSIX## ########checkpoint_flush_after####OS############# ################OS############## ##################fsync################# ################################## # #############shared_buffers#########OS################################

pg_wal########WAL#############min_wal_size#max_wal_size################## ###WAL######### ##WAL#########################(###########################)## ### WAL####################max_wal_size################################## ## ##WAL########### ################### #####WAL################## ################################ ################### ######################## min_wal_size##############WAL################ ######## ########WAL###############WAL####################WAL###############

max_wal_size#########wal_keep_size######WAL#############WAL############## # ##WAL## WAL##########WAL##### #############archive_command#archive_library###################WAL#### #pg_wal############ ##(26.2.6### ###)

#####[54.486540]## ##################WAL#####################################pg_control######## ####pg_wal##########WAL################### ############################## ### ############################## #checkpoint_timeout#############max_wal_size################ ########### ##########################1###########WAL#####max_wal_size########### (# ###max_wal_size###) ### ##,####### ######################## ## #####################pg_#######################WAL#################### ##pg_wal#########WAL################ ##################################### ############################ ################################## pg_stat_checkpointer#############_timed#######################_req#2## ############ ###################checkpoint_timeout################################## ## ###################15########## ##### ##WAL####max_wal_size## ############ ##_wal_################ ## #############1###WAL###### ##_wal_## ############## pg_stat_checkpointer#############_done################### ############

###################WAL################INSERT########################## _req##################### ##

##XLOG################################## ################################## ##
#######_done##

######2#####WAL####### XLogInsertRecord#XLogFlush### XLogInsertRecord#
######WAL################## ##################XLogInsertRecord#######
WAL################################ ################ ######################
#######XLogInsertRecord###
#################### #########WAL##############################WAL######
############### ###WAL###########XLogFlush######### ###############
WAL###############
##XLogInsertRecord##################XLogFlush################## #######
###wal_buffers#########WAL############### full_page_writes#####################
##wal_buffers######################################

commit_delay######XLogFlush##
################# ##
#WAL################# fsync##########commit_siblings#########################
################# ## ##############
############10#######10000#######commit_delay##########################
###

commit_delay###
################################
#############commit_delay################# pg_test_fsync#########WAL#####
################################ #############8kB##########################
####commit_delay########### ##
WAL#####################commit_delay##############################
####################RAID## ####
################################ commit_siblings#######################
#commit_siblings################### #######commit_delay################
####################################

commit_delay#(######)########################## ########################
##############gangway
effect##############commit_delay#########################commit_delay####
############ commit_delay###
######################### ##
################################sibling################################

wal_sync_method######PostgreSQL########WAL#######################
fsync_writethrough#####################fsync_writethrough#############
################################ ##################################
pg_test_fsync################################ ############fsync###############
####

wal_debug################XLogInsertRecord#XLogFlush###WAL###############
#################PostgreSQL################
##

WAL###############2########### XLogWrite#issue_xlog_fsync###
track_wal_io_timing######XLogWrite#WAL#############issue_xlog_fsync#WAL
######################pg_stat_wal#wal_write_time###wal_sync_time#########
XLogWrite##WAL##############issue_xlog_fsync##########
#XLogInsertRecord#WAL#####################XLogFlush#WAL##############
issue_xlog_fsync####WAL################XLogWrite###########
wal_sync_method#open_datasync###open_sync####XLogWrite###############
###WAL##########issue_xlog_fsync########
wal_sync_method#fdatasync#fsync####fsync_writethrough###############
WAL##################issue_xlog_fsync###############

track_wal_io_timing########XLogWrite#######issue_xlog_fsync######
WAL############pg_stat_wal#wal_write#wal_sync###########

recovery_prefetch#################PostgreSQL#################################### ################################## maintenance_io_concurrency#wal_decode_buffer_size# ############################## #######try########posix_fadvise############### ########

28.6. WAL###

WAL############ WAL##28.5########### ############

################WAL#####WAL########### ##############(LSN)###########LSN# WAL############################ LSN###pg_lsn############ 2##LSN######### WAL##

WAL#################pg_wal#########16##################################(## #######initdb#--wal-segsize###########)# ########8################(######--with-wal-blocksize###configure###########)# WAL#########access/xlogrecord.h## ############## #000000010000000000000001#################### ########################## ##############

################################WAL########### ###pg_wal#################### ##

WAL## ################################## ############################## ########## #####PostgreSQL#WAL###(28.1# #######)

#############WAL######################pg_control####### ################ ####pg_control####################### ######################WAL############ #REDO####### ################################WAL#########(full_page_writes#### #################)#######################################

pg_control############WAL## ################ ############ pg_control#########1########################## ############ ##############pg_control#################################### ## ###pg_control####################

#29#

##############################(replication identity)################################## ######### ## PostgreSQL# ################## #26######### ##

#######################(publisher)##################(publications)################## (subscribers)##########(publish)########(subscribe)########## ######################## ##

#######UPDATE####DELETE########################## INSERT#################
###########

##################################

##########CREATE PUBLICATION############################

#######ALTER PUBLICATION############ ADD TABLE###DROP TABLE############
##

29.2.

########################## ################################## ###############
##

##################PostgreSQL##
###############

####################################
##

##############################26.2.6################## ##########################
##

################################26.2.8#################### ####################
########## ####################application_name##################

###########################pg_dump######## ################################
##########################pg_subscription################

##########CREATE SUBSCRIPTION#####ALTER SUBSCRIPTION#################
#DROP SUBSCRIPTION########

################################### ##############################

####################
#####################

##########################
####################

#################### ################################### #################
############################### #####integer#####bigint############## #######
################################### ################################### ######
######################### ###CREATE SUBSCRIPTION#binary##############

29.2.1.

###

###########
######pg_%u_sync_%u_%llu#################*oid*#####*relid*#######*sysid*######
########

#####[77.534791]#####################CREATE SUBSCRIPTION#################
##################DROP SUBSCRIPTION################## ################
######
####################CREATE SUBSCRIPTION############################DROP

SUBSCRIPTION################################ ################################ ### ################################ ##################

- ##################################### #####create_slot = false############ ################################
- ## #######connect = false#### ##################### ################### ####pg_dump########### ############ ##
- #####[78.457447]## ################### ################################## ######################ALTER SUBSCRIPTION# ############# ## ############# ################################### #####################ALTER SUBSCRIPTION################################
- ################################ ######################ALTER SUBSCRIPTION# ################ ## ################## ## ########WAL ############################ #######################

29.2.2. #:

##########################

```
test_pub=# CREATE TABLE t1(a int, b text, PRIMARY KEY(a));
CREATE TABLE
test_pub=# CREATE TABLE t2(c int, d text, PRIMARY KEY(c));
CREATE TABLE
test_pub=# CREATE TABLE t3(e int, f text, PRIMARY KEY(e));
CREATE TABLE
```

######################

```
test_sub=# CREATE TABLE t1(a int, b text, PRIMARY KEY(a));
CREATE TABLE
test_sub=# CREATE TABLE t2(c int, d text, PRIMARY KEY(c));
CREATE TABLE
test_sub=# CREATE TABLE t3(e int, f text, PRIMARY KEY(e));
CREATE TABLE
```

#######################

```
test_pub=# INSERT INTO t1 VALUES (1, 'one'), (2, 'two'), (3, 'three');
INSERT 0 3
test_pub=# INSERT INTO t2 VALUES (1, 'A'), (2, 'B'), (3, 'C');
INSERT 0 3
test_pub=# INSERT INTO t3 VALUES (1, 'i'), (2, 'ii'), (3, 'iii');
INSERT 0 3
```

######################### ########pub2#pub3a#######publish########### ####### #pub3b############29.4#####

```
test_pub=# CREATE PUBLICATION pub1 FOR TABLE t1;
CREATE PUBLICATION
test_pub=# CREATE PUBLICATION pub2 FOR TABLE t2 WITH (publish =
 'truncate');
CREATE PUBLICATION
test_pub=# CREATE PUBLICATION pub3a FOR TABLE t3 WITH (publish =
 'truncate');
```

```
CREATE PUBLICATION
test_pub=# CREATE PUBLICATION pub3b FOR TABLE t3 WHERE (e > 5);
CREATE PUBLICATION

#######################################sub3##pub3a#pub3b##############################
###########################


test_sub=# CREATE SUBSCRIPTION sub1
test_sub-# CONNECTION 'host=localhost dbname=test_pub
 application_name=sub1'
test_sub-# PUBLICATION pub1;
CREATE SUBSCRIPTION
test_sub=# CREATE SUBSCRIPTION sub2
test_sub-# CONNECTION 'host=localhost dbname=test_pub
 application_name=sub2'
test_sub-# PUBLICATION pub2;
CREATE SUBSCRIPTION
test_sub=# CREATE SUBSCRIPTION sub3
test_sub-# CONNECTION 'host=localhost dbname=test_pub
 application_name=sub3'
test_sub-# PUBLICATION pub3a, pub3b;
CREATE SUBSCRIPTION

##################################################


test_sub=# SELECT * FROM t1;
 a |   b
---+-------
 1 | one
 2 | two
 3 | three
(3 rows)

test_sub=# SELECT * FROM t2;
 c | d
---+---
 1 | A
 2 | B
 3 | C
(3 rows)

#####################################pub3a#######################t3##########
#pub3b##################################


test_sub=# SELECT * FROM t3;
 e |  f
---+-----
 1 | i
 2 | ii
 3 | iii
(3 rows)

###########################


test_pub=# INSERT INTO t1 VALUES (4, 'four'), (5, 'five'), (6, 'six');
INSERT 0 3
test_pub=# INSERT INTO t2 VALUES (4, 'D'), (5, 'E'), (6, 'F');
INSERT 0 3
test_pub=# INSERT INTO t3 VALUES (4, 'iv'), (5, 'v'), (6, 'vi');
INSERT 0 3
```

######################

```
test_pub=# SELECT * FROM t1;
 a |   b
---+-------
 1 | one
 2 | two
 3 | three
 4 | four
 5 | five
 6 | six
(6 rows)

test_pub=# SELECT * FROM t2;
 c | d
---+---
 1 | A
 2 | B
 3 | C
 4 | D
 5 | E
 6 | F
(6 rows)

test_pub=# SELECT * FROM t3;
 e |  f
---+-----
 1 | i
 2 | ii
 3 | iii
 4 | iv
 5 | v
 6 | vi
(6 rows)
```

###########pub2## #pub3a#INSERT################### ##########pub3b#pub3b################### ######### ######################:

```
test_sub=# SELECT * FROM t1;
 a |   b
---+-------
 1 | one
 2 | two
 3 | three
 4 | four
 5 | five
 6 | six
(6 rows)

test_sub=# SELECT * FROM t2;
 c | d
---+---
 1 | A
 2 | B
 3 | C
(3 rows)

test_sub=# SELECT * FROM t3;
 e |  f
---+-----
 1 | i
 2 | ii
```

```
 3 | iii
 6 | vi
(4 rows)
```

29.2.3. #:

################################29.2.1###
######## ##################################### #######################
#pgoutput####################################

##########################

```
test_pub=# CREATE PUBLICATION pub1 FOR ALL TABLES;
CREATE PUBLICATION
```

#1: ##########connect = false##########

- ################

  ```
  test_sub=# CREATE SUBSCRIPTION sub1
  test_sub-# CONNECTION 'host=localhost dbname=test_pub'
  test_sub-# PUBLICATION pub1
  test_sub-# WITH (connect=false);
  WARNING:  subscription was created, but is not connected
  HINT:  To initiate replication, you must manually create the replication
   slot, enable the subscription, and refresh the subscription.
  CREATE SUBSCRIPTION
  ```

- ######################## CREATE SUBSCRIPTION################################
  ####################"sub1"###

  ```
  test_pub=# SELECT * FROM pg_create_logical_replication_slot('sub1',
   'pgoutput');
   slot_name |    lsn
  -----------+-----------
   sub1      | 0/19404D0
  (1 row)
  ```

- ################################ ######pub1####################

  ```
  test_sub=# ALTER SUBSCRIPTION sub1 ENABLE;
  ALTER SUBSCRIPTION
  test_sub=# ALTER SUBSCRIPTION sub1 REFRESH PUBLICATION;
  ALTER SUBSCRIPTION
  ```

#######2: ##########connect = false##########slot_name################

- ################

  ```
  test_sub=# CREATE SUBSCRIPTION sub1
  test_sub-# CONNECTION 'host=localhost dbname=test_pub'
  test_sub-# PUBLICATION pub1
  test_sub-# WITH (connect=false, slot_name='myslot');
  WARNING:  subscription was created, but is not connected
  HINT:  To initiate replication, you must manually create the replication
   slot, enable the subscription, and refresh the subscription.
  CREATE SUBSCRIPTION
  ```

- #########CREATE SUBSCRIPTION#####################################"myslot"###

```
test_pub=# SELECT * FROM pg_create_logical_replication_slot('myslot',
 'pgoutput');
 slot_name |    lsn
-----------+-----------
 myslot    | 0/19059A0
(1 row)
```

- ##

```
test_sub=# ALTER SUBSCRIPTION sub1 ENABLE;
ALTER SUBSCRIPTION
test_sub=# ALTER SUBSCRIPTION sub1 REFRESH PUBLICATION;
ALTER SUBSCRIPTION
```

#3: ##########slot_name = NONE#########

- ################ slot_name = NONE####enabled = false#create_slot = false######

```
test_sub=# CREATE SUBSCRIPTION sub1
test_sub-# CONNECTION 'host=localhost dbname=test_pub'
test_sub-# PUBLICATION pub1
test_sub-# WITH (slot_name=NONE, enabled=false, create_slot=false);
CREATE SUBSCRIPTION
```

- #########"myslot"########################

```
test_pub=# SELECT * FROM pg_create_logical_replication_slot('myslot',
 'pgoutput');
 slot_name |    lsn
-----------+-----------
 myslot    | 0/1905930
(1 row)
```

- ##

```
test_sub=# ALTER SUBSCRIPTION sub1 SET (slot_name='myslot');
ALTER SUBSCRIPTION
```

- ##############################

```
test_sub=# ALTER SUBSCRIPTION sub1 ENABLE;
ALTER SUBSCRIPTION
test_sub=# ALTER SUBSCRIPTION sub1 REFRESH PUBLICATION;
ALTER SUBSCRIPTION
```

29.3. Logical Replication Failover

##
######################### ####/###########=####################/################
###################### ########47.2.3########## ##############################
############################ ## ######
##

####
########### ## ##
##synchronized_#####_slots##############

##
######################

1. ######################SQL## ####
   ##

```
test_sub=# SELECT
               array_agg(quote_literal(s.subslotname)) AS slots
           FROM  pg_subscription s
           WHERE s.subfailover AND
                 s.subslotname IS NOT NULL;
 slots
-------
 {'sub1','sub2','sub3'}
(1 row)
```

2. ######################SQL## ####
   ## ##############################
   ##################51.55##### ##
   ##################################

```
test_sub=# SELECT
               array_agg(quote_literal(slot_name)) AS slots
           FROM
           (
               SELECT CONCAT('pg_', srsubid, '_sync_', srrelid, '_',
 ctl.system_identifier) AS slot_name
               FROM pg_control_system() ctl, pg_subscription_rel r,
 pg_subscription s
               WHERE r.srsubstate = 'f' AND s.oid = r.srsubid AND
 s.subfailover
           );
 slots
-------
 {'pg_16394_sync_16385_7394666715149055164'}
(1 row)
```

3. ##

```
test_standby=# SELECT slot_name, (synced AND NOT temporary AND NOT
 conflicting) AS failover_ready
               FROM pg_replication_slots
               WHERE slot_name IN
                   ('sub1','sub2','sub3',
 'pg_16394_sync_16385_7394666715149055164');
  slot_name                                 | failover_ready
---------------------------------------------+----------------
  sub1                                      | t
  sub2                                      | t
  sub3                                      | t
  pg_16394_sync_16385_7394666715149055164   | t
(4 rows)
```

##_ready)########################
SQL

29.4.

###################
############ ## ###################

######################################WHERE######## WHERE# ############## ###CREATE PUBLICATION#########

29.4.1.

############################ ######false###NULL######################### WHERE################################CREATE SUBSCRIPTION##CONNECTION##### ########### ######TRUNCATE##############

29.4.2.

WHERE################ ##

#########UPDATE###DELETE##################WHERE############################ #######(REPLICA IDENTITY###)# ########INSERT####################WHERE##### #########

29.4.3. UPDATE##

UPDATE#####################################(##################)# ######true## ##UPDATE############### ######false#################### #############1####### #########################UPDATE#INSERT###DELETE######## ################### ############################

#################(##############)## ###### ######UPDATE#DELETE########

###############(#################)## ######## ######UPDATE#INSERT########

29.1###############

#29.1 UPDATE####

###	####	##
####	####	###########
####	##	INSERT
##	####	DELETE
##	##	UPDATE

29.4.4.

##publish_via_partition_root## ################## publish_via_partition_root#true###################### ######## ########publish_via_partition_root#false######################### #######

29.4.5.

##WHERE############################### ############

################################WHERE##################*any*############## ## #29.4.6#########

> **##**
>
> ################################publish#############DML############ ################## ######29.8.1########29.2.2##########

> **##**
>
> #########15## ##############################

29.4.6.

###publish##########OR# #################### ##

- ####################
- ###############1##FOR ALL TABLES############ ################
- ###############1##FOR TABLES IN SCHEMA############################ ############

29.4.7.

#######################

```
test_pub=# CREATE TABLE t1(a int, b int, c text, PRIMARY KEY(a,c));
CREATE TABLE
test_pub=# CREATE TABLE t2(d int, e int, f int, PRIMARY KEY(d));
CREATE TABLE
test_pub=# CREATE TABLE t3(g int, h int, i int, PRIMARY KEY(g));
CREATE TABLE
```

#################### ########p1##1######t1###################### ########p2##2# ########### ####t1###############t2############ ########p3##2################# ######

```
test_pub=# CREATE PUBLICATION p1 FOR TABLE t1 WHERE (a > 5 AND c = 'NSW');
CREATE PUBLICATION
test_pub=# CREATE PUBLICATION p2 FOR TABLE t1, t2 WHERE (e = 99);
CREATE PUBLICATION
test_pub=# CREATE PUBLICATION p3 FOR TABLE t2 WHERE (d = 10), t3 WHERE (g
 = 10);
CREATE PUBLICATION
```

psql##

```
test_pub=# \dRp+
                                Publication p1
  Owner   | All tables | Inserts | Updates | Deletes | Truncates | Via
 root
----------+------------+---------+---------+---------+-----------
+----------
 postgres | f          | t       | t       | t       | t         | f
Tables:
```

```
    "public.t1" WHERE ((a > 5) AND (c = 'NSW'::text))

                                Publication p2
  Owner   | All tables | Inserts | Updates | Deletes | Truncates | Via
 root
----------+------------+---------+---------+---------+-----------
+----------
 postgres | f          | t       | t       | t       | t         | f
Tables:
    "public.t1"
    "public.t2" WHERE (e = 99)

                                Publication p3
  Owner   | All tables | Inserts | Updates | Deletes | Truncates | Via
 root
----------+------------+---------+---------+---------+-----------
+----------
 postgres | f          | t       | t       | t       | t         | f
Tables:
    "public.t2" WHERE (d = 10)
    "public.t3" WHERE (g = 10)
```

psql#################################### ####t1#2#################p1########
####t2#2####################################

```
test_pub=# \d t1
                 Table "public.t1"
 Column |  Type   | Collation | Nullable | Default
--------+---------+-----------+----------+---------
 a      | integer |           | not null |
 b      | integer |           |          |
 c      | text    |           | not null |
Indexes:
    "t1_pkey" PRIMARY KEY, btree (a, c)
Publications:
    "p1" WHERE ((a > 5) AND (c = 'NSW'::text))
    "p2"

test_pub=# \d t2
                 Table "public.t2"
 Column |  Type   | Collation | Nullable | Default
--------+---------+-----------+----------+---------
 d      | integer |           | not null |
 e      | integer |           |          |
 f      | integer |           |          |
Indexes:
    "t2_pkey" PRIMARY KEY, btree (d)
Publications:
    "p2" WHERE (e = 99)
    "p3" WHERE (d = 10)

test_pub=# \d t3
                 Table "public.t3"
 Column |  Type   | Collation | Nullable | Default
--------+---------+-----------+----------+---------
 g      | integer |           | not null |
 h      | integer |           |          |
 i      | integer |           |          |
Indexes:
    "t3_pkey" PRIMARY KEY, btree (g)
Publications:
    "p3" WHERE (g = 10)
```

##############################t1############p1##################s1#######

```
test_sub=# CREATE TABLE t1(a int, b int, c text, PRIMARY KEY(a,c));
CREATE TABLE
test_sub=# CREATE SUBSCRIPTION s1
test_sub-# CONNECTION 'host=localhost dbname=test_pub application_name=s1'
test_sub-# PUBLICATION p1;
CREATE SUBSCRIPTION

############ ########p1#t1 WHERE####################


test_pub=# INSERT INTO t1 VALUES (2, 102, 'NSW');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (3, 103, 'QLD');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (4, 104, 'VIC');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (5, 105, 'ACT');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (6, 106, 'NSW');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (7, 107, 'NT');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (8, 108, 'QLD');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES (9, 109, 'NSW');
INSERT 0 1

test_pub=# SELECT * FROM t1;
 a |  b  |  c
---+-----+-----
 2 | 102 | NSW
 3 | 103 | QLD
 4 | 104 | VIC
 5 | 105 | ACT
 6 | 106 | NSW
 7 | 107 | NT
 8 | 108 | QLD
 9 | 109 | NSW
(8 rows)


test_sub=# SELECT * FROM t1;
 a |  b  |  c
---+-----+-----
 6 | 106 | NSW
 9 | 109 | NSW
(2 rows)

#######################p1#t1 WHERE############### UPDATE####################
##


test_pub=# UPDATE t1 SET b = 999 WHERE a = 6;
UPDATE 1

test_pub=# SELECT * FROM t1;
 a |  b  |  c
---+-----+-----
 2 | 102 | NSW
 3 | 103 | QLD
 4 | 104 | VIC
 5 | 105 | ACT
 7 | 107 | NT
 8 | 108 | QLD
```

```
 9 | 109 | NSW
 6 | 999 | NSW
(8 rows)


test_sub=# SELECT * FROM t1;
 a |  b  |  c
---+-----+-----
 9 | 109 | NSW
 6 | 999 | NSW
(2 rows)

#############p1#t1              WHERE#####################################
UPDATE#INSERT#################### ######################


test_pub=# UPDATE t1 SET a = 555 WHERE a = 2;
UPDATE 1

test_pub=# SELECT * FROM t1;
  a  |  b  |  c
-----+-----+-----
   3 | 103 | QLD
   4 | 104 | VIC
   5 | 105 | ACT
   7 | 107 | NT
   8 | 108 | QLD
   9 | 109 | NSW
   6 | 999 | NSW
 555 | 102 | NSW
(8 rows)


test_sub=# SELECT * FROM t1;
  a  |  b  |  c
-----+-----+-----
   9 | 109 | NSW
   6 | 999 | NSW
 555 | 102 | NSW
(3 rows)

#############p1#t1              WHERE#####################################
UPDATE#DELETE#################### ##########################


test_pub=# UPDATE t1 SET c = 'VIC' WHERE a = 9;
UPDATE 1

test_pub=# SELECT * FROM t1;
  a  |  b  |  c
-----+-----+-----
   3 | 103 | QLD
   4 | 104 | VIC
   5 | 105 | ACT
   7 | 107 | NT
   8 | 108 | QLD
   6 | 999 | NSW
 555 | 102 | NSW
   9 | 109 | VIC
(8 rows)


test_sub=# SELECT * FROM t1;
  a  |  b  |  c
-----+-----+-----
```

```
   6 | 999 | NSW
 555 | 102 | NSW
(2 rows)
```

##########################publish_via_partition_root##########################
##################################

##########################

```
test_pub=# CREATE TABLE parent(a int PRIMARY KEY) PARTITION BY RANGE(a);
CREATE TABLE
test_pub=# CREATE TABLE child PARTITION OF parent DEFAULT;
CREATE TABLE
```

#######################

```
test_sub=# CREATE TABLE parent(a int PRIMARY KEY) PARTITION BY RANGE(a);
CREATE TABLE
test_sub=# CREATE TABLE child PARTITION OF parent DEFAULT;
CREATE TABLE
```

########p4################ #############publish_via_partition_root#true#####
##################(parent)########(child)#############

```
test_pub=# CREATE PUBLICATION p4 FOR TABLE parent WHERE (a < 5), child
 WHERE (a >= 5)
test_pub-# WITH (publish_via_partition_root=true);
CREATE PUBLICATION
```

```
test_sub=# CREATE SUBSCRIPTION s4
test_sub-# CONNECTION 'host=localhost dbname=test_pub application_name=s4'
test_sub-# PUBLICATION p4;
CREATE SUBSCRIPTION
```

####################### ##############################
#publish_via_partition_root#true#######

```
test_pub=# INSERT INTO parent VALUES (2), (4), (6);
INSERT 0 3
test_pub=# INSERT INTO child VALUES (3), (5), (7);
INSERT 0 3

test_pub=# SELECT * FROM parent ORDER BY a;
 a
---
 2
 3
 4
 5
 6
 7
(6 rows)
```

```
test_sub=# SELECT * FROM parent ORDER BY a;
 a
---
 2
 3
 4
```

```
(3 rows)
```

##########publish_via_partition_root####### ###########
#publish_via_partition_root########## ############(child)#########

```
test_pub=# DROP PUBLICATION p4;
DROP PUBLICATION
test_pub=# CREATE PUBLICATION p4 FOR TABLE parent, child WHERE (a >= 5)
test_pub-# WITH (publish_via_partition_root=false);
CREATE PUBLICATION
```

```
test_sub=# ALTER SUBSCRIPTION s4 REFRESH PUBLICATION;
ALTER SUBSCRIPTION
```

####################### ####child#####################
(publish_via_partition_root#false#####)#

```
test_pub=# TRUNCATE parent;
TRUNCATE TABLE
test_pub=# INSERT INTO parent VALUES (2), (4), (6);
INSERT 0 3
test_pub=# INSERT INTO child VALUES (3), (5), (7);
INSERT 0 3

test_pub=# SELECT * FROM parent ORDER BY a;
 a
---
 2
 3
 4
 5
 6
 7
(6 rows)
```

```
test_sub=# SELECT * FROM child ORDER BY a;
 a
---
 5
 6
 7
(3 rows)
```

29.5.

####################
############### ## #####CREATE
PUBLICATION#########

################################## ############################ #############
################################ ############################

################################
##########

########################## #################

##############FOR TABLES IN SCHEMA############################

####################publish_via_partition_root#######################
publish_via_partition_root#true############################ ######
##publish_via_partition_root#false##############################

#########UPDATE###DELETE###
(REPLICA IDENTITY###)# ########INSERT#####################################
##

######TRUNCATE############

############################ #############15#################################
################

> **##: ####################**
>
> ## CREATE SUBSCRIPTION###
> ##########################
>
> ##
>
> #####[77.481840]###
> #1################################ALTER SUBSCRIPTION ... DROP PUBLICATION################1################ ########################
> ##
> #######ALTER SUBSCRIPTION ... DROP PUBLICATION###################
> 1##################

29.5.1.

###########t1######

```
test_pub=# CREATE TABLE t1(id int, a text, b text, c text, d text, e text,
 PRIMARY KEY(id));
CREATE TABLE
```

########p1####### ########################t1############### ##################
###########

```
test_pub=# CREATE PUBLICATION p1 FOR TABLE t1 (id, b, a, d);
CREATE PUBLICATION
```

psql##

```
test_pub=# \dRp+
                                Publication p1
  Owner   | All tables | Inserts | Updates | Deletes | Truncates | Via
 root
----------+------------+---------+---------+---------+-----------
+----------
 postgres | f          | t       | t       | t       | t         | f
Tables:
    "public.t1" (id, a, b, d)
```

psql#####################################

```
test_pub=# \d t1
                  Table "public.t1"
 Column |  Type   | Collation | Nullable | Default
--------+---------+-----------+----------+---------
 id     | integer |           | not null |
 a      | text    |           |          |
 b      | text    |           |          |
 c      | text    |           |          |
 d      | text    |           |          |
 e      | text    |           |          |
Indexes:
    "t1_pkey" PRIMARY KEY, btree (id)
Publications:
    "p1" (id, a, b, d)
```

```
#######################t1####################t1############p1##############
####s1######
```

```
test_sub=# CREATE TABLE t1(id int, b text, a text, d text, PRIMARY
 KEY(id));
CREATE TABLE
test_sub=# CREATE SUBSCRIPTION s1
test_sub-# CONNECTION 'host=localhost dbname=test_pub application_name=s1'
test_sub-# PUBLICATION p1;
CREATE SUBSCRIPTION
```

```
###############t1########
```

```
test_pub=# INSERT INTO t1 VALUES(1, 'a-1', 'b-1', 'c-1', 'd-1', 'e-1');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES(2, 'a-2', 'b-2', 'c-2', 'd-2', 'e-2');
INSERT 0 1
test_pub=# INSERT INTO t1 VALUES(3, 'a-3', 'b-3', 'c-3', 'd-3', 'e-3');
INSERT 0 1
test_pub=# SELECT * FROM t1 ORDER BY id;
 id |  a  |  b  |  c  |  d  |  e
----+-----+-----+-----+-----+-----
  1 | a-1 | b-1 | c-1 | d-1 | e-1
  2 | a-2 | b-2 | c-2 | d-2 | e-2
  3 | a-3 | b-3 | c-3 | d-3 | e-3
(3 rows)
```

```
#######p1##########################
```

```
test_sub=# SELECT * FROM t1 ORDER BY id;
 id |  b  |  a  |  d
----+-----+-----+-----
  1 | b-1 | a-1 | d-1
  2 | b-2 | a-2 | d-2
  3 | b-3 | a-3 | d-3
(3 rows)
```

29.6.

```
##############################################################DML###############
############################## ################ UPDATE####DELETE###########
##############################################
```

```
############################################ ################################
##                  ##########################################################
```

#INSERT#UPDATE#DELETE###TRUNCATE########################## ################
###PostgreSQL#######################

############################## ########################## ##################
#############

###
########### ##
################


```
ERROR:  duplicate key value violates unique constraint "test_pkey"
DETAIL:  Key (c)=(1) already exists.
CONTEXT:  processing remote data for replication origin "pg_16395" during
 "INSERT" for replication target relation "public.test" in transaction 725
 finished at 0/14C0378
```

#########################LSN#######################################LSN
0/14C0378###########pg_16395## ##################ALTER SUBSCRIPTION .
##LSN#####LSN 0/14C0378####SKIP# ##LSN###################
###########LSN############ #####pg_replication_origin_advance()#######
######################### ##################################### ALTER
SUBSCRIPTION ... DISABLE#####disable_on_error#######################
####### ###pg_replication_origin_advance()###*node_name*#####pg_16395####
LSN###LSN#####0/14C0379############# ########
##pg_replication_origin_status############## ############################
######################### #######################

######streaming####parallel##################LSN################## #######
#########on###off##############################LSN######################## #
#LSN########ALTER SUBSCRIPTION ... SKIP#########

29.7.

#################################### ##########################

- ############DDL################## ########pg_dump --schema-only########
  ######## ##################### ################################## ########
  ###################### ##
  ## ###########################
  ########################
- ###################### ##############SERIAL##############################
  ##### ################################## ################################
  ########## ##
  #####pg_dump##
- TRUNCATE## ############
  ######################################CASCADE###########################
  ########################## ##
  #### ##
  ########
- ##########33################## ############################
- ## ################################
  ############################
- #####[91.699605]###
  ### (##
  ##) ##

################################ID##############################ID#########
#########(CREATE PUBLICATION########)#

- ############REPLICA IDENTITY FULL############################B########
  ##############################UPDATE#DELETE########################### #
  ######################## ID ########################

29.8.

######################
############################## ################################### #########
##

##########################(26.2.5##)################### walsender#WAL######
#apply################# WAL######WAL#############47####################
#########pgoutput######## ########WAL#########################53.5#######
##
##

#########################session_replication_role#replica########### ####
ALTER TABLE###
##ENABLE TRIGGER###ENABLE RULE####################################

#############COPY#############
####INSERT#################

29.8.1.

#############
########################### ##################################### ###########
################### ##
########################### ## #####
###

##
##############publish########DML#################### ################ ###########################

29.9.

##
##26.2.5.2############

#################pg_stat_subscription########## ########################1#
########## ################0################################

#########################1################ ##################################
#####0#### ## ####streaming########
##################################

29.10.

#######################REPLICATION################################ ####
SUPERUSER#BYPASSRLS################################ #######################

########options=-crow_security=off######## ############################ ########################### #########pg_hba.conf###### LOGIN############

###SELECT############ #### #######################

##############################CREATE################

###

#################### ##################################### ################### ### ######################### ################################PostgreSQL########### ###

##########################pg_create_subscription################CREATE### ############

###########SET ROLE#############

######### run_as_owner = true########################## ############# ######################### ########################### #DELETE#####################SET ROLE####### ############################## ### ################################## ## ### ########

###

##

29.11.

#################################### ################################ ## ##max_replication_slots##

29.11.1.

wal_level#logical############

max_replication_slots###
##

max_wal_senders#######max_replication_slots############################ ###########

###########walsender#wal_sender_timeout########

29.11.2.

max_replication_slots###
########

max_logical_replication_workers###
#######################################

max_worker_processes##
(max_logical_replication_workers + 1)# #################
#max_worker_processes#########################

max_sync_workers_per_subscription#######################################
#########

max_parallel_apply_workers_per_subscription###############streaming =
parallel#################################

###############wal_receiver_timeout#wal_receiver_status_interval###
#wal_retrieve_retry_interval########

29.12.

##postgresql.conf#################

```
wal_level = logical
```

##################################

pg_hba.conf######################## ######################################
###

```
host     all     repuser     0.0.0.0/0     md5
```

#########################

```
CREATE PUBLICATION mypub FOR TABLE users, departments;
```

#######################

```
CREATE SUBSCRIPTION mysub CONNECTION 'dbname=foo host=bar user=repuser'
 PUBLICATION mypub;
```

##########users#departments################### ############################
####

#30# ########(JIT)

############(just-in-time compilation)########PostgreSQL######################

30.1. JIT###########

###(JIT)## ### ##`WHERE a.col = 3`#######SQL##############SQL################################ ####CPU#############################

PostgreSQL#`--with-llvm`###########PostgreSQL##LLVM[1]####JIT################### #######

#######`src/backend/jit/README`########

30.1.1. JIT############

####PostgreSQL#JIT##################(tuple deforming)############### ############### #######

#####`WHERE`############################## ##################################### ##

####################(65.6.1##)################## ################################ #############

30.1.2. ########(Inlining)

PostgreSQL## #36########### #### ########################## ################(36.3##)###################### ###### ###########JIT## ###################### #################

30.1.3.

LLVM######################## ######JIT### ############### ################https://llvm.org/docs/Passes.html#transform-passes########

30.2. ######JIT########

JIT################CPU################ ##########JIT############################ ########################

JIT##################################(#68##19.7.2###)######## ########### #jit_above_cost########### ###############JIT############# ################## # ########jit_inline_above_cost## ######## #jit_optimize_above_cost#################################### ##########JIT###### ################################

################################ ######################(PREPARE##)############# ##

[1] https://llvm.org/

> ##
>
> jit#off##JIT##############--with-llvm################################
> #####JIT######## jit#off#######################

EXPLAIN####JIT################## JIT#############

```
=# EXPLAIN ANALYZE SELECT SUM(relpages) FROM pg_class;
                                                    QUERY PLAN
-----------------------------------------------------------------
-----------------------------------------
 Aggregate  (cost=16.27..16.29 rows=1 width=8) (actual time=0.303..0.303
 rows=1 loops=1)
   ->  Seq Scan on pg_class  (cost=0.00..15.42 rows=342 width=4) (actual
 time=0.017..0.111 rows=356 loops=1)
 Planning Time: 0.116 ms
 Execution Time: 0.365 ms
(4 rows)
```

################JIT################ JIT################### ###########JIT#### #########

```
=# SET jit_above_cost = 10;
SET
=# EXPLAIN ANALYZE SELECT SUM(relpages) FROM pg_class;
                                                    QUERY PLAN
-----------------------------------------------------------------
-----------------------------------------
 Aggregate  (cost=16.27..16.29 rows=1 width=8) (actual time=6.049..6.049
 rows=1 loops=1)
   ->  Seq Scan on pg_class  (cost=0.00..15.42 rows=342 width=4) (actual
 time=0.019..0.052 rows=356 loops=1)
 Planning Time: 0.133 ms
 JIT:
   Functions: 3
   Options: Inlining false, Optimization false, Expressions true,
 Deforming true
   Timing: Generation 1.259 ms (Deform 0.000 ms), Inlining 0.000 ms,
 Optimization 0.797 ms, Emission 5.048 ms, Total 7.104 ms
 Execution Time: 7.416 ms
```

#############JIT################################## ###jit_inline_above_cost### #jit_optimize_above_cost####################

30.3.

########jit##JIT############## ######## #jit_above_cost#jit_inline_above_cost#jit_optimize_above_cost#######JIT################## #####JIT##################

jit_provider###JIT############### ################ 30.4.2##########

19.17#####################################

30.4.

30.4.1.

PostgreSQL#JIT####C#internal#################################### ########### ######################### LLVM JIT#####################PGXS################## #########################

```
########$pkglibdir/bitcode/$extension/#######$pkglibdir/bitcode/
$extension.index.bc#################  ####$pkglibdir##pg_config    --
pkglibdir##########$extension#################
```

> ##
>
> PostgreSQL#######################$pkglibdir/bitcode/postgres######## ####

30.4.2. #####JIT#####

PostgreSQL#LLVM#####JIT######## JIT## ###################LLVM########################## #########jit_provider######## ###

30.4.2.1. JIT############

#######################JIT########### ################################ ###JIT# ############################JIT################_PG_jit_provider_init######C ############## ##

```
struct JitProviderCallbacks
{
    JitProviderResetAfterErrorCB reset_after_error;
    JitProviderReleaseContextCB release_context;
    JitProviderCompileExprCB compile_expr;
};

extern void _PG_jit_provider_init(JitProviderCallbacks *cb);
```

#31#

############PostgreSQL#SQL#################### ###############SQL####### PostgreSQL###SQL#########

31.1.

##########################

31.1.1.

###

```
make check
```

```
#####src/test/regress######################## ################+############ ###########-######### ######################
```

```
# All 213 tests passed.
```

########################### ######################## 31.2 #########

#######################root####################### ####root########### root###### ##################################

##PostgreSQL####################PostgreSQL#################################### #make check## ######## ###################### ##############################configure --disable-rpath############ ######################################

#########################ID################### ######20######################### ####40################### ##############1#########1##psql############## ##50################# ################### ############################ ###############MAX_CONNECTIONS################## ######## ############10#########

```
make MAX_CONNECTIONS=10 check
```

31.1.2.

########17##################18###

```
make installcheck
```

```
##########################
```

```
make installcheck-parallel
```

```
######PGHOST#####PGPORT############################################ ### #regression################### ##########################
```

###########regress_## ##### ##################################installcheck#################

31.1.3.

make check#make installcheck########################## ######PostgreSQL### ################### ## #####

###

```
make check-world
make installcheck-world
```

make check#make installcheck## ################ ################################ make check-world###### #############(############)########make installcheck-world############### #########

###CPU## #####PostgreSQL###### ##############################

```
make check-world -j8 >/dev/null
```

###-j############################## stdout##############################(##### ####################stderr##############)

#######################make check###make installcheck###################### ####### make installcheck###

#############################

- ########################### ###src/pl######
- contrib#####contrib################ ####contrib#####################
- src/interfaces/libpq/test#src/interfaces/ecpg/test################ ########
- ################ src/test/authentication###### (######################## #####)
- ##################### src/test/isolation######
- ######################## src/test/recovery######
- ############## src/test/subscription######
- src/bin##################

installcheck###################regression################### ##pl_regression#contrib_regression### ############################## #installcheck##################

################31.4#####TAP############ #####--enable-tap-tests##### PostgreSQL##########TAP############## ################Perl################### ####

###

##########

```
make check-world PG_TEST_EXTRA='kerberos ldap ssl load_balance
 libpq_encryption'
```

################

kerberos

src/test/kerberos################ ###MIT Kerberos############TCP/IP#####
########

ldap

src/test/ldap################ ###OpenLDAP###########TCP/IP###########
##

ssl

src/test/ssl################ ###TCP/IP##############

load_balance

src/interfaces/libpq/t/004_load_balance_dns.pl######### ########
#hosts###############TCP/IP##############

libpq_encryption

########src/interfaces/libpq/t/005_negotiate_###.pl###### ####
##TCP/IP############## PG_###_EXTRA#Kerberos##########MIT Kerberos#######
##################

wal_consistency_checking

src/test/recovery#############wal_consistency_checking=all###### #
##############################

xid_wraparound

######src/###/#####/xid_################## ###########################
##

#########################PG_TEST_EXTRA################

####make check-world#########make installcheck-world###########src/
test/modules###### ##
##################1##make install#make installcheck##################
##################

31.1.4.

##initdb##########################
##

```
make check LANG=C
make check LC_COLLATE=en_US.utf8 LC_CTYPE=fr_CA.utf8
```

##########LC_ALL############## #####################

###

###########ENCODING####################################

```
make check LANG=C ENCODING=EUC_JP
```

##############################C############### ############################## ##############################

########################

31.1.5.

##debug_discard_caches############################## ##########################

#########PG_#####_INITDB_EXTRA_OPTS############################ #initdb#################### ####################WAL#######work_mem###### #####################

```
make check PG_TEST_INITDB_EXTRA_OPTS='-k --wal-segsize=4 -c work_mem=50MB'
```

#####[53.526971]##PGOPTIONS#### ####### ######pg_regress## #PGOPTIONS############################## ####:

```
make check PGOPTIONS="-c debug_parallel_query=regress -c work_mem=50MB"
```

#########libpq############### #######options##########

################################postgresql.conf#######################

```
echo 'log_checkpoints = on' > test_postgresql.conf
echo 'work_mem = 50MB' >> test_postgresql.conf
make check EXTRA_REGRESS_OPTS="--temp-config=test_postgresql.conf"
```

31.1.6.

###
EXTRA_TESTS##################################### ####numeric_big######

```
make check EXTRA_TESTS=numeric_big
```

31.2.

############################PostgreSQL#################################### ########################## ################################diff########### ######################### ### ######### ################## ##

################src/test/regress/results#################### #######
#diff#############src/test/regress/expected#################### #######
###src/test/regress/regression.diffs####### ########################### ########src/test/regress####################

############diff################PG_REGRESS_DIFF_OPTS='-c'#######
#PG_REGRESS_DIFF_OPTS######## ########diff#############

###31.3#########

31.2.1.

################################## #########PostgreSQL########################## ############# ## ################ ####################################

31.2.2.

C### #### ##

###make#########################
#####

```
make check LANG=de_DE.utf8
```

#################LC_ALL################################## ################### ########################C#######################

```
make check NO_LOCALE=1
```

#########initdb######

##############

31.2.3.

############################# ############ PST8PDT (Berkeley, California) # ## ################################## #PGTZ#PST8PDT#######

31.2.4.

###########64####double precision############################## double precision#################################### float8#geometry######### ## ###############10#################### ######################

###############0#-0######################0#######

###############PostgreSQL#######################pow()#exp()################

31.2.5.

####################################### ######################### ####### ###########SELECT#####ORDER BY###########################SQL######## ############ ## ##ORDER BY################# ##################################### ## #######################C###########work_mem############################## ######################

##########################ORDER BY################################## ##########
##ORDER BY#######

##############
###

31.2.6.

errors####select infinite_recurse()#####################################
##max_stack_depth####################### ########################
#max_stack_depth######4########################### #############
##max_stack_depth#################

getrlimit()#########################max_stack_depth############# ##########
############################

31.2.7. ####

random################################ ######################################
######

diff results/random.out expected/random.out

######################### #########################

31.2.8.

##
##enable_seqscan#enable_indexscan#############EXPLAIN##################
#########

31.3.

################################expected######################### ##############
##################################
###########

1## ######src/test/regress/
resultmap### ######################
#######################################resultmap#####1#########

#######################

testname:output:platformpattern=comparisonfilename

testname####################### output######################### ##################
###out### #################### platformpattern###expr Unix##############^########
######## ###config.guess######################### comparisonfilename############
#####################

########## ############strtof######################float4##################
#####float4-misrounded-input.out################################
######## Cygwin#######################################resultmap#########

float4:out:.*-.*-cygwin.*=float4-misrounded-input.out

####config.guess####.*-.*-cygwin.*#################### resultmap#####
##############################

2############################## ##
####################################*testname*.out#*testname_digit*.out##
##*digit*#0-9########1########################### ###########################
############ ##########diff################################### #resultmap#####
#############resultmap########*testname*##########

####char############char.out##C#####POSIX################### #
##char_1.out##################################

####################################
######################### ##
##
###############

31.4. TAP###

########src/bin#################Perl TAP######Perl#######prove##########
make##PROVE_FLAGS#######prove#################### ####

```
make -C src/bin check PROVE_FLAGS='--timer'
```

######prove##################

######t/*.pl#######################make##PROVE_TESTS#######prove####
#Makefile###################### #########

```
make check PROVE_TESTS='t/001_test1.pl t/003_test3.pl'
```

TAP####Perl#####IPC::Run###### #######CPAN[1]############################## #
##PostgreSQL#--enable-tap-tests##################

#######TAP#####make installcheck#####################################
##make check################################## #################(########)#
###################### ######2################### ############################
####

make installcheck#######TAP############################## #######TAP####
#########TAP#################################### PostgreSQL################
#####TAP####################make installcheck###########################
############

31.4.1.

####PG_TEST_NOCLEAN###########
############################## #####pg_dump#################################
##########

```
PG_TEST_NOCLEAN=1 make -C src/bin/pg_dump check
```

##################################

################180## ###
#PG_TEST_TIMEOUT_DEFAULT#####################################

[1] https://metacpan.org/dist/IPC-Run

31.5.

PostgreSQL## ############################ ######GCC####################gcov###lcov###### #######

31.5.1. Autoconf#Make#######

####################

```
./configure --enable-coverage ... OTHER OPTIONS ...
make
make check # or other test suite
make coverage-html
```

####HTML#####coverage/index.html######

lcov########HTML##############################

```
make coverage
```

make coverage-html##########################.gcov############ #make coverage#make coverage-html##################################

##

```
make coverage-clean
```

##########################make coverage-html####make coverage########## ################

############make distclean#######

31.5.2. Meson#######

####################

```
meson setup -Db_coverage=true ... OTHER OPTIONS ... builddir/
meson compile -C builddir/
meson test -C builddir/
cd builddir/
ninja coverage-html
```

###HTML#####./meson-logs/coveragereport/index.html######

###

IV.

#####[85.288967]#####PostgreSQL################################## ################ ###### ### ###H####### ################ ########################SQL####### II################################ ################### ###########PostgreSQL################################## ########### ######### ## #######################H############### ########## ############################SQL####### II################################### #####

##

32. libpq — C ##### 796
32.1. ########### 796
32.1.1. ##### 803
32.1.2. ######### 805
32.2. ###### 813
32.3. ######## 819
32.3.1. ##### 819
32.3.2. ############### 827
32.3.3. ########### 830
32.3.4. SQL###################### 831
32.4. ######### 833
32.5. ######## 838
32.5.1. ########### 838
32.5.2. ############# 840
32.5.3. ################# 841
32.6. Retrieving Query Results in Chunks 842
32.7. ############### 843
32.7.1. Functions for Sending Cancel Requests 843
32.7.2. Obsolete Functions for Sending Cancel Requests 847
32.8. ######### 848
32.9. ##### 849
32.10. COPY######## 849
32.10.1. COPY######## 850
32.10.2. COPY######## 851
32.10.3. ###COPY### 851
32.11. #### 853
32.12. ##### 855
32.13. #### 858
32.14. ######## 859
32.14.1. ####### 859
32.14.2. ################ 861
32.14.3. ######### 862
32.14.4. ###### 863
32.15. #### 865
32.16. ######### 867
32.17. ######### 867
32.18. #######LDAP## 868
32.19. SSL#### 868
32.19.1. ############### 869
32.19.2. ######## 869
32.19.3. ############## 870
32.19.4. SSL############## 871
32.19.5. SSL######## 871
32.20. ############## 872
32.21. libpq######## 872
32.22. ######## 874
33. ######### 884
33.1. #### 884
33.2. #### 884
33.3. ############ 884
33.3.1. ########### 884
33.3.2. ############### 885
33.3.3. ################ 885
33.3.4. ################# 885
33.3.5. ################### 886

33.3.6. ################### 886
33.3.7. ############ 886
33.3.8. ################## 887
33.3.9. ############### 887
33.3.10. ################ 887
33.3.11. ############ 888
33.4. ####### 888
33.5. ######## 889
34. ECPG — C########SQL 894
34.1. ## 894
34.2. ########## 894
34.2.1. ############ 894
34.2.2. ##### 895
34.2.3. ###### 897
34.3. SQL####### 897
34.3.1. SQL#### 897
34.3.2. ####### 898
34.3.3. ########## 899
34.3.4. ###### 899
34.4. ######## 900
34.4.1. ## 900
34.4.2. ####### 900
34.4.3. ########### 901
34.4.4. ####### 902
34.4.5. #######SQL######## 908
34.4.6. ### 912
34.5. ##SQL 913
34.5.1. #########SQL#### 913
34.5.2. #########SQL#### 913
34.5.3. #########SQL#### 913
34.6. pgtypes ##### 914
34.6.1. ### 915
34.6.2. numeric # 915
34.6.3. ### 918
34.6.4. timestamp# 921
34.6.5. interval# 925
34.6.6. decimal# 925
34.6.7. pgtypeslib#errno# 926
34.6.8. pgtypeslib###### 927
34.7. ######## 927
34.7.1. ####SQL##### 927
34.7.2. SQLDA##### 929
34.8. ##### 938
34.8.1. ######## 939
34.8.2. sqlca 940
34.8.3. `SQLSTATE#SQLCODE` 942
34.9. ######### 945
34.9.1. ########## 945
34.9.2. define###undef### 945
34.9.3. ifdef#ifndef#elif#else#endif### 946
34.10. ####SQL######## 947
34.11. ####### 947
34.12. ######## 948
34.13. C++######## 949
34.13.1. ######### 950
34.13.2. ###C########C++########### 951
34.14. ####SQL#### 952
34.15. Informix##### 974

34.15.1. #### 974
34.15.2. #############SQL# 974
34.15.3. Informix##SQLDA##### 975
34.15.4. #### 978
34.15.5. ##### 986
34.16. Oracle##### 987
34.17. ## 987
35. ###### 990
35.1. #### 990
35.2. #### 990
35.3. `information_schema_catalog_name` 991
35.4. `administrable_role_authorizations` 991
35.5. `applicable_roles` 991
35.6. `attributes` 991
35.7. `character_sets` 993
35.8. `check_constraint_routine_usage` 994
35.9. `check_constraints` 994
35.10. `collations` 995
35.11. `collation_character_set_applicability` 995
35.12. `column_column_usage` 996
35.13. `column_domain_usage` 996
35.14. `column_options` 997
35.15. `column_privileges` 997
35.16. `column_udt_usage` 998
35.17. `columns` 998
35.18. `constraint_column_usage` 1001
35.19. `constraint_table_usage` 1001
35.20. `data_type_privileges` 1002
35.21. `domain_constraints` 1002
35.22. `domain_udt_usage` 1003
35.23. `domains` 1003
35.24. `element_types` 1005
35.25. `enabled_roles` 1006
35.26. `foreign_data_wrapper_options` 1006
35.27. `foreign_data_wrappers` 1007
35.28. `foreign_server_options` 1007
35.29. `foreign_servers` 1008
35.30. `foreign_table_options` 1008
35.31. `foreign_tables` 1008
35.32. `key_column_usage` 1009
35.33. `parameters` 1009
35.34. `referential_constraints` 1011
35.35. `role_column_grants` 1012
35.36. `role_routine_grants` 1012
35.37. `role_table_grants` 1013
35.38. `role_udt_grants` 1013
35.39. `role_usage_grants` 1014
35.40. `routine_column_usage` 1015
35.41. `routine_privileges` 1015
35.42. `routine_routine_usage` 1016
35.43. `routine_sequence_usage` 1016
35.44. `routine_table_usage` 1017
35.45. `routines` 1017
35.46. `schemata` 1021
35.47. `sequences` 1022
35.48. `sql_features` 1023
35.49. `sql_implementation_info` 1023
35.50. `sql_parts` 1023

35.51. sql_sizing 1024
35.52. table_constraints 1024
35.53. table_privileges 1025
35.54. tables 1025
35.55. transforms 1026
35.56. triggered_update_columns 1027
35.57. triggers 1027
35.58. udt_privileges 1029
35.59. usage_privileges 1029
35.60. user_defined_types 1030
35.61. user_mapping_options 1031
35.62. user_mappings 1032
35.63. view_column_usage 1032
35.64. view_routine_usage 1033
35.65. view_table_usage 1033
35.66. views 1034

#32# libpq — C

libpq##C##################PostgreSQL######### libpq##############PostgreSQL#### ######################################

libpq##C++#Perl#Python#Tcl#ECPG######PostgreSQL################################ ## ####libpq################################## ###32.15#32.16###32.19###libpq####### ###########################

#######libpq###################(32.22)###### ##############`src/test/examples`## ######libpq##########################

libpq#######################`libpq-fe.h`###############libpq ###############

32.1.

PostgreSQL################################ ################################## ##### ########1############################## ##### ##PQconnectdb#PQconnectdbParams###PQsetdbLogin#############PGconn##### ########## ##########PGconn##################################NULL####### ############## ##########################PQstatus########################### ##############

> ##
>
> ##search_path### ######################### ###options###########-csearch_path=####### ######## ############PQexec(*conn*, "SELECT pg_catalog.set_config('search_path', '', false)")########### ### ######libpq############## ###SQL###########################

> ##
>
> Unix###libpq## ######### ##########################exec##############################

`PQconnectdbParams`

```
######################


PGconn *PQconnectdbParams(const char * const *keywords,
                          const char * const *values,
                          int expand_dbname);
```

```
######2##NULL##################################1#######   1###############
#keywords################## 2###values################ ###PQsetdbLogin##
################################################################(############
##########PQconnectStartParams#PQconnectPoll)##############
```

```
################32.1.2######
```

```
#####################################1#####################  #############
######  ###keywords#######NULL##########  ####NULL  keywords##########
#values#####NULL##############################################

expand_dbname#0########dbname######################### ###############
#################### ##(=)##########URI##################################
#########  (###############32.1.1#########)  dbname#################### ##
#dbname##########################

########################### ##################(NULL#######)######## ######
####################keywords########################### ##################
######################################### #####dbname####################
####################dbname######################(#########################
####)#

#######################################################   ################
(32.15##)###################### ##########################################
```

PQconnectdb

```
######################


PGconn *PQconnectdb(const char *conninfo);

#####conninfo#####################################

################################## #############1######################### ###
URI############ #######32.1.1#########
```

PQsetdbLogin

```
######################


PGconn *PQsetdbLogin(const char *pghost,
                     const char *pgport,
                     const char *pgoptions,
                     const char *pgtty,
                     const char *dbName,
                     const char *login,
                     const char *pwd);

##############PQconnectdb###### ##################################### ####
########NULL################################

dbName##=##################URI#########PQconnectdb###############
#conninfo########### ###########PQconnectdbParams#############

pgtty######################
```

PQsetdb

```
######################


PGconn *PQsetdb(char *pghost,
                char *pgport,
                char *pgoptions,
                char *pgtty,
```

```
                    char *dbName);
```

####*login*#*pwd*#NULL########PQsetdbLogin########## ################### ########

PQconnectStartParams
PQconnectStart
PQconnectPoll

##############################

```
PGconn *PQconnectStartParams(const char * const *keywords,
                             const char * const *values,
                             int expand_dbname);

PGconn *PQconnectStart(const char *conninfo);

PostgresPollingStatusType PQconnectPoll(PGconn *conn);
```

###3##########I/O###
########I/O######PQconnectdbParams###PQconnectdb#################################
############### ##

PQconnectStartParams#####PQconnectdbParams################
#keywords###values########expand_dbname#######################

PQconnectStart#####PQconnectdb########conninfo#################################
############

PQconnectStartParams#PQconnectStart#PQconnectPoll################
########

- hostaddr#######DNS################################ #######32.1.2########
  ##########

- PQtrace###

- ################PQconnectPoll##################################

#################PQconnectStart#PQconnectStartParams####### #####
NULL####libpq####PGconn################ ############PGconn############
#################################### ##PQstatus(conn)####### #####
#CONNECTION_BAD##

#####[78.169014]#PQconnectStart####PQconnectStartParams###############
########libpq######### ############################PQsocket(conn)#####
######PQconnectPoll########################### #########
#PQconnectPoll(conn)#PGRES_POLLING_READING#####select()#poll()####
################################ ######PQconnectPoll(conn)####### #####
#PQconnectPoll(conn)#PGRES_POLLING_WRITING#######################
##PQconnectPoll(conn)######### ##############PQconnectPoll#########
#####PGRES_POLLING_WRITING##############   ######
#PQconnectPoll(conn)############PGRES_POLLING_FAILED###############
####PGRES_POLLING_OK##########   ######PQconnectStart##
#PQconnectStartParams############libpq####################### ########
####################PQsocket(conn)####### ###: #####PQconnectPoll######
#################   ##############   PQconnectPoll(conn)###
#PGRES_POLLING_READING###########################(select()#poll()###
###############)# #############PQsocketPoll#select(2)#poll(2)########

```
####################        ###PQconnectPoll(conn)#########        #
##PQconnectPoll(conn)####PGRES_POLLING_WRITING#####################
#######PQconnectPoll(conn)####### ############PQconnectPoll########
#######PGRES_POLLING_WRITING###############             ############
#PGRES_POLLING_FAILED################PGRES_POLLING_OK##############
##

############PQstatus######################### ########CONNECTION_BAD#
#################CONNECTION_OK################          ##############
##PQconnectPoll############### ################################## ######
############################################## ###########

CONNECTION_STARTED

   ############

CONNECTION_MADE

   ###OK############

CONNECTION_AWAITING_RESPONSE

   ##############

CONNECTION_AUTH_OK

   #######################

CONNECTION_SSL_STARTUP

   SSL###########

CONNECTION_GSS_STARTUP

   #####[51.851852]#SSL########### ######GSS #############

CONNECTION_CHECK_WRITABLE

   ##############################

CONNECTION_CHECK_STANDBY

   ############# #########################

CONNECTION_CONSUME

   #####################

#########################################################################
######################### ########################


switch(PQstatus(conn))
{
        case CONNECTION_STARTED:
            feedback = "Connecting...";
            break;

        case CONNECTION_MADE:
            feedback = "Connected to server...";
```

```
                break;
    .
    .
    .
            default:
                feedback = "Connecting...";
    }
```

PQconnectPoll#######connect_timeout############## ################## ############## ######PQconnectStart###PQconnectPoll######PQconnectdb# ########

PQconnectStart#PQconnectStartParams##NULL######################## ################PQfinish####################### ##################### ######################

PQsocketPoll

######PQsocket#################################### ############## ##PQconnectStartParams################################

```
typedef pg_int64 pg_usec_time_t;

int PQsocketPoll(int sock, int forRead, int forWrite,
                 pg_usec_time_t end_time);
```

################################## ################## *forRead*########## ###################### *forWrite*################################

############*end_time*####### ####Unix############################## ##time_t×100### *end_time*#-1################# end_time#0################ ######################### ########PQgetCurrentTimeUSec################ ################## ## #####

##############################0################0##############-1###### #####errno(3)################## ######*forRead*#*forWrite*############## ##############

######PQsocketPoll#########poll(2)###########(2)########### ###### ##poll(2)#POLLIN#POLLOUT#####(2)#*readfds*#*writefds*#########

PQconndefaults

##################

```
PQconninfoOption *PQconndefaults(void);

typedef struct
{

    char   *keyword;   /* ############# */
    char   *envvar;    /* ############ */
    char   *compiled;  /* ######################## */
    char   *val;       /* ###############NULL */
    char   *label;     /* #################### */
    char   *dispchar;  /* ##############################
                          #:
                          ""        #############
```

```
                                "*"       ###############
                                "D"       ######################### */
        int     dispsize;  /* ################(#####) */
    } PQconninfoOption;

    ############### ########PQconnectdb#################################
    ###### #####PQconninfoOption#############keyword#####NULL############
    ### ###############NULL######### ########(val #####)###################
    ########################################

    ##################PQconninfoFree######### #########PQconndefaults####
    #######################

PQconninfo

    ########################


    PQconninfoOption *PQconninfo(PGconn *conn);

    ##########################PQconnectdb################################
    ###### ####PQconninfoOption################null keyword #############
    ##PQconndefaults##############PQconninfo##########

PQconninfoParse

    ###############################


    PQconninfoOption *PQconninfoParse(const char *conninfo, char **errmsg);

    ################################################NULL##### ###############
    ##PQconnectdb######################### ####PQconninfoOption############
    ##NULL#keyword###############

    ######################## ##########################PQconninfoOption#NULL#
    #####val###### #############

    errmsg##NULL##########*errmsg#NULL####################malloc########
    ###### #*errmsg#NULL#############NULL########### #################

    ################PQconninfoFree############                  ########
    ##PQconninfoParse##########################     ##############errmsg#
    #NULL#####PQfreemem######################

PQfinish

    ############# ###PGconn###################


    void PQfinish(PGconn *conn);

    ##################PQstatus###############PQfinish#####PGconn#########
    ############ ###PQfinish#########PGconn################

PQreset

    ####################
```

```
void PQreset(PGconn *conn);

##################################################### ####################
############
```

PQresetStart
PQresetPoll

```
##############################


int PQresetStart(PGconn *conn);

PostgresPollingStatusType PQresetPoll(PGconn *conn);

################################################################ #########
###################### PQreset###########2###################### #########
#PQconnectStartParams#PQconnectStart###PQconnectPoll##########

##############PQresetStart#######   ########################   ####1##
##PQconnectPoll####################PQresetPoll####################
```

PQpingParams

```
PQpingParams#############  #######PQconnectdbParams#################
############################################ ############################
#############


PGPing PQpingParams(const char * const *keywords,
                    const char * const *values,
                    int expand_dbname);

###################

PQPING_OK

   #######################

PQPING_REJECT

   ##############################################

PQPING_NO_RESPONSE

   ############ ################################################################
   ##########################################

PQPING_NO_ATTEMPT

   ######################################################################
```

PQping

```
PQping############# #######PQconnectdb################# ##############
############################## ########################################


PGPing PQping(const char *conninfo);

####PQpingParams######
```

PQsetSSLKeyPassHook_OpenSSL

PQsetSSLKeyPassHook_OpenSSL#########sslpassword##############libpq###### ################################

```
void PQsetSSLKeyPassHook_OpenSSL(PQsslKeyPassHook_OpenSSL_type hook);
```

#####################################

```
int callback_fn(char *buf, int size, PGconn *conn);
```

libpq######PQdefaultSSLKeyPassHook_OpenSSL################## ####################*size*###########*buf*######### *buf*######NULL######### ### #######*buf*###########NULL#################### ##############buf[0] = '\0'######0######## #####libpq#######PQdefaultSSLKeyPassHook_OpenSSL## ########

#######################################conn->sslkey####### ############## ############## ################OpenSSL################################## ####

############################PQdefaultSSLKeyPassHook_OpenSSL######### #####0##

#########longjmp(...)######################## #################

PQgetSSLKeyPassHook_OpenSSL

PQgetSSLKeyPassHook_OpenSSL## #NULL######

```
PQsslKeyPassHook_OpenSSL_type PQgetSSLKeyPassHook_OpenSSL(void);
```

32.1.1.

###libpq################################## #################/#####URI######### ######### URI###RFC 3986[1]##

32.1.1.1.

##################################*keyword*=*value*######## ################### ########################keyword = 'a value'################ ############## ############################## ###\'#\\###

#########

```
host=localhost port=5432 dbname=mydb connect_timeout=10
```

##############32.1.2######

32.1.1.2. ##URI

##URI################

[1] https://datatracker.ietf.org/doc/html/rfc3986

```
postgresql://[userspec@][hostspec][/dbname][?paramspec]

where userspec is:

user[:password]

and hostspec is:

[host][:port][,...]

and paramspec is:

name=value[&...]
```

URI#######postgresql://###postgres://############### ###URI############## #######URI###########

```
postgresql://
postgresql://localhost
postgresql://localhost:5433
postgresql://localhost/mydb
postgresql://user@localhost
postgresql://user:secret@localhost
postgresql://other@localhost/otherdb?
connect_timeout=10&application_name=myapp
postgresql://host1:123,host2:456/somedb?
target_session_attrs=any&application_name=myapp
```

URI#####################################:

```
postgresql:///mydb?host=localhost&port=5433
```

JDBC##URI#########ssl=true#######sslmode=require#######################32.1.2#######################

##URI##[2]################## #####(=) #%3D####%20###########

```
postgresql://user@localhost:5433/mydb?options=-c%20synchronous_commit
%3Doff
```

##############IP########### IPv6###################:

```
postgresql://[2001:db8::1234]/database
```

###########host############## ##############################Unix################ ######TCP/IP########### ###URI################################ #########Unix## ####################URI################################URI################## ##################

```
postgresql:///dbname?host=/var/lib/postgresql
postgresql://%2Fvar%2Flib%2Fpostgresql/dbname
```

###URI##################################### postgresql:// host1:port1,host2:port2,host3:port3/######URI ##host=host1,host2,host3 port=port1,port2,port3################ ####### ################################

[2] https://datatracker.ietf.org/doc/html/rfc3986#section-2.1

32.1.1.3.

################################ #####/######host#hostaddr#port########### ############### ################################ #######hostaddr########### ########hostaddr######################## ########port################### #########

##URI#####host##############host:port#########

#########IPv4#IPv6###############

######################################

###########################

32.1.2.

############################

host

############### ######################TCP/IP#########Unix############# ### ################################ #Unix####################Windows######### ############### #####@#####################Linux#Windows#############Unix## ############### host###########################/tmp#####PostgreSQL######### #############Unix################### Windows########localhost#########

################### ###################### ############################### ## ###32.1.1.3########

hostaddr

########IP########### ####172.28.40.9#######IPv4################ ###### ##IPv6############################ ######################TCP/IP######### ## ##################IP#########host##############host#IP############### #########

hostaddr################################## ############################ ####GSSAPI#SSPI################### verify-fullSSL############### ####### ######

- hostaddr#####host##################### #PQconnectPoll#### ##PQconnectPoll################PQconnectPoll######################## ####
- host#####hostaddr########hostaddr###################### ############## ##############
- host#hostaddr###########hostaddr#################### host############ ################################

host#hostaddr### # ##host#hostaddr###########host#########(32.16###)##############

#######hostaddr############ ###################### ######################### #### ############################## ###32.1.1.3########

####################libpq######Unix################### Windows###localhost# #########

port

#####################Unix############################## ########host### #hostaddr## ################### ############ ########################PostgreSQL########################### ###

dbname

############# ############### ######################## #######32.1.1##########

user

###########PostgreSQL########### ### ######

password

#######################################

passfile

#####################32.16#### ######~/.pgpass####Microsoft Windows# #%APPDATA%\postgresql\pgpass.conf### ##########################

require_auth

########################## ## ############### ##1############ ## ######################################

####!############## ############################## ########################### ################## ## ######### ################################

############none########################### ########################### ###

############

password

###########################

md5

#####MD5#######################

gss

####GSSAPI####Kerberos##############GSS######################## #gssencmode##########

sspi

####Windows SSPI###############

scram-sha-256

############SCRAM-SHA-256####################

none

########################## #####TLS#################################
GSS#################

channel_binding

################################## require##########################
##prefer############################disable####################
PostgreSQL#SSL#########################prefer##############disable###

####SCRAM########PostgreSQL 11###
#####SSL##############

connect_timeout

#####[58.643326]###############10############10######### ################
########## ##############2#### #####1#2####### ##############IP#######
####### ###############connect_timeout#5##########5##################
##############10############# ###########################10############
#:10## ################################ ##################IP############
#####2#########connect_#######5####5################################
#####################10######

client_encoding

####client_encoding############ ####################################
####Unix########LC_CTYPE##################auto############

options

#####[82.481752]################################ #######-c geqo=off#####
##geqo############off###### #################(\)#########################
######### #################\\######## ####################19########## ##
##########################-############# ########-c geqo=#####--geqo=###
######geqo######################## ################-################# ####
#########################\);write\\################## ###################19
##########

application_name

application_name###############

fallback_application_name

application_name################ ##########PGAPPNAME######
#application_name####################### ##############################
##

keepalives

##########TCP################ #######1################# ################
############################## ########Unix####################

keepalives_idle

TCP## ########################
Unix## ###TCP_KEEPIDLE##
##########################Windows########### #################

`keepalives_interval`

TCP## ##################
Unix###
#TCP_KEEPINTVL#########################Windows########### ###########
#####

`keepalives_count`

##################
Unix###
#TCP_KEEPCNT################################### ################

`tcp_user_timeout`

#0################
Unix################################## TCP_USER_TIMEOUT################
################

`replication`

###PostgreSQL###########
pg_basebackup## ###############
######53.4##########

##############################

`true`, `on`, `yes`, `1`

#####################

`database`

##################dbname######################

`false`, `off`, `no`, `0`

############################

##

`gssencmode`

#########GSS######TCP/IP################################## 3##########

`disable`

#GSSAPI########

`prefer` (#####)

GSSAPI#####(##############)#######GSSAPI########### ##################
#########GSSAPI########### ###PostgreSQL#GSSAPI######################
####

`require`

GSSAPI########

`gssencmode`#Unix################## PostgreSQL#GSSAPI###########
##`require`##################`prefer`##########libpq#####GSSAPI##########
##

sslmode

##########SSL######TCP/IP###################### 6##########

disable

#SSL######

allow

####SSL############SSL#####

prefer #######

###SSL#############SSL#####

require

SSL####### ###CA###########verify-ca#######################

verify-ca

SSL#######################CA###########

verify-full

SSL####################CA####################################

#######################32.19#########

sslmode#Unix################## SSL#######PostgreSQL########## ##require#verify-ca#verify-full############## ###allow#prefer####### #####libpq#SSL##########

#####[86.407767]#GSSAPI#########sslmode#######SSL####################### ### GSSAPI##################(Kerberos#####)#SSL############# ##gssencmode#disable####### ######GSSAPI#########sslmode#######SSL### ######################## GSSAPI###################Kerberos#######SSL######## #######gssencmode#disable#######

requiressl

#######sslmode######################

1##############SSL######### (###sslmode#require#####)# ####SSL######### ##libpq######### 0(#####)########libpq################## (sslmode#prefer#### ##) SSL#######PostgreSQL###########################

sslnegotiation

###############SSL###########SSL############################ #### #POSTGRES###################SSL################### ###############TCP/IP ################SSL############ ###(#################################### ## PostgreSQL#########SSL############SSL################1################## ################SSL######################## ##SSL#######PostgreSQL#####17#### #####

postgres

######PostgreSQL################### ##########################

`direct`

######TCP/IP##############SSL############## ####sslmode=require########## ##### ##############SSL## #########

`sslcompression`

1#########SSL#################### 0################## ######0### ########SSL# ###############

#####[85.948478]#SSL################################### OpenSSL 1.1.0######### #############OS## on############## PostgreSQL 14######################## ######SSL############ ######################## OpenSSL 1.1.0## ####################################on############################## PostgreSQL 14#######################

CPU###################

`sslcert`

#########~/.postgresql/postgresql.crt####################SSL########### ##### ########SSL####################

`sslkey`

####################################### ######~/.postgresql/ postgresql.key################################OpenSSL################## ############### ## SSL###### ####################

`sslpassword`

########sslkey## ##############################

###libpq#########OpenSSL######## #Enter PEM pass phrase:###########

############################## OpenSSL###########OpenSSL#################### ##############################

#########################.pgpass############ ############################ ################OpenSSL#####PKCS#11#USB############################

`sslcertmode`

###3#########

`disable`

#######################sslcert############################

`allow` (default)

##

`require`

#################### ##
######

##
sslcertmode=require## ######## PostgreSQL##############################TLS########## ###### ########TLS#########################

`sslrootcert`

########SSL###(CA)################ ############################1########### ######### ######~/.postgresql/root.crt###

#########system##########################CA############ ################ ##SSL#################### OpenSSL######SSL_CERT_DIR#SSL_CERT_FILE#### ########################

##
sslrootcert=system############sslmode#verify-full############ ######### ##verify-ca####################
#####system########################### ######################### #sslrootcert=./system#################

`sslcrl`

########SSL############CRL############## ################################ ################ sslcrl#sslcrldir##############~/.postgresql/root.crl###### ##

`sslcrldir`

#########SSL###########(CRL)############## ################################ ####################

########OpenSSL####openssl rehash###c_rehash################ ########## ###########

sslcrl#sslcrldir#############

`sslsni`

1(#####)###########libpq#TLS###Server Name Indication#(SNI)#SSL############# ### #####0#####################

###############SSL#######SSL################################ ########### PostgreSQL######################sslnegotiation#direct############# ## ##SNI###

`requirepeer`

############requirepeer=postgres############################## Unix#### ##

###################### #########TCP/IP######SSL################################ #### #Unix#########/tmp## ###### ## ########peer###### ######################### 20.9##########

ssl_min_protocol_version

################SSL/TLS#################### #####TLSv1#TLSv1.1#TLSv1.2### ##TLSv1.3### ##################OpenSSL#################################### ###### #############TLSv1.2#####################

ssl_max_protocol_version

################SSL/TLS#################### #####TLSv1#TLSv1.1#TLSv1.2### ##TLSv1.3### ##################OpenSSL#################################### ####### ## ##### ##

krbsrvname

GSSAPI#########Kerberos######## ################Kerberos##################### #### (20.6##########) #########postgres####PostgreSQL#configure#--with-krb-srvnam######################### ############################ ######### (POSTGRES)##########Microsoft Active Directory#########Kerberos################## ########

gsslib

GSSAPI########GSS######## #########GSSAPI#SSPI###########Windows########## ##### #############SSPI#####GSSAPI##########libpq###########gssapi######## ##

gssdelegation

GSS###########(##)#### ######0########################### ################ #1#######

service

######################## pg_service.conf############################## ##### ## 32.17##########

target_session_attrs

6##########:

any (default)

##############

read-write

#####################################(#########################
##default_transaction_read_only######off##########)#

read-only

#####################################(#)

`primary`

##########################

`standby`

##########################

`prefer-standby`

###any###########

`load_balance_hosts`

##########################IP###DNS################## ########################## ##########target_session_attrs############## ################################### ####### ###2###########

`disable` (default)

################## ####################DNS#########################

`random`

########################## ## ###### ###########PostgreSQL################

1###########2###################### ###################### ############## ################################### ################################### ## ##############################

###########connect_timeout#################### ##########################1 ##########################

32.2.

##

> ###
>
> libpq##################PGconn############### PGconn################## ############ PGconn################################libpq-int.h######## ##############

########################## ################# ########################PGconn## ####################PQhost#PQport#PQpass############## #####PGconn####### ######

`PQdb`

#################

```
char *PQdb(const PGconn *conn);
```

PQuser

```
##############

char *PQuser(const PGconn *conn);
```

PQpass

```
###############

char *PQpass(const PGconn *conn);
```

PQpass######################### ##
################ ####################################PQpass###########
#########PQstatus#######

PQhost

```
################### ########IP########Unix############################ ###
######/#######################

char *PQhost(const PGconn *conn);
```

host#hostaddr##########PQhost####host####### hostaddr###############
#########################PQhost####################

conn###NULL####PQhost#NULL###### ####################################
#####################################

################################PQhost################# #######PQstatus#
########

PQhostaddr

```
##########IP#########  ####################hostaddr#############IP########
##

char *PQhostaddr(const PGconn *conn);
```

conn###NULL####PQhostaddr#NULL###### ################################
###

PQport

```
###############

char *PQport(const PGconn *conn);
```

#########################PQport##################

conn###NULL####PQport#NULL###### ####################################
#####################################

################################PQport################# #######PQstatus#
########

PQtty

```
    ############################## ################# conn###NULL####NULL####
    ##


    char *PQtty(const PGconn *conn);
```

PQoptions

```
    ###########################


    char *PQoptions(const PGconn *conn);
```

#######PGconn################################

PQstatus

```
    ###########


    ConnStatusType PQstatus(const PGconn *conn);

    #####################                          ########################
    ##CONNECTION_OK#CONNECTION_BAD#######              ################
    ##CONNECTION_OK#######   ###########CONNECTION_BAD#######   ###OK##
    #PQfinish####################CONNECTION_BAD##########   #############
    #PQreset##################

    #######################PQconnectStartParams#PQconnectStart##
    #PQconnectPoll############
```

PQtransactionStatus

```
    ########################


    PGTransactionStatusType PQtransactionStatus(const PGconn *conn);

    ######PQTRANS_IDLE (#####)#PQTRANS_ACTIVE (#######)#PQTRANS_INTRANS (###
    ################)#PQTRANS_INERROR (###################)####### ###########
    #PQTRANS_UNKNOWN####### ############################PQTRANS_ACTIVE##
    ######
```

PQparameterStatus

```
    ####################


    const char *PQparameterStatus(const PGconn *conn, const char
     *paramName);

    ################################################## PQparameterStatus###
    ############# ##########################NULL######

    ##########################

    application_name                 is_superuser
    client_encoding                  scram_iterations
    DateStyle                        server_encoding
```

```
default_transaction_read_only   server_version
in_hot_standby                  session_authorization
integer_datetimes               standard_conforming_strings
IntervalStyle                   TimeZone
```

#######default_transaction_read_only#in_hot_standby#14############# ####### scram_iterations#16#################### # ##server_version#server_encoding####integer_datetimes############

standard_conforming_strings#######################off############ #### ############################## #########################(E'...')######### #############

#########const##############PGconn######################## ############## #########################

PQprotocolVersion

############/##################

```
int PQprotocolVersion(const PGconn *conn);
```

#########3(3.0#####)#### #0(####)### ################################### ######################### 3.0 ######PostgreSQL########7.4##############

PQserverVersion

###################

```
int PQserverVersion(const PGconn *conn);
```

####################10000# ###################### #########10.1##100001#########11.0##110000###### ###### ##0#######

#####10#######PostgreSQL######2#################3######################## # ###########PQserverVersion########2######### ##########9.1.5##90105##### #####9.2.0##90200#######

################################PQserverVersion####10000#####100######### ################ #############2################## #############

PQerrorMessage

##

```
char *PQerrorMessage(const PGconn *conn);
```

########libpq####### PQerrorMessage ############# libpq############ PQerrorMessage ## ################ ######## ####PGconn#####PQfinish################ PGconn################## ###################

PQsocket

############################ ##########0##### -1############################ (###################### ##################)

```
int PQsocket(const PGconn *conn);
```

PQbackendPID

##################ID(PID)######

```
int PQbackendPID(const PGconn *conn);
```

#######PID###########NOTIFY##########################PID################ ##PID## #########

PQconnectionNeedsPassword

##################################1####### #######0#######

```
int PQconnectionNeedsPassword(const PGconn *conn);
```

###

PQconnectionUsedPassword

#####################1#########0#######

```
int PQconnectionUsedPassword(const PGconn *conn);
```

##

PQconnectionUsedGSSAPI

####### GSSAPI ##########true (1)###### ##########false (0)######

```
int PQconnectionUsedGSSAPI(const PGconn *conn);
```

#########GSSAPI#######################

######SSL############ #######################

PQsslInUse

###SSL########(1)##########(0)######

```
int PQsslInUse(const PGconn *conn);
```

PQsslAttribute

######SSL##########

```
const char *PQsslAttribute(const PGconn *conn, const char
 *attribute_name);
```

##################SSL#################### ###SSL#################### ##################NULL######

##################

library

#######SSL######## ####"OpenSSL"############

protocol

#######SSL/TLS######## #######"TLSv1"#"TLSv1.1"#"TLSv1.2"#########
##########################

key_bits

#######################

cipher

####################"DHE-RSA-DES-CBC3-SHA"### ######SSL###########

compression

SSL############"on"##############"off"######

alpn

######TLS#######################(ALPN)####################### libpq##
##########postgresql################ALPN##########################
ALPN##################

#########library####conn#####NULL######################### ########
SSL#########libpq#SSL#################NULL###### #PostgreSQL#####15####
##conn#####NULL######NULL###### ##################################
##LIBPQ_HAS_SSL_LIBRARY_DETECTION################

PQsslAttributeNames

PQsslAttribute()######SSL########### ###########NULL#########

```
const char * const * PQsslAttributeNames(const PGconn *conn);
```

conn#NULL#########SSL################### ####libpq#SSL##################
########### conn#NULL###############SSL##################################
############

PQsslStruct

SSL############################## ################################SSL####
#########NULL######

```
void *PQsslStruct(const PGconn *conn, const char *struct_name);
```

###############SSL######### OpenSSL###OpenSSL##############1####OpenSSL
#SSL############## ############################

```
#include <libpq-fe.h>
#include <openssl/ssl.h>

...
```

```
SSL *ssl;

dbconn = PQconnectdb(...);
...

ssl = PQsslStruct(dbconn, "OpenSSL");
if (ssl)
{

    /* ssl#########OpenSSL###### */
}
```

##################################### ################OpenSSL##############
##

PQgetssl

#########SSL######### SSL#########NULL######

```
void *PQgetssl(const PGconn *conn);
```

#####PQsslStruct(conn, "OpenSSL")###### #######OpenSSL########SSL
###SSL####################
#PQsslInUse######### #################PQsslAttribute#######

32.3.

#######################################SQL#################

32.3.1.

PQexec

#####################

```
PGresult *PQexec(PGconn *conn, const char *command);
```

####PGresult#############NULL###### ################################
###############NULL########## PQresultStatus##############NULL#####
############PGRES_FATAL_ERROR########################## #############
PQerrorMessage ##########

#########################SQL################ ###PQexec####################
################ #################BEGIN/COMMIT############################
##############################53.2.2.1########## #######PGresult###########
################################## #######1################################
#PGresult#######

PQexecParams

#################### ####SQL###########################

```
PGresult *PQexecParams(PGconn *conn,
                       const char *command,
                       int nParams,
                       const Oid *paramTypes,
                       const char * const *paramValues,
```

```
                        const int *paramLengths,
                        const int *paramFormats,
                        int resultFormat);
```

PQexecParams#PQexec###################### ########################### ##

###############

conn

##########################

command

#####SQL######### ######################$1#$2############

nParams

############# ####
#paramTypes[]#paramValues[]#paramLengths[]#paramFormats[]#####
##########*nParams*#0####NULL###########

paramTypes[]

###################OID######### *paramTypes*#NULL###########0#######
###

paramValues[]

################ ####NULL##############NULL########### #############
##

paramLengths[]

######################## NULL############################ #############
#############NULL##########

paramFormats[]

#########################0#########################1########### #######
##NULL##############################

##################################### ##############################
numeric#####src/backend/utils/adt/numeric.c::numeric_send()##
#src/backend/utils/adt/numeric.c::numeric_recv()###############
#############

resultFormat

##################0#################1####### ##########################
###########################

PQexec####PQexecParams#################################### ###############
############################

PQexec#####PQexecParams##########1##SQL############### ##################
########2############### #################SQL##############################
####

> **###**
>
> OID##########################OID############################## ######### ## SQL############# ## #######
>
> ```
> SELECT * FROM mytable WHERE x = $1::bigint;
> ```
>
> ###########$1###x##########################bigint######## ######### OID## ######### ##

PQprepare

###################################

```
PGresult *PQprepare(PGconn *conn,
                    const char *stmtName,
                    const char *query,
                    int nParams,
                    const Oid *paramTypes);
```

PQprepare####PQexecPrepared#################### ######################### ######################### ####PREPARE#########

#####query#####stmtName################# *query*####SQL############## *stmtName*#""################### ############################### #### ################################ ########################$1#$2######## *nParams*######### ##########*paramTypes[]*########### (*nParams*##### ##########NULL##########) *paramTypes[]*##OID########################## ## *paramTypes*#NULL## ######################## #########*nParams*######################## #### ##################### ##########################PQdescribePrepared### #######

PQexec#######PGresult########################## NULL################ ##################### ################## PQerrorMessage #########

######PQexecPrepared###############SQLPREPARE################

PQexecPrepared

################################

```
PGresult *PQexecPrepared(PGconn *conn,
                         const char *stmtName,
                         int nParams,
                         const char * const *paramValues,
                         const int *paramLengths,
                         const int *paramFormats,
                         int resultFormat);
```

PQexecPrepared#PQexecParams## ################### ## ## ##############################

```
    ########################################PQexecParams######            #
    ##paramTypes[]############ (####################################)

PQdescribePrepared

    ##################################


    PGresult *PQdescribePrepared(PGconn *conn, const char *stmtName);

    PQdescribePrepared##################################

    stmtName#""###NULL#######################    ###########################
    ###     ######PGRES_COMMAND_OK########PGresult######     PQnparams##
    #PQparamtype#####PGresult################################           #
    ##PQnfields#PQfname#PQftype##################################

PQdescribePortal

    ################################


    PGresult *PQdescribePortal(PGconn *conn, const char *portalName);

    PQdescribePortal######################################## #libpq###########
    ##########################DECLARE CURSOR SQL############################

    portalName#""###NULL#######################  ##########################
    ##                      ######PGRES_COMMAND_OK########PGresult######
    PQnfields#PQfname#PQftype#######PGresult############################
    ########

PQclosePrepared

    #####[54.651163]################################### ######################
    ##############


    PGresult *PQclosePrepared(PGconn *conn, const char *stmtName);

    ######PQclosePrepared################################### ##############
    ################################

    ######stmtName#""###NULL############### ########################## #####
    #######PGRES_COMMAND_OK###PGresult######

PQclosePortal

    ##################################


    PGresult *PQclosePortal(PGconn *conn, const char *portalName);

    #####[58.578431]#PQdescribePortal########################################
    #libpq####################################DECLARE CURSOR SQL##############
    ############### ######PQclosePortal#####################################
    ################################################### #libpq##################
    ############DECLARE CURSORSQL##########################

    ######portalName#""###NULL################## ###########################
    ### ############PGRES_COMMAND_OK###PGresult######
```

PGresult######################## libpq##################PGresult########### #### ###############PGresult############# ##############PGresult########## ###############

PQresultStatus

##############

```
ExecStatusType PQresultStatus(const PGresult *res);
```

PQresultStatus###############

PGRES_EMPTY_QUERY

#################

PGRES_COMMAND_OK

####################

PGRES_TUPLES_OK

##########(SELECT#SHOW##)##########

PGRES_COPY_OUT

(######)##################

PGRES_COPY_IN

(#####)#################

PGRES_BAD_RESPONSE

###############

PGRES_NONFATAL_ERROR

#######(##########)###########

PGRES_FATAL_ERROR

##############

PGRES_COPY_BOTH

###################/############### ############################## ###### ######################

PGRES_SINGLE_TUPLE

PGresult######################## #########################32.6######## ###

PGRES_TUPLES_CHUNK

#####[50.943396]#PGresult######################## ################### #####32.6########## ######PGresult######################## ##########

```
        ############################32.6##########                          ####
        #PQsetChunkedRowsMode##################

    PGRES_PIPELINE_SYNC

        #####[68.992248]#PGresult##PQpipelineSync###########################
        ##      ##############################      ######PGresult##########
        #PQpipelineSync###PQsendPipelineSync################ ############
        ###################

    PGRES_PIPELINE_ABORTED

        PGresult######################### PQgetResult################### ##
        ############################PGRES_PIPELINE_SYNC################

    #####[80.310078]######PGRES_TUPLES_OK###PGRES_SINGLE_TUPLE############
    ###########################             #######SELECT###################
    #PGRES_TUPLES_OK############## PGRES_COMMAND_OK##########(RETURNING#
    ###INSERT###UPDATE##)####### PGRES_EMPTY_QUERY#######################
    ##########          ###########PGRES_TUPLES_OK#PGRES_SINGLE_TUPLE###
    #PGRES_TUPLES_CHUNK############################################ ########
    ###SELECT######PGRES_TUPLES_OK############## PGRES_COMMAND_OK########
    #########RETURNING####INSERT#UPDATE############ PGRES_EMPTY_QUERY###
    ##############################

    PGRES_NONFATAL_ERROR########PQexec######################## ##########
    ############(32.13###)#######

PQresStatus

    PQresultStatus############################ ####################


    char *PQresStatus(ExecStatusType status);

PQresultErrorMessage

    #################### ####################


    char *PQresultErrorMessage(const PGresult *res);

    ##############################  #######################  ####PGresult####
    #PQclear################

    #######     PQerrorMessage     ##PQexec###PQgetResult##############
    ##PQresultErrorMessage###########   ################################
    #####PGresult############################   ##PQresultErrorMessage###
    #PGresult################### PQerrorMessage #########################
    ####

PQresultVerboseErrorMessage

    PGresult######################################


    char *PQresultVerboseErrorMessage(const PGresult *res,
                                      PGVerbosity verbosity,
                                      PGContextVisibility show_context);

    ##################################################
    PQresultVerboseErrorMessage#####PGresult############################
```

##PQresultErrorMessage################################ PGresult#######
########PGresult is not an error result############ ######################

PGresult## ####################
##PQfreemem()##################

############NULL#############

PQresultErrorField

###################

```
char *PQresultErrorField(const PGresult *res, int fieldcode);
```

fieldcode############## ################## PGresult###############
###########################NULL###### ################## ##########
#PGresult#####PQclear#############

##################

PG_DIAG_SEVERITY

##########(##########)ERROR#FATAL######PANIC#(##########
#)WARNING#NOTICE#DEBUG#INFO######LOG### ############################
########

PG_DIAG_SEVERITY_NONLOCALIZED

##########(##########)ERROR#FATAL######PANIC#(##########
#)WARNING#NOTICE#DEBUG#INFO######LOG### ####################
##PG_DIAG_SEVERITY###### ###PostgreSQL######9.6##################

PG_DIAG_SQLSTATE

####SQLSTATE###### SQLSTATE################### ####################
#####SQLSTATE############A##
####### ############################

PG_DIAG_MESSAGE_PRIMARY

################### (###1####) ########

PG_DIAG_MESSAGE_DETAIL

###############

PG_DIAG_MESSAGE_HINT

######(detail)################(##############)##
################### ##############

PG_DIAG_STATEMENT_POSITION

#####################################10########### ##########1#########
#################

PG_DIAG_INTERNAL_POSITION

#####PG_DIAG_STATEMENT_POSITION########## ########################
################################ ##############
#PG_DIAG_INTERNAL_QUERY###########

PG_DIAG_INTERNAL_QUERY

################### ########PL/pgSQL########SQL###########

PG_DIAG_CONTEXT

################# ## ##### ####1#######################

PG_DIAG_SCHEMA_NAME

##

PG_DIAG_TABLE_NAME

############################ ####################################

PG_DIAG_COLUMN_NAME

########################## ####################################

PG_DIAG_DATATYPE_NAME

############################ ####################################

PG_DIAG_CONSTRAINT_NAME

######################## ## ### ##

PG_DIAG_SOURCE_FILE

##########################

PG_DIAG_SOURCE_LINE

###########################

PG_DIAG_SOURCE_FUNCTION

########################

##
##A######## ## ## ##################### ##################################### ################### ########### ############################

######################### ############################ ##################### ##############################

libpq##

#########PGresult########PGconn########### PQerrorField###########

PQclear

PGresult################## ####################PQclear##########

```
void PQclear(PGresult *res);
```

###NULL##################

PGresult###################### ##############################PGresult#### ### PGresult#######PQclear############## ################################ ##########

32.3.2.

#####[87.091503]############################(#########PGRES_TUPLES_OK## #PGRES_SINGLE_TUPLE######)PGresult#################### #######Describe## ################ Describe## # ###########################0#0############### ############################## #PGresult#####PGRES_TUPLES_OK#PGRES_SINGLE_TUPLE### #PGRES_TUPLES_CHUNK########################## #######Describe############## ######## Describe##0### ############## ###############0##0##############

PQntuples

##########(###)####### # PGresult######INT_MAX###########int#########

```
int PQntuples(const PGresult *res);
```

PQnfields

############(#####)########

```
int PQnfields(const PGresult *res);
```

PQfname

#################### ####0######## ####################### ####PGresult### ##PQclear################

```
char *PQfname(const PGresult *res,
              int column_number);
```

##############NULL######

PQfnumber

####################

```
int PQfnumber(const PGresult *res,
              const char *column_name);
```

##################-1######

#######SQL################ ############################## #######SQL####### ############

```
SELECT 1 AS FOO, 2 AS "BAR";
```

###################

```
PQfname(res, 0)              foo
PQfname(res, 1)              BAR
PQfnumber(res, "FOO")        0
PQfnumber(res, "foo")        0
PQfnumber(res, "BAR")        -1
PQfnumber(res, "\"BAR\"")    1
```

PQftable

################OID###### ####0########

```
Oid PQftable(const PGresult *res,
             int column_number);
```

##################################InvalidOid####### pg_class############ ################################

libpq#################Oid##InvalidOid######### #################

PQftablecol

###################(############)######### ############0################0### ############

```
int PQftablecol(const PGresult *res,
                int column_number);
```

###

PQfformat

#################### ####0########

```
int PQfformat(const PGresult *res,
              int column_number);
```

#######################1################### (#######################)

PQftype

#################### ##############OID##### ####0########

```
Oid PQftype(const PGresult *res,
            int column_number);
```

pg_type#################################### #########OID##PostgreSQL#### ######include########catalog/pg_type_d.h##############

PQfmod

####################### ####0########

```
int PQfmod(const PGresult *res,
           int column_number);
```

##################### ##################### -1######################### #########
############ #######-1##########

PQfsize

############################ ####0########

```
int PQfsize(const PGresult *res,
            int column_number);
```

PQfsize############################ ################################ (##
############################) ################

PQbinaryTuples

PGresult#############1###############0######

```
int PQbinaryTuples(const PGresult *res);
```

############ (COPY###############) ###PGresult#################### ######################### PQfformat########## ############(##1)#### #PQbinaryTuples#1######

PQgetvalue

PGresult#1################### ########0######## ######################### ### #PGresult#####PQclear#################

```
char *PQgetvalue(const PGresult *res,
                 int row_number,
                 int column_number);
```

############PQgetvalue#############NULL############# ################ #####typsend###typreceive################# (############################ ### ################NULL##############################)

#######NULL############### NULL##################PQgetisnull##########

PQgetvalue############PGresult################## ######################### ##### ###PGresult##

PQgetisnull

######NULL########### ########0########

```
int PQgetisnull(const PGresult *res,
                int row_number,
                int column_number);
```

############NULL####1########NULL#######0###### (PQgetvalue###NULL##### #NULL##########################)

PQgetlength

######################### ########0########

```
int PQgetlength(const PGresult *res,
                int row_number,
                int column_number);
```

######################## ####PQgetvalue##################### ##########
#strlen()###### ################# ##############PQfsize###########

PQnparams

##################

```
int PQnparams(const PGresult *res);
```

#####PQdescribePrepared################ ################

PQparamtype

#################### ########0########

```
Oid PQparamtype(const PGresult *res, int param_number);
```

######PQdescribePrepared################ ################

PQprint

########(####)##################

```
void PQprint(FILE *fout,      /* ####### */
             const PGresult *res,
             const PQprintOpt *po);
typedef struct
{
    pqbool  header;      /* ################## */
    pqbool  align;       /* ################### */
    pqbool  standard;    /* ############ */
    pqbool  html3;       /* HTML### */
    pqbool  expanded;    /* ###### */
    pqbool  pager;       /* ############# */
    char    *fieldSep;   /* ########## */
    char    *tableOpt;   /* HTML###### */
    char    *caption;    /* HTML #### */
    char    **fieldName; /* ############NULL##### */
} PQprintOpt;
```

#######################psql######################### ######################
##################

32.3.3.

#######PGresult###########################

PQcmdStatus

PGresult#####SQL##################

```
char *PQcmdStatus(PGresult *res);
```

##################################### ####################### ###
#PGresult#####PQclear################

PQcmdTuples

SQL####################

```
char *PQcmdTuples(PGresult *res);
```

#####PGresult#####SQL######################### #####SELECT#CREATE TABLE AS#INSERT#UPDATE#DELETE#MERGE#MOVE#FETCH#COPY#######
##INSERT#UPDATE#DELETE#MERGE#############EXECUTE################
PGresult######################PQcmdTuples########## ##################
####PGresult#####PQclear################

PQoidValue

SQL######OID########1#######INSERT#############INSERT############
#EXECUTE#############OID###### #####InvalidOid###### ###INSERT#########
###OID##############InvalidOid######

```
Oid PQoidValue(const PGresult *res);
```

PQoidStatus

#####PQoidValue############# ################## #########OID#########
##PQoidValue#OID######

```
char *PQoidStatus(const PGresult *res);
```

32.3.4. SQL#######################

PQescapeLiteral

```
char *PQescapeLiteral(PGconn *conn, const char *str, size_t length);
```

PQescapeLiteral##SQL######################## ####SQL################
############ #####(###########)##SQL#################################
PQescapeLiteral##########

PQescapeLiteral#*str*###############malloc()############### #########
########PQfreemem()################ ############*length*############
(*length*######################PQescapeLiteral########## #####strncpy#
######) #########PostgreSQL## #####
######### PostgreSQL######################################

#####PQescapeLiteral#NULL####*conn*####################

> ###
>
> ## ##################
> #### #SQL#########################SQL##################

PQexecParams###
#########

PQescapeIdentifier

```
char *PQescapeIdentifier(PGconn *conn, const char *str, size_t length);
```

PQescapeIdentifier##############SQL########################## ######## ############SQL## ###########################

PQescapeIdentifier#*str*######SQL###############malloc()############# ## ##################PQfreemem()################# ############*length*### ########## (*length*#####################PQescapeIdentifier########### #####strncpy######) #########SQL################################### ## ############ ######################

#####PQescapeIdentifier#NULL####*conn*####################

> ###
>
> ###########SQL##################################SQL################ ####

PQescapeStringConn

```
size_t PQescapeStringConn(PGconn *conn,
                          char *to, const char *from, size_t length,
                          int *error);
```

PQescapeStringConn##PQescapeLiteral#################### PQescapeLiteral#################################### ## #PQescapeStringConn#PostgreSQL################################ ####### SQL######################## *from*############################## *length*# ################### ###############*length*############## (##*length*## ###################PQescapeStringConn########### #####strncpy######) *to*#####*length*#2####1################################ ################ *to*#*from*##################

error######NULL######*error#####0######0######### ##################### ##################### ## ####### #######*error*#NULL#########*conn*##############

PQescapeStringConn#*to*############### #######################

PQescapeString

PQescapeString#PQescapeStringConn#############

```
size_t PQescapeString (char *to, const char *from, size_t length);
```

PQescapeStringConn########PQescapeString#PGconn#*error*############ ## ####(###########)################### ################### ############## ########

PQescapeString#####1##PostgreSQL############################## (######### ############################) #################PQescapeStringConn###### ##############

PQescapeByteaConn

bytea####SQL######################### PQescapeStringConn#######SQL## ###########################

```
unsigned char *PQescapeByteaConn(PGconn *conn,
                                 const unsigned char *from,
                                 size_t from_length,
                                 size_t *to_length);
```

SQL###bytea## PQescapeByteaConn# 16## ####8.4##########

from################################## *from_length*#################################### (#################################### *to_length*#################################### ### ##########################

PQescapeByteaConn##*from*##########################malloc()############ ## ##################PQfreemem()################## ##########PostgreSQL#### #######bytea#################################### ############### PostgreSQL##############################

#####NULL##################*conn*############ ##########################

PQescapeBytea

PQescapeBytea##PQescapeByteaConn#############

```
unsigned char *PQescapeBytea(const unsigned char *from,
                             size_t from_length,
                             size_t *to_length);
```

PQescapeBytea#PQescapeByteaConn########PGconn####### ### #PQescapeBytea#######PostgreSQL################################## (##### ################################) ## #########PQescapeByteaConn##########

PQunescapeBytea

########################### ####PQescapeBytea##### ####bytea############ ############# ######################

```
unsigned char *PQunescapeBytea(const unsigned char *from, size_t
 *to_length);
```

from###########bytea##PQgetvalue################################ PQunescapeBytea#################################### malloc()#################### #### #NULL### #############*to_length*####### ############PQfreemem########### #######

######PQescapeBytea######### ####PQgetvalue########################### ## ############################PGconn#####################

32.4.

PQexec#################################### ################################ ##

- PQexec ################# #############(####################)############## #### ####################

- ## (### ################################)

- PQexec########PGresult####1##### #################SQL########## ##PQexec##################PGresult##########

- PQexec###################PGresult####### ################################ ##################

#####[74.937028]##################################PQexec#####
#PQsendQuery#PQgetResult######### #
##PQsendQueryParams#PQsendPrepare#PQsendQueryPrepared#PQsendDescribePrepared#PQs
###PQgetResult########
#PQexecParams#PQprepare#PQexecPrepared#PQdescribePrepared#PQdescribePortal#
############### ########################PQexec########
#PQsendQuery#PQgetResult############
PQsendQueryParams#PQsendPrepare#PQsendQueryPrepared#PQsendDescribePrepared#PQsen
####PQgetResult######
##PQexecParams#PQprepare#PQexecPrepared#PQdescribePrepared##########
PQdescribePortal PQclosePrepared, and PQclosePortal respectively.

PQsendQuery

######################### #############1########0####### (######
PQerrorMessage ############################)

```
int PQsendQuery(PGconn *conn, const char *command);
```

PQsendQuery##########PQgetResult#################### PQgetResult#
NULL#########################(#####)PQsendQuery###############

#########################

PQsendQueryParams

##################################

```
int PQsendQueryParams(PGconn *conn,
                      const char *command,
                      int nParams,
                      const Oid *paramTypes,
                      const char * const *paramValues,
                      const int *paramLengths,
                      const int *paramFormats,
                      int resultFormat);
```

#####################################PQsendQuery##### ##########
#PQexecParams######### PQexecParams############1###############

PQsendPrepare

########################### ###########

```
int PQsendPrepare(PGconn *conn,
                  const char *stmtName,
                  const char *query,
```

```
                    int nParams,
                    const Oid *paramTypes);
```

###PQprepare####### ###########1#######0###### ######################
#############PQgetResult########## ##########PQprepare#########

PQsendQueryPrepared

####################################

```
int PQsendQueryPrepared(PGconn *conn,
                        const char *stmtName,
                        int nParams,
                        const char * const *paramValues,
                        const int *paramLengths,
                        const int *paramFormats,
                        int resultFormat);
```

###PQsendQueryParams## ##
#########PQexecPrepared#########

PQsendDescribePrepared

######################### #############

```
int PQsendDescribePrepared(PGconn *conn, const char *stmtName);
```

###PQdescribePrepared####### ###########1#######0###### #########
##PQgetResult################# ##########PQdescribePrepared#########
##

PQsendDescribePortal

######################### ##########

```
int PQsendDescribePortal(PGconn *conn, const char *portalName);
```

###PQdescribePortal####### ###########1#######0###### #########
##PQgetResult################# ##########PQdescribePortal###########

PQsendClosePrepared

#####[59.782609]######################### ############ ####################
#############

```
int PQsendClosePrepared(PGconn *conn, const char *stmtName);
```

#####[85.230769]####PQdescribePrepared####### ###########1#######0#####
##########PQgetResult#################
#PQdescribePrepared########### ########PQclosePrepared####### ##
#########1#########0###### #######PQgetResult############## #########
#PQclosePrepared###########

PQsendClosePortal

####################################

```
int PQsendClosePortal(PGconn *conn, const char *portalName);
```

#####[85.046729]####PQdescribePortal####### ###########1########0###### # ##########PQgetResult################# ##########PQdescribePortal##### ###### ########PQclosePortal####### ##########1#########0###### ####### ############PQgetResult####### ##########PQclosePortal###########

PQgetResult

#####[67.238422]#######
#PQsendQuery#PQsendQueryParams#PQsendPrepare#PQsendQueryPrepared#PQsendDescri
################## ####################NULL########## #######
#PQsendQuery#PQsendQueryParams#PQsendPrepare#PQsendQueryPrepared#PQsendDescri
###PQpipelineSync############### ####################NULL##########

```
PGresult *PQgetResult(PGconn *conn);
```

PQgetResult###########NULL########################## (###############
##PQgetResult###NULL#########) PQgetResult##NULL###########
#PGresult######################## ############################
#PQclear################### ####################PQconsumeInput #####
##########PQgetResult##################

###########PQgetResult################### ########################(###
#########)##############PGRES_PIPELINE_ABORTED###############NULL###
####### ###############PGRES_PIPELINE_SYNC########## #################
######(#########NULL##########)#

##
PQresultStatus##################libpq################NULL###### #####PQgetResult##############

PQsendQuery#PQgetResult#####PQexec####1###### ###########SQL########## ############################## ################### ########################### ######################################

#####[88.059701]#PQsendQuery###PQgetResult################################1# ######## ######32.6####### ######PQsendQuery#PQgetResult################## ################## #######32.6#######

######SQL#######################PQgetResult######################### ######2 ########################

PQconsumeInput

###########################

```
int PQconsumeInput(PGconn *conn);
```

PQconsumeInput ############1################0########### PQerrorMessage ########### ## #### PQconsumeInput ###############PQisBusy#########PQnotifies###### ##################

PQconsumeInput ## ############# #################select#################### ########## #PQconsumeInput ####select()## ##

PQisBusy

#####1#####PQgetResult##### 0#####
####PQgetResult#####

```
int PQisBusy(PGconn *conn);
```

PQisBusy##### #########PQconsumeInput ########### ##
##################

###3#################select()####poll()################################
####1######################### ####select()######PQsocket#########
######################### ##############################PQconsumeInput ##
########### ####PQisBusy####PQisBusy##(0)#####PQgetResult########### #
##PQnotifies######NOTIFY#####32.9################

#####[92.702170]####PQsendQuery/PQgetResult################################
########### 32.7######### ####PQcancel#################PQgetResult######
##################### #####################################

################################
######### ####################SQL############################ (###########
#COPY IN####################) ##
##

PQsetnonblocking

################

```
int PQsetnonblocking(PGconn *conn, int arg);
```

arg#1################### *arg*#0############## #######0#######-1######

#########PQsendQuery#PQputline#PQputnbytes#PQputCopyData##
#PQendcopy## ############
##################

PQexec########################### ########################

PQisnonblocking

####################

```
int PQisnonblocking(const PGconn *conn);
```

##############1###########0######

PQflush

############################## ###(############)#0###### ##############-1####
######################1###### (######################)

```
int PQflush(PGconn *conn);
```

#########################PQflush########## 1################################
###########PQflush########### ###########PQconsumeInput
##PQflush########### ###PQflush#0############## (###NOTICE###############

##
#PQconsumeInput##################)PQflush#0##############################
############

32.5.

libpq## #############
###1#######################/##################################

###
#######################################

##
################################

libpq#######API##PostgreSQL 14##v3######
########################## ########53.2.4##########

32.5.1.

#####[86.274510]## ##
#PQenterPipelineMode###### PQpipelineStatus###########################
####### ###########################SQL###############COPY#########
PQfn#PQexec#PQexecParams#PQprepare#PQexecPrepared#PQdescribePrepared#PQdescribeP
############################# PQsendQuery##################################
##PQexitPipelineMode######
################### ## ##
#PQenterPipelineMode###### PQpipelineStatus###########################
##SQL####################COPY##
#######
PQfn#PQexec#PQexecParams#PQprepare#PQexecPrepared#PQdescribePrepared#PQdescribeP
########################### ####################PQsendQuery######### ######
##PQexitPipelineMode########
#############

##
##########libpq######################## ####################/############### ######### [3]

32.5.1.1.

#####[80.000000]#######################PQsendQueryParams###################
###PQsendQueryPrepared############# ####################################
####PQpipelineSync################################PQflush#############
PQsendPrepare#PQsendDescribePrepared#PQsendDescribePortal#########
########## ############### ############################PQsendQueryParams#
##############PQsendQueryPrepared#################   ####################
###################   ####PQpipelineSync#########################PQflush#
############ #
#PQsendPrepare#PQsendDescribePrepared#PQsendDescribePortal#PQsendClosePrepared#P
############### ##############

[3] ### ##########################
##

#####[86.711712]################################ ############################## ############## ########################;#######PQpipelineSync########### #PQsendFlushRequest########################## ############################## ##################################### ###########PGRES_PIPELINE_ABORTED## #######(####################################)# ###################### ######## ############################## ### ### ########################## ####PQpipelineSync#PQsendPipelineSync######## #PQsendFlushRequest############################## ########################## ##################################### ###########PGRES_PIPELINE_ABORTED#### ####### ## ##################

1########################## #####1## ## ##

32.5.1.2.

#######1##########################PQgetResult#########PQgetResult#NULL# ############# ####################PQgetResult##################### ######## ################### ########################PQgetResult### #PGRES_PIPELINE_SYNC##########

32.5.1.4##########

#####[86.036036]#PQgetResult######################## #PGresult#PGRES_PIPELINE_SYNC#PGRES_PIPELINE_ABORTED############ PGRES_PIPELINE_SYNC##################PQpipelineSync###1######### ####### ##################PGRES_PIPELINE_ABORTED########PGRES_PIPELINE_SYNC##### ########### 32.5.1.3######### ######PQgetResult######################## #PGresult#PGRES_PIPELINE_SYNC#PGRES_PIPELINE_ABORTED########## PGRES_PIPELINE_SYNC################PQpipelineSync## #PQsendPipelineSync###1######### ###########PGRES_PIPELINE_SYNC######## ####################PGRES_PIPELINE_ABORTED######## 32.5.1.3##########

################PQisBusy#PQconsumeInput############## ########### #PQisBusy#######################################0######

libpq#####################################(PQgetResult#NULL################# #########)# ## ##################### FIFO##########

32.5.1.3.

#####[84.826325]#############PQresultStatus#PGRES_FATAL_ERROR############ ############### PQresultStatus######################## #PGRES_PIPELINE_ABORTED######## PQpipelineSync### #PGRES_PIPELINE_SYNC## ################ ###PQresultStatus#PGRES_FATAL_ERROR############################### ##### #####################PQresultStatus#PGRES_PIPELINE_ABORTED########## PQpipelineSync###PQsendPipelineSync################################### #PGRES_PIPELINE_SYNC##########

##############PQgetResult################

##################################(########BEGIN######COMMIT)########## ## ## ##################################

```
############ ##################################################ROLLBACK########
####################################
```

```
                                    ##

########COMMIT——############################ ##########################
########### ##############################################################
#######################
```

32.5.1.4.

```
################################select#poll#WaitForMultipleObjectEx#####
##############################################
```

```
##################################################################### #########
##########################  ##############################################
####### ######################################:##############################
## ##########(#############)#########1################################ ########
##################################1#########################
```

```
###############select()##################PostgreSQL######src/test/
modules/libpq_pipeline/libpq_pipeline.c######
```

32.5.2.

```
PQpipelineStatus
```

libpq#####################

```
PGpipelineStatus PQpipelineStatus(const PGconn *conn);
```

```
PQpipelineStatus###################
```

```
PQ_PIPELINE_ON
```

libpq##############

```
PQ_PIPELINE_OFF
```

libpq##################

```
PQ_PIPELINE_ABORTED
```

libpq#####################################
PQgetResult#PGRES_PIPELINE_SYNC#####################

```
PQenterPipelineMode
```

```
#####################################################
```

```
int PQenterPipelineMode(PGconn *conn);
```

```
#######1###### ##################################################0########
####### ######################libpq###########
```

PQexitPipelineMode

##

```
int PQexitPipelineMode(PGconn *conn);
```

#######1###### ###############1########### ###################
#PQgetResult##0#####(############
######### PQerrorMessage ######)#

PQpipelineSync

######################
################ 32.5.1.3##########

```
int PQpipelineSync(PGconn *conn);
```

#######1###### ##################################0######

PQsendPipelineSync

#####[86.428571]## ######
################################## 32.5.1.3########## ###################
#################################### ####################################
32.5.1.3##########

```
int PQsendPipelineSync(PGconn *conn);
```

#####[50.545455]########1###### ################################0###### ####
#########1###### ######################################0###### #############
###################### ######PQflush##########

PQsendFlushRequest

#####[86.206897]########################### ################################

```
int PQsendFlushRequest(PGconn *conn);
```

#####[81.250000]########1###### #######0###### #############1###### #######0####
##

#####[86.473430]######PQpipelineSync#####################################
################### ##
######PQflush##########
##PQpipelineSync###
###################
############## ######PQflush##########

32.5.3.

###################
###
##########################

################################(#ping###)################################
##################/##

#######300##########100################################30####### #############
###############0.3###########

#########COPY###############INSERT#UPDATE#DELETE##########################
############

##################################1################################## #########
##################
###################### ####-##-################## ##############

```
BEGIN;
SELECT x FROM mytable WHERE id = 42 FOR UPDATE;
-- result: x=2
-- client adds 1 to x:
UPDATE mytable SET x = 3 WHERE id = 42;
COMMIT;
```

could be much more efficiently done with:

```
UPDATE mytable SET x = x + 1 WHERE id = 42;
```

##32.5.1.3#####

32.6. Retrieving Query Results in Chunks

#####[76.381910]####libpq#SQL##################PGresult################ #####
########################## ###############PQsendQuery#PQgetResult#######
########### ## ########libpq#SQL#
##################PGresult################ ######################## ########
##########################PQsendQuery#PQgetResult############ ##########
###################1################################

#####[73.358586]##############PQsendQuery#####################
#PQsetSingleRowMode########## ############################ ####32.4#####
##NULL#########PQgetResult############## ##############
##PGRES_TUPLES_OK####PGRES_SINGLE_TUPLE############################
#PGresult########### ##########################PGRES_TUPLES_OK########
########## ##################### ####NULL#####PQgetResult##############
########## PGresult#######################PGresult####################
###########PQclear############ #######################PQsendQuery
######## #######PQsetSingleRowMode###PQsetChunkedRowsMode####### ##
######################### ###PQgetResult#32.4###########NULL#############
##################1####PGresult##############
##PGRES_SINGLE_TUPLE############PGRES_TUPLES_OK##############
#PGRES_TUPLES_CHUNK##### #PGRES_SINGLE_TUPLE#######1########
##PGRES_TUPLES_CHUNK############1######################## ##############
################PGRES_TUPLES_OK################## ########################
#####NULL #######PQgetResult######################## ####PGresult########
##########PGresult########################### ###########PQclear#########
###

#####[90.763052]#########################PQgetResult#########################
######################### ###32.5#########

PQsetSingleRowMode

  ###########################

```
int PQsetSingleRowMode(PGconn *conn);
```

#####PQsendQuery##################PQconsumeInput #PQgetResult#####
########################## ##
################################

PQsetChunkedRowsMode

#####[70.370370]########################## ##############################
##

```
int PQsetChunkedRowsMode(PGconn *conn, int chunkSize);
```

#####[60.220994]######PQsendQuery##################PQconsumeInput
#PQgetResult################################ ######################
######################## ########################## #############
################## ##########PQsetSingleRowMode#######PGresult####
#*chunkSize*############# ####1########## ######PQsendQuery##############
#PQconsumeInput #PQgetResult########################## ##############
##############################1###### #####################0###### #######
#######################

##
#####[85.390071]### # ##libpq####################### ################################### ##### ########PGRES_SINGLE_TUPLE###PGresult############# #PGRES_FATAL_ERROR################ ############################## ## ################## ################################ ###libpq####################### # ## ############# #PGRES_SINGLE_TUPLE###PGRES_TUPLES_CHUNK#PGresult####### #PGRES_FATAL_ERROR################ ############################## ##

32.7.

32.7.1. Functions for Sending Cancel Requests

PQcancelCreate

##########################

```
PGcancelConn *PQcancelCreate(PGconn *conn);
```

######PQcancelCreate#PGcancelConn################################
############ ########PQcancelBlocking##############PQcancelStart####
######################## ####PQcancelStatus#####PGcancelConn#########
################### PGcancelConn#######################################
##PGcancelConn##

####################
##########sslmode###gssencmode##

############

#####[59.735974]#PQconnectStart#PQconnectStartParams##NULL########### ################################PQfinish####################### ######## ##################################### ######PQcancelCreate#NULL###### ##############################PQcancelFinish############ ############### ####################

PQcancelBlocking

#####[67.415730]####################### ################################### ####

```
int PQcancelBlocking(PGcancelConn *cancelConn);
```

###########PQcancelCreate##############PGcancel######### ############ ########PQcancelBlocking###1### ############### PQcancelErrorMessage ###########

#####[78.917379]## ############### ################################ ## ###################### ################################## ############### ########################### ########### ################################### ##########

PQcancelStart
PQcancelPoll

#####[64.516129]####################### ################################### #######

```
int PQcancelStart(PGcancelConn *cancelConn);

PostgresPollingStatusType PQcancelPoll(PGcancelConn *cancelConn);
```

###########PQcancelCreate##################PGcancel######### ######## ######PQcancelStart###1### ################### PQcancelErrorMessage ### ########

#####[74.797943]#PQconnectStart####PQconnectStartParams############### ########libpq######### ############################PQsocket(conn)##### ## ######PQconnectPoll############################ ######### ## #PQconnectPoll(conn)#PGRES_POLLING_READING#####select()#poll()#### ################################ ######PQconnectPoll(conn)######## ##### #PQconnectPoll(conn)#PGRES_POLLING_WRITING######################### ##PQconnectPoll(conn)######### ##############PQconnectPoll######### #####PGRES_POLLING_WRITING############## ###### #PQconnectPoll(conn)###########PGRES_POLLING_FAILED############### ####PGRES_POLLING_OK########## ######PQcancelStart############libpq### ############################## ############################ ##PQcancelSocket###### ###: #####PQcancelPoll######################## ## ############ PQcancelPoll(cancelConn)####PGRES_POLLING_READING##### ########################(select()#poll()##############)# ##### #PQcancelPoll(cancelConn)####### ###PQcancelPoll(cancelConn)### #PGRES_POLLING_WRITING############################ #PQcancelPoll(cancelConn)####### ########## #PQcancelPoll(cancelConn)################PGRES_POLLING_WRITING####

```
############                  ##############PGRES_POLLING_FAILED######
##PGRES_POLLING_OK##########################################

#####[78.917379]############################################ ##############
################################  ########################################
###################### ################################### ###############
########################## ########### ###################################
##########

#####[85.043988]#############PQstatus##########################   #######
#CONNECTION_BAD##################CONNECTION_OK################ #######
##########PQconnectPoll###############  ################################
## ################################################### ########### #######
########PQcancelStatus#################### #######CONNECTION_BAD######
################ #######CONNECTION_OK############################ #####
##########PQcancelPoll############### ################################
## ############################################ #################

CONNECTION_ALLOCATED

   ######PQcancelStart###PQcancelBlocking##########################
   ## ####PQcancelCreate###PQcancelReset############### ###############
   ########### #################PQcancelStart###PQcancelBlocking######
   ####

CONNECTION_STARTED

   ############

CONNECTION_MADE

   ###OK############

CONNECTION_AWAITING_RESPONSE

   ###############

CONNECTION_SSL_STARTUP

   SSL###########

CONNECTION_GSS_STARTUP

   #####[51.851852]#SSL########### ######GSS #############

##############################################################################
########################## ########################


switch(PQcancelStatus(conn))
{
        case CONNECTION_STARTED:
            feedback = "Connecting...";
            break;

        case CONNECTION_MADE:
            feedback = "Connected to server...";
            break;
.
.
.
        default:
```

```
                feedback = "Connecting...";
    }
```

#####[82.795699]#PQconnectPoll#######connect_timeout############## ### ############################## ######PQconnectStart###PQconnectPoll## ####PQconnectdb######## ######PQcancelPoll#######connect_timeout## ############ ############################ #######PQcancelStart##### #PQcancelPoll####PQcancelBlocking######

PQcancelStatus

#####[68.181818]########### ####################

```
ConnStatusType PQcancelStatus(const PGcancelConn *cancelConn);
```

############################## ##CONNECTION_ALLOCATED#CONNECTION_OK#CONNECTION_BAD#3###### PQcancelCreate############PGcancelConn#####CONNECTION_ALLOCATED# ## ##################CONNECTION_OK######## ############## #CONNECTION_BAD####### PQcancelFinish###PQcancelReset#########OK## ##########

##########################PQcancelStart############

#####[78.917379]## ############### ################################ ## ###################### ################################## ############### ########################## ########### ################################## ##########

PQcancelSocket

#####################################

```
int PQcancelSocket(const PGcancelConn *cancelConn);
```

################ 0 ##### -1 ############################## ########## #PGcancelConn# PQcancelErrorMessage #PQcancelSocket################# ###########

PQcancelErrorMessage

##################################

```
char *PQcancelErrorMessage(const PGcancelConn *cancelconn);
```

#####[77.664234]########libpq####### PQerrorMessage ############## libpq### ######### PQerrorMessage ## ##### ################## ####PGconn#####PQfinish################ PGconn###### ############################## ######PGcancelConn#####libpq############### PQcancelErrorMessage ############ libpq######### PQcancelErrorMessage ############################## ##################### ####PGcancelConn### ##PQcancelFinish############ #######PGcancelConn##################### ########

PQcancelFinish

#####[55.974843]############## ###PGconn################## ############### ########################### ###PGcancelConn####################

```
void PQcancelFinish(PGcancelConn *cancelConn);
```

#####[78.364116]##################PQstatus###############PQfinish#### #PGconn#################### ###PQfinish#########PGconn############## ## ##################PQcancelStatus################## #PQcancelFinish#####PGcancelConn########################## PGcancelConn#####PQcancelFinish####################

PQcancelReset

########################PGcancelConn########

```
void PQcancelReset(PGcancelConn *cancelConn);
```

######PGcancelConn#################################### ############### ############PGcancelConn############

#########PGconn####1##PGcancelConn######PGconn#####################

32.7.2. Obsolete Functions for Sending Cancel Requests

################################## ############################sslmode## #gssencmode## ######################## ####################################

PQgetCancel

#####[55.084746]## #### ##PQcancel######################################

```
PGcancel *PQgetCancel(PGconn *conn);
```

#####[89.511754]#PQgetCancel######PGconn########PGcancel############ # ####*conn*#NULL###############NULL###### PGcancel#################### #################### ###PQcancel####PQfreeCancel############ #### ##PQgetCancel#PGconn############PGcancel############ #### #*conn*#NULL###############NULL##### PGcancel########################## ######### ###PQcancel###PQfreeCancel############

PQfreeCancel

PQgetCancel################

```
void PQfreeCancel(PGcancel *cancel);
```

PQfreeCancel####PQgetCancel####################

PQcancel

######PQcancel##PQcancelBlocking##################################

```
int PQcancel(PGcancel *cancel, char *errbuf, int errbufsize);
```

######PQcancel################### ####PQcancelBlocking######### PQcancel#######*errbuf*## ##

######PGcancel######PQcancel################PGconn################## #################

#####[78.143713]##################1########0###### #######*errbuf*########### ############ *errbuf*#*errbufsize*################## (######256######) ####################PQcancel#####1### ###############*errbuf*############## ##### *errbuf*#*errbufsize*##########256##################

PQrequestCancel

#####[71.074380]#PQrequestCancel#PQcancel########### #### ##PQrequestCancel##PQcancelBlocking##############

```
int PQrequestCancel(PGconn *conn);
```

######PQrequestCancel############### ####PQcancelBlocking######### PQrequestCancel#PQcancelBlocking###############

###################### ###PGconn#############################PGconn###### ######## (PQerrorMessage ##############) ################################ ###############PGconn###

32.8.

PostgreSQL###################### (fast-path) ################

> ###
>
> ################### ## ## ##

PQfn##############################

```
PGresult *PQfn(PGconn *conn,
               int fnid,
               int *result_buf,
               int *result_len,
               int result_is_int,
               const PQArgBlock *args,
               int nargs);

typedef struct
{
    int len;
    int isint;
    union
    {
        int *ptr;
        int integer;
    } u;
} PQArgBlock;
```

fnid#########OID### *args*#*nargs*################# ########################### ### ########*isint*######*u.integer*###################### (###2####4########### ###) ################### *isint*######**u.ptr*################### ######### ### (*u.ptr*#int *################void *############) *result_buf*####################### ###########################

######### ##################### ###############result_len########### ###2#4 #############result_is_int#1########0###### result_is_int#1############## ##########libpq########### ################int########### 4################ #*result_buf################ result_is_int#0############################## ######## (#####result_buf#void *##############)

PQfn######PGresult*######### #######PGRES_COMMAND_OK######### #PGRES_FATAL_ERROR############ ############################## ############ ##PQclear#####PGresult#################

###NULL##################len######-1####### isint######u###########

#####NULL######*result_len#-1######*result_buf########

################

32.9.

PostgreSQL##LISTEN#NOTIFY###################### ############LISTEN####### ###################### ############UNLISTEN########### ################# #NOTIFY## ##################### ###############

libpq############SQL#############LISTEN#UNLISTEN###NOTIFY################ NOTIFY############PQnotifies.############

PQnotifies##################################### ############NULL######### PQnotifies####################################

```
PGnotify *PQnotifies(PGconn *conn);

typedef struct pgNotify
{

    char *relname;              /* ####### */
    int  be_pid;                /* ###############ID */
    char *extra;                /* ######### */
} PGnotify;
```

PQnotifies#####PGnotify###############PQfreemem################ PGnotify################ relname#extra################### (############### ####################################)

32.2##########################

PQnotifies######################### #######libpq######################### ## libpq#########NOTIFY## #PQexec#######PQnotifies############### ################################## #########

##############NOTIFY#################PQconsumeInput #########PQnotifies# ######## ############select()###############CPU################### #select()######################PQsocket########## ########## #PQsendQuery#PQgetResult###############PQexec############ ############## ##########PQgetResult####PQexec######PQnotifies####################

32.10. COPY########

PostgreSQL#COPY######libpq## ### ##

###################PQexec###########COPY SQL########## ################### ###################PGRES_COPY_OUT####PGRES_COPY_IN###########PGresult## #### ## ##################### ########PGresult############ ##########PGRES_COMMAND_OK################## PGRES_FATAL_ERROR###### ########SQL#####PQexec############## #COPY######## ##########SQL##################

COPY#####################PQexec###################COPY###### ##PQgetResult##################### PQexec################################ ######PQgetResult#NULL########

#######PQexec####PQgetResult##PGRES_COPY_OUT####PGRES_COPY_IN######### #################

############PGresult###########COPY################ ##################### #########################

PQnfields

#######(#####)#######

PQbinaryTuples

0###############(#######################)########## 1##################### ### ###COPY#########

PQfformat

#####################(######0#######1)###### ############################### ### ################################### (#######COPY##################### #### ###########################)

32.10.1. COPY########

########COPY FROM STDIN################### ###COPY_IN################### #####

PQputCopyData

COPY_IN##################

```
int PQputCopyData(PGconn *conn,
                  const char *buffer,
                  int nbytes);
```

####*buffer*###COPY####*nbytes*############ ###################1###### ##### ####################### ########################### ##########-1###### #### #-1############## PQerrorMessage ######### ############################## ####

#########COPY#################################### ######################### ####### ############COPY############################## ###COPY#########

PQputCopyEnd

COPY_IN#######################

```
int PQputCopyEnd(PGconn *conn,
                 const char *errormsg);
```

errormsg#NULL###COPY_IN########### *errormsg*#NULL######*errormsg*#### ##################COPY########### ################################### ###############COPY################

##################1###### ### ##### ######################################PQflush#########0############ ## ################################### ######################### ########### ##########PQputCopyEnd############# #############-1###### ############## PQerrorMessage #######

PQputCopyEnd###########PQgetResult######COPY###################### ## ############################## #################

32.10.2. COPY########

#######COPY TO STDOUT################## COPY_OUT##################

PQgetCopyData

COPY_OUT##################

```
int PQgetCopyData(PGconn *conn,
                  char **buffer,
                  int async);
```

COPY####################### ##1##1############ ###################### ### ## *buffer*#######NULL########## **buffer*##################################NULL####### #NULL############## #PQfreemem################

################################## (#####0#########) ##########NULL######## #####COPY############ ########COPY########################## (*async*#### ###############) -1######COPY############ -2##################### (####### # PQerrorMessage #########)

async##(#0)####PQgetCopyData################## COPY############### #PQgetCopyData#0###### (############################## PQconsumeInput ## ################) *async*##(0)####PQgetCopyData########################### ###########

PQgetCopyData#-1######PQgetResult#######COPY#################### ##### ########################## #################

32.10.3. ###COPY###

######COPY############### ## #######

PQgetline

##########################*length*################

```
int PQgetline(PGconn *conn,
              char *buffer,
```

```
                int length);
```

##########*length*-1#################1############# PQgetline#######
#EOF############0###### ##1######

###################\.###2#################### ##2####COPY###############
############ ############*length*-1##########################\.###########
######################################

PQgetlineAsync

COPY##

```
int PQgetlineAsync(PGconn *conn,
                   char *buffer,
                   int bufsize);
```

PQgetline#######COPY##
COPY###########PGRES_COPY_OUT##############################
#PQconsumeInput #PQgetlineAsync#######

PQgetline###########################

PQgetlineAsync##########libpq################################ #########
################ #########################-1##############0################
############ ##-1##############PQendcopy############## ##############

################################ ################# ########################
################1####\n########
####### #####COPY###################COPY#################### ###########
NULL######### #NULL##############################-1#*bufsize*############
####

PQputline

####NULL############ ######0###### ################EOF######

```
int PQputline(PGconn *conn,
              const char *string);
```

PQputline#############COPY#########PQgetlineAsync################# ##
###########PQputline#####1#################### ####################
######

##
PostgreSQL#####3.0#####################COPY####################### ####\.###2################### ########## ################\.########## ################### ############PQendcopy############

PQputnbytes

NULL################### ######0###### ################EOF######

```
int PQputnbytes(PGconn *conn,
                const char *buffer,
```

```
                int nbytes);
```

######PQputline###### ###################NULL############# ############## #################

PQendcopy

##########

```
int PQendcopy(PGconn *conn);
```

######################## ######PQputline######################## #PQgetline## ################### ################### ##################SQL############### #############0## ## ########### ########### PQerrorMessage ##################

PQgetResult##############PQgetline#########PGRES_COPY_OUT########## ######PQendcopy############## #####PQgetResult#NULL####### ##PQgetResult################ ##### PGRES_COPY_IN#####PQputline### ######PQendcopy########PQgetResult######### ##############SQL####### #COPY#######################

################COPY#PQexec#####PQendcopy############################### ### ###########SQL#COPY##################

32.11.

#######libpq##############

PQclientEncoding

################

```
int PQclientEncoding(const PGconn *conn);
```

###EUC_JP##################ID############### ############-1###### #####ID### ########################

```
char *pg_encoding_to_char(int encoding_id);
```

PQsetClientEncoding

##################

```
int PQsetClientEncoding(PGconn *conn, const char *encoding);
```

conn#########*encoding*############# ########################## #####-1#### ## ################PQclientEncoding################

PQsetErrorVerbosity

PQerrorMessage #PQresultErrorMessage###################

```
typedef enum
{
    PQERRORS_TERSE,
    PQERRORS_DEFAULT,
```

```
    PQERRORS_VERBOSE,
    PQERRORS_SQLSTATE
} PGVerbosity;

PGVerbosity PQsetErrorVerbosity(PGconn *conn, PGVerbosity verbosity);
```

PQsetErrorVerbosity########################## *TERSE*################## ################## ############## *DEFAULT*############################## ########################### *VERBOSE*########################## *SQLSTATE*## ####################SQLSTATE#############(###########*TERSE*##########)#

#############PGresult########################## ###################### ### ##########################PQresultVerboseErrorMessage##########

PQsetErrorContextVisibility

PQerrorMessage ###PQresultErrorMessage############CONTEXT########## #######

```
typedef enum
{
    PQSHOW_CONTEXT_NEVER,
    PQSHOW_CONTEXT_ERRORS,
    PQSHOW_CONTEXT_ALWAYS
} PGContextVisibility;

PGContextVisibility PQsetErrorContextVisibility(PGconn *conn,
 PGContextVisibility show_context);
```

PQsetErrorContextVisibility################################## ###### ######CONTEXT################### *NEVER*########CONTEXT#######*ALWAYS*# #CONTEXT################ *ERRORS*############CONTEXT################### ########## #############*TERSE#SQLSTATE*####################CONTEXT###### ########

###############PGresult## ########## ####################PQresultVerboseErrorMessage##########

PQtrace

##

```
void PQtrace(PGconn *conn, FILE *stream);
```

##F######################B## ################################## ## ######## ################### ## ## ########16################ ##################53.7#########

> **##**
>
> Windows#####libpq#######################################FILE######### ########################## ################################/######### ####/##########/##################################

PQsetTraceFlags

#########################

```
void PQsetTraceFlags(PGconn *conn, int flags);
```

flags################################
flags#PQTRACE_SUPPRESS_TIMESTAMPS#################################
flags#PQTRACE_REGRESS_MODE############################OID#########
########################### ######PQtrace###################

PQuntrace

PQtrace######################

```
void PQuntrace(PGconn *conn);
```

32.12.

################################

PQfreemem

libpq###############

```
void PQfreemem(void *ptr);
```

#####PQescapeByteaConn#PQescapeBytea#PQunescapeBytea###PQnotifies#
##libpq############### Microsoft Windows####free()####################
DLL############################DLL#################/##############/#
######/###################### Microsoft Windows#######################
#free()#######

PQconninfoFree

PQconndefaults####PQconninfoParse################

```
void PQconninfoFree(PQconninfoOption *connOptions);
```

###NULL##################

###PQfreemem####################################

PQencryptPasswordConn

PostgreSQL####################

```
char *PQencryptPasswordConn(PGconn *conn, const char *passwd, const
 char *user, const char *algorithm);
```

######ALTER USER joe PASSWORD 'pwd'##################################
########## ##
#######################################

passwd#user####################SQL######## *algorithm*################
################## ##################md5#scram-sha-256### (##########
#########md5######on#off#######) scram-sha-256######PostgreSQL#####10
algorithm#NULL################
#password_encryption####### ##

```
###############                    ####################################
##PQencryptPasswordConn######password_encryption#########
#algorithm########

####malloc############# ##################################################
##### ########PQfreemem################ #######NULL#####################
########
```

PQchangePassword

```
######PostgreSQL############


PGresult *PQchangePassword(PGconn *conn, const char *user, const char
 *passwd);

###########PQencryptPasswordConn#########ALTER USER .##### '...',.###
#################### ###PQencryptPasswordConn#########################
######## PQencryptPasswordConn#########NULL#########password_encryption#
###############

######user#passwd#########SQL##############

#####[58.859470]#####PGresult#############NULL######  ##################
##############################NULL##########  PQresultStatus###########
###NULL################PGRES_FATAL_ERROR########################## ##
###########  PQerrorMessage  ##########  ######ALTER    USER########
#PGresult##############################NULL######### #######NULL#######
#PGRES_FATAL_ERROR#############PQresultStatus############## ########
####### PQerrorMessage #########
```

PQencryptPassword

```
md5######PostgreSQL###########


char *PQencryptPassword(const char *passwd, const char *user);

PQencryptPassword##############PQencryptPasswordConn###          ##
##PQencryptPassword###################md5######################
```

PQmakeEmptyPGresult

```
#############PGresult############


PGresult *PQmakeEmptyPGresult(PGconn *conn, ExecStatusType status);

#####PGresult#################libpq#######  ######################NULL###
###  ######################################################################
########  conn##NUL##status################################PGresult##
#######  ####conn##NULL#######################PGresult########  ####
#PGEVT_RESULTCREATE###########PQfireResultCreateEvents######## libpq#
######PGresult#####################PQclear#######################
```

PQfireResultCreateEvents

```
PGresult##############################PGEVT_RESULTCREATE#####32.14#####
###### ##################################
```

```
int PQfireResultCreateEvents(PGconn *conn, PGresult *res);
```

conn################################ ##################NULL#######

###########PGEVT_RESULTCREATE####PGEVT_RESULTCOPY################## ##############

#####PQmakeEmptyPGresult##################################PGresult## ########################

PQcopyResult

PGresult############### ##################################PQclear###### ########## #######NULL#######

```
PGresult *PQcopyResult(const PGresult *src, int flags);
```

############################ ########PGRES_TUPLES_OK################## ################### ################### *flags*################## ###### #################OR### PG_COPYRES_ATTRS################## ##### PG_COPYRES_TUPLES################### ################### ### PG_COPYRES_NOTICEHOOKS################### PG_COPYRES_EVENTS# ################## ################################### ########## ##PGEVT_RESULTCOPY##########

PQsetResultAttrs

PGresult###############

```
int PQsetResultAttrs(PGresult *res, int numAttributes, PGresAttDesc
 *attDescs);
```

#####*attDescs*########### ##*attDescs*#####NULL####*numAttributes######## ############### *res*######################## ################ ########## ##########

PQsetvalue

PGresult######################

```
int PQsetvalue(PGresult *res, int tup_num, int field_num, char *value,
 int len);
```

############################### ####*tup_num*###PQntuples################ #### ################################## ################################ ######## ##*field_num*################## *len*#-1####*value*#NULL############ SQL#NULL####### *value*### ####### ########## ####################

PQresultAlloc

PGresult######################

```
void *PQresultAlloc(PGresult *res, size_t nBytes);
```

res############################## ###########NULL### malloc############## ############################

PQresultMemorySize

PGresult##########################

```
size_t PQresultMemorySize(const PGresult *res);
```

#####[88.888889]#####PGresult##########malloc##########PQclear######## #### ###################### ##########PGresult#############malloc##### #PQclear################# ######################

PQlibVersion

####libpq###########

```
int PQlibVersion(void);
```

##############################libpq################################ ########### ##PQconnectdb############################

###################10000####################### #########10.1##100001###### ###11.0##110000######

#####10#######PostgreSQL######2#################3######################## # ###########PQlibVersion########2########## ##########9.1.5##90105########## 9.2.0##90200#######

################################PQlibVersion####10000#####100############ ############# #############2################## #############

##
#####PostgreSQL#####9.1######## ##################################### ##### ###############9.1###################

PQgetCurrentTimeUSec

################ ####Unix##########################time_t×100#)#

```
pg_usec_time_t PQgetCurrentTimeUSec(void);
```

############PQsocketPoll#######################

32.13.

#############stderr################################

##

##PQsetNoticeReceiver################################ ## ##PQsetNoticeProcessor#######################

```
typedef void (*PQnoticeReceiver) (void *arg, const PGresult *res);
```

```
PQnoticeReceiver
PQsetNoticeReceiver(PGconn *conn,
                    PQnoticeReceiver proc,
                    void *arg);

typedef void (*PQnoticeProcessor) (void *arg, const char *message);

PQnoticeProcessor
PQsetNoticeProcessor(PGconn *conn,
                     PQnoticeProcessor proc,
                     void *arg);
```

#######NULL################
##############

#######/###################libpq#####/##############################
PGRES_NONFATAL_ERROR PGresult################ ###########
#PQresultErrorField#####################PQresultErrorMessage###
#PQresultVerboseErrorMessage################################
PQsetNoticeReceiver#####void############### ##############################
###############

###############PQresultErrorMessage################################

#######################/################### ############################
##PQsetNoticeProcessor########void########### ############################
##################

###########################

```
static void
defaultNoticeProcessor(void *arg, const char *message)
{
    fprintf(stderr, "%s", message);
}
```

######################PGconn################PGresult#####################
#################### PGresult######PGconn###############PQgetvalue######
###########PGresult########

32.14.

libpq###########PGconn###PGresult####################libpq##################
################## ##########################PGconn###PGresult##############
##################

###################libpq######void *########################### ######
###PGconn################################ #########PGconn################
##PGresult################### ############################### ##############
#############PGconn#PGresult#NULL####### ####
##PQinstanceData#PQsetInstanceData#PQresultInstanceData##
#PQresultSetInstanceData################# #############PGconn############
#######PGresult############## libpq######################################
############### ################

32.14.1.

PGEventId################################ #######PGEVT############## #######
################

PGEVT_REGISTER

#####[88.253012]#######PQregisterEventProc########### ################
#######instanceData################### ################################
#################################### ##############
##PQregisterEventProc############## ###############instanceData#####
########## ###############1################## ##################0#########
#########

```
typedef struct
{
    PGconn *conn;
} PGEventRegister;
```

PGEVT_REGISTER###########*evtInfo*#####PGEventRegister *############
######CONNECTION_OK###########PGconn####### ###########PGconn####
####PQregisterEventProc############ ##########PGEVT_CONNDESTROY####
#############################

PGEVT_CONNRESET

##########PQreset###PQresetPoll############ ##########################
PostgreSQLv15######################### ################################ ##
##############

```
typedef struct
{
    PGconn *conn;
} PGEventConnReset;
```

PGEVT_CONNRESET###########*evtInfo*#####PGEventConnReset *######
########## ####PGconn################################ ##############
#instanceData############################## #########
#PGEVT_CONNRESET###################PGEVT_CONNDESTROY###############
##########

PGEVT_CONNDESTROY

#########PQfinish########### libpq#######################################
############## #################

```
typedef struct
{
    PGconn *conn;
} PGEventConnDestroy;
```

PGEVT_CONNDESTROY###########*evtInfo*#####PGEventConnDestroy *######
########## #######PQfinish################### ###############PQfinish##
################### ######################################

PGEVT_RESULTCREATE

##########PQgetResult################################ #################
###########

```
typedef struct
{
```

```
    PGconn *conn;
    PGresult *result;
} PGEventResultCreate;
```

PGEVT_RESULTCREATE############*evtInfo*#####PGEventResultCreate *### ############# *conn*################## ####################instanceData# ################## ## ## ##########################PGEVT_RESULTCOPY###PGEVT_RESULTDESTROY# ###########

PGEVT_RESULTCOPY

##########PQcopyResult############ ######################### ###### #PGEVT_RESULTCREATE####PGEVT_RESULTCOPY#################### ##PGEVT_RESULTCOPY###########

```
typedef struct
{
    const PGresult *src;
    PGresult *dest;
} PGEventResultCopy;
```

PGEVT_RESULTCOPY############*evtInfo*#####PGEventResultCopy *###### ########## *src*#########*dest*######### ######instanceData############ ######### PQcopyResult################## ############################## ############################## ######################PGEVT_RESULTCOPY## #PGEVT_RESULTDESTROY############

PGEVT_RESULTDESTROY

########PQclear########### libpq## ############# ################

```
typedef struct
{
    PGresult *result;
} PGEventResultDestroy;
```

PGEVT_RESULTDESTROY########*evtInfo*#####PGEventResultDestroy *# ############### ######PQclear######################### ############## ##PQclear##################### ####################################

32.14.2.

PGEventProc

PGEventProc####################typedef### ####libpq######################## ############################

```
int eventproc(PGEventId evtId, void *evtInfo, void *passThrough)
```

evtId#######PGEVT############### *evtInfo*############################## ################ *passThrough*#################### ##PQregisterEventProc########### ################################

################PGconn############## ################################## ##############

##

Windows########################## #######DLL#######DLL### libpq### ### ############# ######### #######################static################### ################## ###

32.14.3.

PQregisterEventProc

libpq#######################

```
int PQregisterEventProc(PGconn *conn, PGEventProc proc,
                        const char *name, void *passThrough);
```

################PGconn########################## ######################### ################ ########################

libpq###########*proc*######## #########instanceData############## *name*# ################################### ####NULL################## ##name# ###PGconn################################ *passThrough*################## #*proc*####### #####NULL##########

PQsetInstanceData

proc#########*conn*###instanceData#*data*###### ##################### #*conn*#*proc*############################

```
int PQsetInstanceData(PGconn *conn, PGEventProc proc, void *data);
```

PQinstanceData

proc##########*conn*###instanceData##########NULL######

```
void *PQinstanceData(const PGconn *conn, PGEventProc proc);
```

PQresultSetInstanceData

proc######instanceData#*data*###### ##################### ####*proc*###### #####################

```
int PQresultSetInstanceData(PGresult *res, PGEventProc proc, void
 *data);
```

data########PQresultAlloc###############PQresultMemorySize######### ######### (###############################PQresultAlloc############## ##)

PQresultInstanceData

proc#######instanceData##########NULL######

```
void *PQresultInstanceData(const PGresult *res, PGEventProc proc);
```

32.14.4.

###libpq#################################

```
/* libpq###############libpq-fe.h######## */
#include <libpq-events.h>


/* instanceData */
typedef struct
{
    int n;
    char *str;
} mydata;

/* PGEventProc */
static int myEventProc(PGEventId evtId, void *evtInfo, void *passThrough);

int
main(void)
{
    mydata *data;
    PGresult *res;
    PGconn *conn =
        PQconnectdb("dbname=postgres options=-csearch_path=");

    if (PQstatus(conn) != CONNECTION_OK)
    {
        /* PQerrorMessage's result includes a trailing newline */
        fprintf(stderr, "%s", PQerrorMessage(conn));
        PQfinish(conn);
        return 1;
    }


    /* ########################
     * myEventProc#PGEVT_REGISTER###
     */
    if (!PQregisterEventProc(conn, myEventProc, "mydata_proc", NULL))
    {
        fprintf(stderr, "Cannot register PGEventProc\n");
        PQfinish(conn);
        return 1;
    }


    /* conn instanceData### */
    data = PQinstanceData(conn, myEventProc);


    /* myEventProc#PGEVT_RESULTCREATE### */
    res = PQexec(conn, "SELECT 1 + 1");


    /* ## instanceData### */
    data = PQresultInstanceData(res, myEventProc);


    /* PG_COPYRES_EVENTS########PGEVT_RESULTCOPY#myEventProc### */
    res_copy = PQcopyResult(res, PG_COPYRES_TUPLES | PG_COPYRES_EVENTS);
```

```
    /* PQcopyResult########PG_COPYRES_EVENTS#########
     * ## instanceData###
     */
    data = PQresultInstanceData(res_copy, myEventProc);


    /* ###clear#PGEVT_RESULTDESTROY#myEventProc### */
    PQclear(res);
    PQclear(res_copy);


    /* PGEVT_CONNDESTROY#myEventProc### */
    PQfinish(conn);

    return 0;
}

static int
myEventProc(PGEventId evtId, void *evtInfo, void *passThrough)
{
    switch (evtId)
    {
        case PGEVT_REGISTER:
        {
            PGEventRegister *e = (PGEventRegister *)evtInfo;
            mydata *data = get_mydata(e->conn);


            /* ################### */
            PQsetInstanceData(e->conn, myEventProc, data);
            break;
        }

        case PGEVT_CONNRESET:
        {
            PGEventConnReset *e = (PGEventConnReset *)evtInfo;
            mydata *data = PQinstanceData(e->conn, myEventProc);

            if (data)
              memset(data, 0, sizeof(mydata));
            break;
        }

        case PGEVT_CONNDESTROY:
        {
            PGEventConnDestroy *e = (PGEventConnDestroy *)evtInfo;
            mydata *data = PQinstanceData(e->conn, myEventProc);


            /* conn#################### */
            if (data)
              free_mydata(data);
            break;
        }

        case PGEVT_RESULTCREATE:
        {
            PGEventResultCreate *e = (PGEventResultCreate *)evtInfo;
            mydata *conn_data = PQinstanceData(e->conn, myEventProc);
            mydata *res_data = dup_mydata(conn_data);


            /* ##############conn############ */
```

```
            PQresultSetInstanceData(e->result, myEventProc, res_data);
            break;
        }

        case PGEVT_RESULTCOPY:
        {
            PGEventResultCopy *e = (PGEventResultCopy *)evtInfo;
            mydata *src_data = PQresultInstanceData(e->src, myEventProc);
            mydata *dest_data = dup_mydata(src_data);


            /* ############################ */
            PQresultSetInstanceData(e->dest, myEventProc, dest_data);
            break;
        }

        case PGEVT_RESULTDESTROY:
        {
            PGEventResultDestroy *e = (PGEventResultDestroy *)evtInfo;
            mydata *data = PQresultInstanceData(e->result, myEventProc);


            /* ######################## */
            if (data)
              free_mydata(data);
            break;
        }


        /* #############true### */
        default:
            break;
    }


    return true; /* ######## */
}
```

32.15.

##PQconnectdb#PQsetdbLogin###PQsetdb########

- PGHOST#host#################
- #####[79.032258]#PGGSSDELEGATION#gssdelegation################# #### ##PGSSLNEGOTIATION#sslnegotiation#################
- PGHOSTADDR#hostaddr################# PGHOST##############PGHOST######DNS### ######################
- PGPORT#port#################
- PGDATABASE#dbname#################
- PGUSER#user#################
- PGPASSWORD#password################# ##########################root#######ps### ## #############(32.16#########)############
- PGPASSFILE#passfile#################

- `PGREQUIREAUTH`#require_auth################
- `PGCHANNELBINDING`#channel_binding################
- `PGSERVICE`#service################
- `PGSERVICEFILE`#####################(32.17##)# ##### #`~/.pg_service.conf`#Microsoft Windows##`%APPDATA%\postgresql\.pg_service.conf`###
- `PGOPTIONS`#options################
- `PGAPPNAME`#application_name################
- `PGSSLMODE`#sslmode################
- `PGREQUIRESSL`#requiressl################ #######`PGSSLMODE`################### ###########`PGREQUIRESSL`##########
- `PGSSLCOMPRESSION`#sslcompression################
- `PGSSLCERT`#sslcert################
- `PGSSLKEY`#sslkey################
- `PGSSLCERTMODE`#sslcertmode################
- `PGSSLROOTCERT`#sslrootcert################
- `PGSSLCRL`#sslcrl################
- `PGSSLCRLDIR`##sslcrldir################
- `PGSSLSNI`#sslsni################
- `PGREQUIREPEER`#requirepeer################
- `PGSSLMINPROTOCOLVERSION`#ssl_min_protocol_version################
- `PGSSLMAXPROTOCOLVERSION`#ssl_max_protocol_version################
- `PGGSSENCMODE`#gssencmode################
- `PGKRBSRVNAME`#krbsrvname################
- `PGGSSLIB`#gsslib################
- `PGGSSDELEGATION`#gssdelegation################
- `PGCONNECT_TIMEOUT`#connect_timeout################
- `PGCLIENTENCODING`#client_encoding################
- `PGTARGETSESSIONATTRS`#target_session_attrs################
- `PGLOADBALANCEHOSTS`#load_balance_hosts################

#############PostgreSQL########################## (############################## #############ALTER ROLE###ALTER DATABASE##############)

- `PGDATESTYLE`########/############ (`SET datestyle TO ...`######)
- `PGTZ`################ (`SET timezone TO ...`######)
- `PGGEQO`################################ (`SET geqo TO ...`######)

##################SET SQL#############

#########libpq########### ######################

- PGSYSCONFDIR#pg_service.conf################## ######################
  ##############
- PGLOCALEDIR#################locale##################

32.16.

##############.pgpass###############(######################)################
Microsoft Windows############%APPDATA%\postgresql\pgpass.conf(##
#%APPDATA%############################)### ##########passfile##########
#PGPASSFILE##################

###################

hostname:*port*:*database*:*username*:*password*

###4################
###############*############ ##################################### (#######
##################################) ####:###\##########\###################
##########host##################hostaddr###########
#localhost####### ###Unix###########host#####libpq######################
#####localhost####### ##########replication############################
######################### ##
#######

Unix## ###chmod 0600
~/.pgpass############### ########################### Microsoft Windows#######
###

32.17.

##############libpq############################## #######libpq################
############## ###########libpq################################### #####
#PGSERVICE##################

############################
################ ######################~/.pg_service.conf##### Microsoft
Windows###%APPDATA%\postgresql\.pg_service.conf#######%APPDATA%#####
#######Application Data########### #######PGSERVICEFILE############### #######
#####pg_service.conf####### #######PostgreSQL######etc##################
###########pg_config--sysconfdir####### ####PGSYSCONFDIR###############
##########################

##############INI############## ############################## 32.1.2###########
#########

```
# comment
[mydb]
host=somehost
port=5433
user=admin
```

#########PostgreSQL######share/pg_service.conf.sample######

########################
################################## ###########################service=mydb
port=5434##hostsomehost#port5434#useradmin################################
#######

32.18. ########LDAP##

libpq#LDAP#####configure##--with-ldap##########################LDAP### #host#dbname###################### ### #################

LDAP##########pg_service.conf#############32.17########## pg_service.conf##ldap://#######LDAP URL#######LDAP################## ## ####keyword = value###################### ###################### ##URL#RFC 1959[4]#########################

```
ldap://[hostname[:port]]/search_base?attribute?search_scope?filter
```

####*hostname*######localhost#*port*######389###

pg_service.conf####LDAP############## ######LDAP######################## # ################LDAP#########LDAP######keyword = value################## ################ ######################LDAP URL######################

LDIF###########LDAP############

```
version:1
dn:cn=mydatabase,dc=mycompany,dc=com
changetype:add
objectclass:top
objectclass:device
cn:mydatabase
description:host=dbserver.mycompany.com
description:port=5439
description:dbname=mydb
description:user=mydb_user
description:sslmode=require
```

##########LDAP URL#######

```
ldap://ldap.mycompany.com/dc=mycompany,dc=com?description?one?
(cn=mydatabase)
```

###LDAP############################## pg_service.conf##################

```
# only host and port are stored in LDAP, specify dbname and user
 explicitly
[customerdb]
dbname=customer
user=appuser
ldap://ldap.acme.com/cn=dbserver,cn=hosts?pgconnectinfo?base?
(objectclass=*)
```

32.19. SSL####

PostgreSQL##TLS##SSL################# #####SSL#########18.9#########

libpq##########OpenSSL############## ##############openssl.cnf##openssl version -d###################### #######OPENSSL_CONF######################## ###########

[4] https://datatracker.ietf.org/doc/html/rfc1959

32.19.1.

#######PostgreSQL#################### ########DNS###################IP#######
###########################(chain of trust)######################### #####################CA#######################
##################################
################

#####
#########
##

####
#sslmode#verify-ca##########libpq##
######## sslmode#verify-full#########libpq##############################
SSL####################### #####################verify-full#######

verify-full#####################(Subject Alternative Name(SAN))#######
#dNSName################################ ################*################
###########.################ ############################ ############IP####
#############iPAddress###dNSName#SAN####IP##########(DNS########
#)#iPAddressSAN##########dNSNameSAN###########IP#######################

> ##
>
> PostgreSQL#######################IP#####RFC 6125[5]############## ###IP#####
> #DNsNameSAN###IpAddressSAN##############SAN########Common Name##
> ##########

#################1#########################~/.postgresql/root.crt########
####### #Microsoft Windows#############%APPDATA%\postgresql\root.crt#
##
######

~/.postgresql/root.crl#####Microsoft Windows##%APPDATA%\postgresql
\root.crl#################CRL#############

##########CRL############sslrootcert#sslcrl#######
#PGSSLROOTCERT#PGSSLCRL############ sslcrldir######PGSSLCRLDIR####
##CRL#########################

> ##
>
> #########PostgreSQL##############CA###########sslmode=require###
> #verify-ca############# ##########CA########## ################## ####
> ####################verify-ca###verify-full########

32.19.2.

##libpq###############
#~/.postgresql/postgresql.crt################### ######################
########## ####~/.postgresql/postgresql.key#################### ####
############################## Microsoft Windows################%APPDATA%

[5] https://datatracker.ietf.org/doc/html/rfc6125

\postgresql\postgresql.crt#%APPDATA%\postgresql\postgresql.key### ### ############sslcert###sslkey##########PGSSLCERT###PGSSLKEY############

Unix## ###chmod 0600~/.postgresql/postgresql.key################ ###########root######### ##############0640################# ## ################ libpq## #Microsoft Windows######%APPDATA%\postgresql#####################################

postgresql.crt############################## ############################ #### ############################ ######################ssl_ca_file#############

#######PEM###ASN.1 DER#####

################OpenSSL######AES-128################################## ####### #################sslpassword################# #################sslpassword#### #####TTY##############OpenSSL###Enter PEM pass phrase:################## ## ########################sslpassword## ### PQsetSSLKeyPassHook_OpenSSL#########

##############18.9.5########

32.19.3.

sslmode################################ SSL############################

##

> ## ####### SSL###################

######MITM#

> ## ########### ## ############DNS## ## ################### ############ SSL#####################################

#####

> ## ################ ######### SSL###

SSL###############SSL## ############## ## libpq### ##sslmode#####verify-full###verify-ca#################################### ################## ########web######https URL############

##################################

####SSL################################## ################################ # 32.1####sslmode##################################

#32.1 SSL######

sslmode	####	MITM##	##
disable	###	###	############################## ##
allow	###	###	############################## ###################

sslmode	####	MITM##	##
prefer	###	###	## ####
require	##	###	## #####
verify-ca	##	CA######	## ##############
verify-full	##	##	## #######

verify-ca#verify-full######CA######### ###CA########verify-ca###CA#### ######################### #####verify-full################ ##CA############## ############verify-ca#############

sslmode#######prefer### ### ### ##

32.19.4. SSL###############

32.2##########SSL#################

#32.2 libpq/##########SSL#########

####	##	##
~/.postgresql/postgresql.crt	########	############
~/.postgresql/postgresql.key	##########	##
~/.postgresql/root.crt	#######	## #####
~/.postgresql/root.crl	#############	###

32.19.5. SSL#########

############libssl#libcrypto#######################libpq#SSL############ ##libssl#libcrypto#######################################libpq##### #PQinitOpenSSL############## ######libpq#################### ####OpenSSL#### #1.1.0#####################################

PQinitOpenSSL

```
##########################################

void PQinitOpenSSL(int do_ssl, int do_crypto);
```

do_ssl######libpq###################OpenSSL############ *do_crypto*##### ##libcrypto############# ########PQinitOpenSSL######################## ## SSL#####################################

#############OpenSSL#########libcrypto## ####

PQinitSSL

#######################################

```
void PQinitSSL(int do_ssl);
```

#####PQinitOpenSSL(do_ssl, do_ssl)###### OpenSSL###libcrypto#######
####################################

PostgreSQL 8.0###PQinitSSL#########PQinitOpenSSL#PostgreSQL 8.4######## ##
#########libpq##################PQinitSSL##############

32.20.

#####[75.124378]########1#######################PGconn#################### ####
(############################)
#########17###libpq################## ####1#######2#########PGconn##############
######### ##
############

PGresult## ###32.12#32.14####
#PGresult#########################PGresult#######################

################libpq###
#####libpq#########################

PQisthreadsafe

libpq####################

```
int PQisthreadsafe();
```

#####[59.523810]#libpq##########1#######0###### ######libpq#############1#####
#####0###### #####17######1######

#####[92.582418]########PQrequestCancel#PQoidStatus#######################
################# PQrequestCancel#PQcancel##PQoidStatus#PQoidValue#####
#########

(libpq#######)##########Kerberos##########Kerberos##################Kerberos#######
################ libpq##########################libpq#######
#PQregisterThreadLock###########

32.21. libpq########

libpq############(############)#########################

- libpq-fe.h#################

```
#include <libpq-fe.h>
```

#####################################

```
foo.c: In function `main':
foo.c:34: `PGconn' undeclared (first use in this function)
foo.c:35: `PGresult' undeclared (first use in this function)
foo.c:54: `CONNECTION_BAD' undeclared (first use in this function)
foo.c:68: `PGRES_COMMAND_OK' undeclared (first use in this function)
```

```
foo.c:95: `PGRES_TUPLES_OK' undeclared (first use in this function)
```

- ######-I*directory*##################PostgreSQL########################### ############################## ####################### ##################### ###

```
cc -c -I/usr/local/pgsql/include testprog.c
```

Makefile############CPPFLAGS###################

```
CPPFLAGS += -I/usr/local/pgsql/include
```

```
############################################################          #####
#pg_config##############################################
```

```
$ pg_config --includedir
/usr/local/include
```

```
####pkg-config############################
```

```
$ pkg-config --cflags libpq
-I/usr/local/include
```

```
############-I##################
```

```
##############################################
```

```
testlibpq.c:8:22: libpq-fe.h: No such file or directory
```

- ###############-lpq##########libpq############### ###-L*directory*########## ######libpq################# ##################################### ######### ####-lpq########-L########## ##########

```
cc -o testprog testprog1.o testprog2.o -L/usr/local/pgsql/lib -lpq
```

```
###pg_config##############################
```

```
$ pg_config --libdir
/usr/local/pgsql/lib
```

```
#############pkg-config######
```

```
$ pkg-config --libs libpq
-L/usr/local/pgsql/lib -lpq
```

```
#####################################
```

```
####################################
```

```
testlibpq.o: In function `main':
testlibpq.o(.text+0x60): undefined reference to `PQsetdbLogin'
testlibpq.o(.text+0x71): undefined reference to `PQstatus'
testlibpq.o(.text+0xa4): undefined reference to `PQerrorMessage'
```

```
###-lpq##########
```

```
/usr/bin/ld: cannot find -lpq
```

```
###-L###########################
```

32.22.

```
#########################src/test/examples###########
```

#32.1 libpq ######### 1

```
/*
 * src/test/examples/testlibpq.c
 *
 *
 * testlibpq.c
 *

 *      C##PostgreSQL############libpq####
 */
#include <stdio.h>
#include <stdlib.h>
#include "libpq-fe.h"

static void
exit_nicely(PGconn *conn)
{
    PQfinish(conn);
    exit(1);
}

int
main(int argc, char **argv)
{
    const char *conninfo;
    PGconn     *conn;
    PGresult   *res;
    int         nFields;
    int         i,
                j;

    /*

     * #########################conninfo###########
     * ################dbname=postgres######
     * ################################
     */
    if (argc > 1)
        conninfo = argv[1];
    else
        conninfo = "dbname = postgres";


    /* ############### */
    conn = PQconnectdb(conninfo);


    /* ######################## */
    if (PQstatus(conn) != CONNECTION_OK)
    {
        fprintf(stderr, "%s", PQerrorMessage(conn));
        exit_nicely(conn);
```

```
    }


    /* ################################# */
    res = PQexec(conn,
                 "SELECT pg_catalog.set_config('search_path', '',
false)");
    if (PQresultStatus(res) != PGRES_TUPLES_OK)
    {
        fprintf(stderr, "SET failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }

    /*

     * ############################PGresult#
     * PQclear###
     */
    PQclear(res);

    /*

     * ###################
     * ##############################
     * #######"select * from pg_database"###PQexec()#####
     * #################
     */


    /* ################## */
    res = PQexec(conn, "BEGIN");
    if (PQresultStatus(res) != PGRES_COMMAND_OK)
    {
        fprintf(stderr, "BEGIN command failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }
    PQclear(res);

    /*

     * ###############pg_database#########
     */
    res = PQexec(conn, "DECLARE myportal CURSOR FOR select * from
pg_database");
    if (PQresultStatus(res) != PGRES_COMMAND_OK)
    {
        fprintf(stderr, "DECLARE CURSOR failed: %s",
PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }
    PQclear(res);

    res = PQexec(conn, "FETCH ALL in myportal");
    if (PQresultStatus(res) != PGRES_TUPLES_OK)
    {
        fprintf(stderr, "FETCH ALL failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }
```

```
    /* ########### */
    nFields = PQnfields(res);
    for (i = 0; i < nFields; i++)
        printf("%-15s", PQfname(res, i));
    printf("\n\n");


    /* ########## */
    for (i = 0; i < PQntuples(res); i++)
    {
        for (j = 0; j < nFields; j++)
            printf("%-15s", PQgetvalue(res, i, j));
        printf("\n");
    }

    PQclear(res);


    /* ##########################… */
    res = PQexec(conn, "CLOSE myportal");
    PQclear(res);


    /* ############# */
    res = PQexec(conn, "END");
    PQclear(res);


    /* ###################### */
    PQfinish(conn);

    return 0;
}
```

#32.2 libpq ######### 2

```
/*
 * src/test/examples/testlibpq2.c
 *
 *
 * testlibpq2.c

 *      ###############
 *

 * ####################psql#################
 *   NOTIFY TBL2;
 * 4####################
 *

 * #######################
 * #######(src/test/examples/testlibpq2.sql###)##############
 *
 *   CREATE SCHEMA TESTLIBPQ2;
 *   SET search_path = TESTLIBPQ2;
 *   CREATE TABLE TBL1 (i int4);
 *   CREATE TABLE TBL2 (i int4);
 *   CREATE RULE r1 AS ON INSERT TO TBL1 DO
 *     (INSERT INTO TBL2 VALUES (new.i); NOTIFY TBL2);
 *
```

```
 * ############psql#####4#######
 *
 *   INSERT INTO TESTLIBPQ2.TBL1 VALUES (10);
 */

#ifdef WIN32
#include <windows.h>
#endif
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <errno.h>
#include <sys/select.h>
#include <sys/time.h>
#include <sys/types.h>

#include "libpq-fe.h"

static void
exit_nicely(PGconn *conn)
{
    PQfinish(conn);
    exit(1);
}

int
main(int argc, char **argv)
{
    const char *conninfo;
    PGconn     *conn;
    PGresult   *res;
    PGnotify   *notify;
    int         nnotifies;

    /*

     * ##########################conninfo###########
     * ###############dbname=postgres######
     * ################################
     */
    if (argc > 1)
        conninfo = argv[1];
    else
        conninfo = "dbname = postgres";


    /* ################ */
    conn = PQconnectdb(conninfo);


    /* ######################## */
    if (PQstatus(conn) != CONNECTION_OK)
    {
        fprintf(stderr, "%s", PQerrorMessage(conn));
        exit_nicely(conn);
    }


    /* ################################# */
    res = PQexec(conn,
                 "SELECT pg_catalog.set_config('search_path', '',
 false)");
    if (PQresultStatus(res) != PGRES_TUPLES_OK)
    {
```

```
        fprintf(stderr, "SET failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }

    /*

     * ############################PGresult#
     * PQclear###
     */
    PQclear(res);

    /*

     * LISTEN##############NOTIFY############
     */
    res = PQexec(conn, "LISTEN TBL2");
    if (PQresultStatus(res) != PGRES_COMMAND_OK)
    {
        fprintf(stderr, "LISTEN command failed: %s",
PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }
    PQclear(res);


    /* 4############# */
    nnotifies = 0;
    while (nnotifies < 4)
    {
        /*

         * ##############################
         * select(2)######poll()###############
         * ####
         */
        int         sock;
        fd_set      input_mask;

        sock = PQsocket(conn);

        if (sock < 0)

            break;              /* ########## */

        FD_ZERO(&input_mask);
        FD_SET(sock, &input_mask);

        if (select(sock + 1, &input_mask, NULL, NULL, NULL) < 0)
        {
            fprintf(stderr, "select() failed: %s\n", strerror(errno));
            exit_nicely(conn);
        }


        /* ########### */
        PQconsumeInput(conn);
        while ((notify = PQnotifies(conn)) != NULL)
        {
            fprintf(stderr,
                    "ASYNC NOTIFY of '%s' received from backend PID %d\n",
                    notify->relname, notify->be_pid);
            PQfreemem(notify);
```

```
            nnotifies++;
            PQconsumeInput(conn);
        }
    }

    fprintf(stderr, "Done.\n");


    /* ##################### */
    PQfinish(conn);

    return 0;
}
```

#32.3 libpq ######### 3

```
/*
 * src/test/examples/testlibpq3.c
 *
 *
 * testlibpq3.c

 *      ##############I/O####
 *

 * ############(src/test/examples/testlibpq3.sql###)#####
 * ################
 *
 * CREATE SCHEMA testlibpq3;
 * SET search_path = testlibpq3;
 * SET standard_conforming_strings = ON;
 * CREATE TABLE test1 (i int4, t text, b bytea);
 * INSERT INTO test1 values (1, 'joe''s place', '\000\001\002\003\004');
 * INSERT INTO test1 values (2, 'ho there', '\004\003\002\001\000');
 *

 * #############
 *
 * tuple 0: got
 *  i = (4 bytes) 1
 *  t = (11 bytes) 'joe's place'
 *  b = (5 bytes) \000\001\002\003\004
 *
 * tuple 0: got
 *  i = (4 bytes) 2
 *  t = (8 bytes) 'ho there'
 *  b = (5 bytes) \004\003\002\001\000
 */

#ifdef WIN32
#include <windows.h>
#endif

#include <stdio.h>
#include <stdlib.h>
#include <stdint.h>
#include <string.h>
#include <sys/types.h>
#include "libpq-fe.h"
```

```
/* ntohl/htonl# */
#include <netinet/in.h>
#include <arpa/inet.h>


static void
exit_nicely(PGconn *conn)
{
    PQfinish(conn);
    exit(1);
}

/*

 * ######################################
 * ############
 * main() ###2###############
 */
static void
show_binary_results(PGresult *res)
{
    int         i,
                j;
    int         i_fnum,
                t_fnum,
                b_fnum;


    /* ################# PQfnumber ##### */
    /* PQfnumber  */
    i_fnum = PQfnumber(res, "i");
    t_fnum = PQfnumber(res, "t");
    b_fnum = PQfnumber(res, "b");

    for (i = 0; i < PQntuples(res); i++)
    {
        char       *iptr;
        char       *tptr;
        char       *bptr;
        int         blen;
        int         ival;


        /* ######(NULL########) */
        iptr = PQgetvalue(res, i, i_fnum);
        tptr = PQgetvalue(res, i, t_fnum);
        bptr = PQgetvalue(res, i, b_fnum);

        /*

         * INT4#########################
         * #########################
         */
        ival = ntohl(*((uint32_t *) iptr));

        /*

         * TEXT#################
         * ##libpq###################
         * C######################
         *

         * BYTEA ##################
         * null ##############################
```

```
         */
        blen = PQgetlength(res, i, b_fnum);

        printf("tuple %d: got\n", i);
        printf(" i = (%d bytes) %d\n",
               PQgetlength(res, i, i_fnum), ival);
        printf(" t = (%d bytes) '%s'\n",
               PQgetlength(res, i, t_fnum), tptr);
        printf(" b = (%d bytes) ", blen);
        for (j = 0; j < blen; j++)
            printf("\\%03o", bptr[j]);
        printf("\n\n");
    }
}

int
main(int argc, char **argv)
{
    const char *conninfo;
    PGconn     *conn;
    PGresult   *res;
    const char *paramValues[1];
    int         paramLengths[1];
    int         paramFormats[1];
    uint32_t    binaryIntVal;

    /*

     * ##########################conninfo###########
     * ###############dbname=postgres######
     * ################################
     */
    if (argc > 1)
        conninfo = argv[1];
    else
        conninfo = "dbname = postgres";


    /* ############### */
    conn = PQconnectdb(conninfo);


    /* ######################## */
    if (PQstatus(conn) != CONNECTION_OK)
    {
        fprintf(stderr, "%s", PQerrorMessage(conn));
        exit_nicely(conn);
    }


    /* ################################ */
    res = PQexec(conn, "SET search_path = testlibpq3");
    if (PQresultStatus(res) != PGRES_COMMAND_OK)
    {
        fprintf(stderr, "SET failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }
    PQclear(res);

    /*

     * #########################PQexecParams()######
     * ########################
```

```
 *

 * #######################
 * #####################
 * ################################
 * ################################
 * ####################################
 * ####
 */


/* ################ */
paramValues[0] = "joe's place";

res = PQexecParams(conn,
                   "SELECT * FROM test1 WHERE t = $1",

                   1,           /* ######1## */
                   NULL,        /* ##################### */
                   paramValues,
                   NULL,        /* ################## */
                   NULL,        /* ##################### */
                   1);          /* ########## */

if (PQresultStatus(res) != PGRES_TUPLES_OK)
{
    fprintf(stderr, "SELECT failed: %s", PQerrorMessage(conn));
    PQclear(res);
    exit_nicely(conn);
}

show_binary_results(res);

PQclear(res);

/*

 * 2#############################
 * #######################
 *

 * ######################### PQexecParams ######
 * ############################## #########
 * ##############################
 */


/* ### "2" ################## */
binaryIntVal = htonl((uint32_t) 2);


/* PQexecParams ############### */
paramValues[0] = (char *) &binaryIntVal;
paramLengths[0] = sizeof(binaryIntVal);

paramFormats[0] = 1;        /* #### */

res = PQexecParams(conn,
                   "SELECT * FROM test1 WHERE i = $1::int4",

                   1,       /* ######1# */
                   NULL,    /* ##################### */
                   paramValues,
                   paramLengths,
```

```
                        paramFormats,

                        1);      /* ########## */

    if (PQresultStatus(res) != PGRES_TUPLES_OK)
    {
        fprintf(stderr, "SELECT failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }

    show_binary_results(res);

    PQclear(res);


    /* ###################### */
    PQfinish(conn);

    return 0;
}
```

#33#

PostgreSQL################## ## ##

#####PostgreSQL## libpq C####################PostgreSQL################################ ## ###########

33.1.

#############pg_largeobject######################## ############ #pg_largeobject_metadata################## ##########################/#### API#######################

PostgreSQL###TOAST############ ####### ########################## ################1########4############### #### TOAST#######1############### ##################################### ## TOAST###

33.2.

B-tree################

######################### ############################1000000################## ###########1000000########################## ############################## # ## ###Unix######################### ###########

PostgreSQL 9.0############################## ###GRANT###REVOKE############ ################SELECT######## ################UPDATE######## #############(# ###############)################################## ############################ #lo_compat_privileges#################

33.3.

#####PostgreSQL#libpq## PostgreSQL#################Unix######################## ##open#read#write#lseek##################

###SQL######## ################## INV_WRITE#####lo_open################################

##0###-1###### ######################### ###### PQerrorMessage ###############

##########################libpq/libpq-fs.h################libpq############## ######

libpq##

33.3.1.

#####

```
Oid lo_create(PGconn *conn, Oid lobjId);
```

################## #######OID#*lobjId*############ ########OID############### ############### *lobjId*#InvalidOid#######lo_create#####OID####### ####### ##############OID######InvalidOid##########

#########

```
inv_oid = lo_create(conn, desired_oid);
```

####

```
Oid lo_creat(PGconn *conn, int mode);
```

#########################OID####### ######################OID### ###### #InvalidOid(##)#######

PostgreSQL####8.1#####*mode*####### ####lo_creat#2######0###lo_create###### ### ####8.1##################lo_creat################ ############### ##lo_create####lo_creat############# *mode*#INV_READ#INV_WRITE#INV_READ|INV_WRITE################ ########## ##libpq/libpq-fs.h################

#######

```
inv_oid = lo_creat(conn, INV_READ|INV_WRITE);
```

33.3.2.

##

```
Oid lo_import(PGconn *conn, const char *filename);
```

filename## ################## #####OID### ####InvalidOid#0####### ## ########## ##

#####

```
Oid lo_import_with_oid(PGconn *conn, const char *filename, Oid lobjId);
```

#################### #######OID#*lobjId*############ ########OID############ ################## *lobjId*#InvalidOid#0#####lo_import_with_oid#####OID### #######lo_import######## ######################OID######InvalidOid#0#### ####

lo_import_with_oid#PostgreSQL 8.4#######8.1#######lo_create########### ##### 8.0##################InvalidOid#######

33.3.3.

##

```
int lo_export(PGconn *conn, Oid lobjId, const char *filename);
```

lobjId######################OID#####*filename*############################ ### ## #####1######-1#######

33.3.4.

##

```
int lo_open(PGconn *conn, Oid lobjId, int mode);
```

lobjId#################OID###### *mode*######################INV_READ######
##INV_WRITE########################### ###########libpq/libpq-fs.h#######
##########
lo_open##lo_read#lo_write#lo_lseek#lo_lseek64#lo_tell#lo_tell64#lo_truncate#lo_tr
######################### ######################### #####-1#######

##########INV_WRITE####INV_READ | INV_WRITE########### ###############
###########INV_READ#################### INV_READ################
##lo_open############
INV_WRITE######################
#######SELECT SQL#######
#REPEATABLE READ###########READ COMMITTED###################

##########SELECT##############INV_WRITE##########UPDATE##########
##lo_open###### #PostgreSQL 11##
################### ##########lo_compat_privileges####################

##########

```
inv_fd = lo_open(conn, inv_oid, INV_READ|INV_WRITE);
```

33.3.5.

#####

```
int lo_write(PGconn *conn, int fd, const char *buf, size_t len);
```

len#####*buf*#*len*###############*fd*################### *fd*#########lo_open####
########### ####################################*len*############ ##############
##-1######

len######size_t#################INT_MAX############ ######################
#############

33.3.6.

#####

```
int lo_read(PGconn *conn, int fd, char *buf, size_t len);
```

len#######*fd*##############*buf*#*len*#################### *fd*#########lo_open##
############# ################### ####################*len*############ #########
#######-1#######

len######size_t#################INT_MAX############ ######################
##############

33.3.7.

###

```
int lo_lseek(PGconn *conn, int fd, int offset, int whence);
```

#####*fd*################################*offset*################ *whence*#######
##SEEK_SET###################SEEK_CUR###########SEEK_END################
######## ####################-1#######

2GB#######################################

```
pg_int64 lo_lseek64(PGconn *conn, int fd, pg_int64 offset, int whence);
```

#####lo_lseek##########*offset*###2GB#########2GB############## lo_lseek#2GB####################################

lo_lseek64#PostgreSQL 9.3######### ################################-1######

33.3.8.

##

```
int lo_tell(PGconn *conn, int fd);
```

##########-1#######

####2GB#######################################

```
pg_int64 lo_tell64(PGconn *conn, int fd);
```

#####lo_tell##########2GB############## lo_tell#2GB####################

lo_tell64#PostgreSQL 9.3######### ################################-1######

33.3.9.

#################################

```
int lo_truncate(PGconn *conn, int fd, size_t len);
```

################*fd*#*len*######## *fd*#######lo_open################ *len*######## ####################################('\0')######## ###lo_truncate######### ######-1###

fd###########/##############

len######size_t############lo_truncate#INT_MAX############

2GB#######################################

```
int lo_truncate64(PGconn *conn, int fd, pg_int64 len);
```

#####lo_truncate##########2GB####*len*##############

lo_truncate#PostgreSQL 8.3########### ##############################-1#######

lo_truncate64#PostgreSQL 9.3######### ################################-1#######

33.3.10.

################################

```
int lo_close(PGconn *conn, int fd);
```

####*fd*#lo_open#################### ######lo_close#0###### ######-1######

##

33.3.11.

##################################

```
int lo_unlink(PGconn *conn, Oid lobjId);
```

lobjId################OID###### ####1######-1######

33.4.

SQL############################## 33.1#######

#33.1 SQL############

##	
lo_from_bytea (*loid* oid, *data* bytea) → oid ################*data*###### *loid*#0##############OID############OID# ####(#####OID####################)# ################OID###### lo_from_bytea(0, '\xffffff00') → 24528	
lo_put (*loid* oid, *offset* bigint, *data* bytea) → void ######################*data*########################## lo_put(24528, 1, '\xaa') →	
lo_get (*loid* oid [, *offset* bigint, *length* integer]) → bytea ############################ lo_get(24528, 0, 3) → \xffaaff	

##################################
########## SQL################
##lo_creat#lo_create#lo_unlink#lo_import#lo_export### ############

```
CREATE TABLE image (
    name            text,
    raster          oid
);


SELECT lo_creat(-1);       -- ###############OID#####


SELECT lo_create(43213);   -- OID 43213###################


SELECT lo_unlink(173454);  -- OID 173454################

INSERT INTO image (name, raster)
    VALUES ('beautiful image', lo_import('/etc/motd'));


INSERT INTO image (name, raster)  -- ###########OID######
    VALUES ('beautiful image', lo_import('/etc/motd', 68583));

SELECT lo_export(image.raster, '/tmp/motd') FROM image
    WHERE name = 'beautiful image';
```

#####lo_import###lo_export######################### ##2#################### ############################### ################################### ######### ## ############## ##

##

######lo_import#lo_export######GRANT############################## ################ ## ##################### ## ############# ## ######### ##############lo_import####lo_export##################### ############################## ##############################

##lo_read###lo_write############################## ######################### ######################### loread###lowrite######################

33.5.

33.1##libpq#################################### ############################## ################ ################src/test/examples######

#33.1 Libpq#########################

```
/*-------------------------------------------------------------------
 *
 * testlo.c

 *    libpq###################
 *
 * Portions Copyright (c) 1996-2024, PostgreSQL Global Development Group
 * Portions Copyright (c) 1994, Regents of the University of California
 *
 *
 * IDENTIFICATION
 *    src/test/examples/testlo.c
 *
 *-------------------------------------------------------------------
 */
#include <stdio.h>
#include <stdlib.h>

#include <sys/types.h>
#include <sys/stat.h>
#include <fcntl.h>
#include <unistd.h>

#include "libpq-fe.h"
#include "libpq/libpq-fs.h"

#define BUFSIZE         1024

/*
 * importFile -

 *    "in_filename"##############"lobjOid"##################
 *
 */
static Oid
```

```
importFile(PGconn *conn, char *filename)
{
    Oid         lobjId;
    int         lobj_fd;
    char        buf[BUFSIZE];
    int         nbytes,
                tmp;
    int         fd;

    /*

     * ###########
     */
    fd = open(filename, O_RDONLY, 0666);
    if (fd < 0)

    {                           /* #### */
        fprintf(stderr, "cannot open unix file\"%s\"\n", filename);
    }

    /*

     * ##############
     */
    lobjId = lo_creat(conn, INV_READ | INV_WRITE);
    if (lobjId == 0)
        fprintf(stderr, "cannot create large object");

    lobj_fd = lo_open(conn, lobjId, INV_WRITE);

    /*

     * Unix#######################
     */
    while ((nbytes = read(fd, buf, BUFSIZE)) > 0)
    {
        tmp = lo_write(conn, lobj_fd, buf, nbytes);
        if (tmp < nbytes)
            fprintf(stderr, "error while reading \"%s\"", filename);
    }

    close(fd);
    lo_close(conn, lobj_fd);

    return lobjId;
}

static void
pickout(PGconn *conn, Oid lobjId, int start, int len)
{
    int         lobj_fd;
    char       *buf;
    int         nbytes;
    int         nread;

    lobj_fd = lo_open(conn, lobjId, INV_READ);
    if (lobj_fd < 0)
        fprintf(stderr, "cannot open large object %u", lobjId);

    lo_lseek(conn, lobj_fd, start, SEEK_SET);
    buf = malloc(len + 1);

    nread = 0;
    while (len - nread > 0)
```

```
    {
        nbytes = lo_read(conn, lobj_fd, buf, len - nread);
        buf[nbytes] = '\0';
        fprintf(stderr, ">>> %s", buf);
        nread += nbytes;
        if (nbytes <= 0)
            break;              /* no more data? */
    }
    free(buf);
    fprintf(stderr, "\n");
    lo_close(conn, lobj_fd);
}

static void
overwrite(PGconn *conn, Oid lobjId, int start, int len)
{
    int         lobj_fd;
    char       *buf;
    int         nbytes;
    int         nwritten;
    int         i;

    lobj_fd = lo_open(conn, lobjId, INV_WRITE);
    if (lobj_fd < 0)
        fprintf(stderr, "cannot open large object %u", lobjId);

    lo_lseek(conn, lobj_fd, start, SEEK_SET);
    buf = malloc(len + 1);

    for (i = 0; i < len; i++)
        buf[i] = 'X';
    buf[i] = '\0';

    nwritten = 0;
    while (len - nwritten > 0)
    {
        nbytes = lo_write(conn, lobj_fd, buf + nwritten, len - nwritten);
        nwritten += nbytes;
        if (nbytes <= 0)
        {
            fprintf(stderr, "\nWRITE FAILED!\n");
            break;
        }
    }
    free(buf);
    fprintf(stderr, "\n");
    lo_close(conn, lobj_fd);
}


/*
 * exportFile -

 *    #########"lobjOid"#"out_filename"##############
 *
 */
static void
exportFile(PGconn *conn, Oid lobjId, char *filename)
{
    int         lobj_fd;
    char        buf[BUFSIZE];
    int         nbytes,
                tmp;
    int         fd;
```

```
    /*

     * ##############
     */
    lobj_fd = lo_open(conn, lobjId, INV_READ);
    if (lobj_fd < 0)
        fprintf(stderr, "cannot open large object %u", lobjId);

    /*

     * ###########
     */
    fd = open(filename, O_CREAT | O_WRONLY | O_TRUNC, 0666);
    if (fd < 0)

    {                           /* #### */
        fprintf(stderr, "cannot open unix file\"%s\"",
                filename);
    }

    /*

     * #############Unix##########
     */
    while ((nbytes = lo_read(conn, lobj_fd, buf, BUFSIZE)) > 0)
    {
        tmp = write(fd, buf, nbytes);
        if (tmp < nbytes)
        {
            fprintf(stderr, "error while writing \"%s\"",
                    filename);
        }
    }

    lo_close(conn, lobj_fd);
    close(fd);
}

static void
exit_nicely(PGconn *conn)
{
    PQfinish(conn);
    exit(1);
}

int
main(int argc, char **argv)
{
    char       *in_filename,
               *out_filename;
    char       *database;
    Oid         lobjOid;
    PGconn     *conn;
    PGresult   *res;

    if (argc != 4)
    {
        fprintf(stderr, "Usage: %s database_name in_filename out_filename
\n",
                argv[0]);
        exit(1);
    }
```

```
    database = argv[1];
    in_filename = argv[2];
    out_filename = argv[3];

    /*

     * #######
     */
    conn = PQsetdb(NULL, NULL, NULL, NULL, database);


    /* ######################## */
    if (PQstatus(conn) != CONNECTION_OK)
    {
        fprintf(stderr, "%s", PQerrorMessage(conn));
        exit_nicely(conn);
    }


    /* ###################################### */
    res = PQexec(conn,
                 "SELECT pg_catalog.set_config('search_path', '',
 false)");
    if (PQresultStatus(res) != PGRES_TUPLES_OK)
    {
        fprintf(stderr, "SET failed: %s", PQerrorMessage(conn));
        PQclear(res);
        exit_nicely(conn);
    }
    PQclear(res);

    res = PQexec(conn, "begin");
    PQclear(res);
    printf("importing file \"%s\" ...\n", in_filename);
/*  lobjOid = importFile(conn, in_filename); */
    lobjOid = lo_import(conn, in_filename);
    if (lobjOid == 0)
        fprintf(stderr, "%s\n", PQerrorMessage(conn));
    else
    {
        printf("\tas large object %u.\n", lobjOid);

        printf("picking out bytes 1000-2000 of the large object\n");
        pickout(conn, lobjOid, 1000, 1000);

        printf("overwriting bytes 1000-2000 of the large object with X's
\n");
        overwrite(conn, lobjOid, 1000, 1000);

        printf("exporting large object to file \"%s\" ...\n",
 out_filename);
/*      exportFile(conn, lobjOid, out_filename); */
        if (lo_export(conn, lobjOid, out_filename) < 0)
            fprintf(stderr, "%s\n", PQerrorMessage(conn));
    }

    res = PQexec(conn, "end");
    PQclear(res);
    PQfinish(conn);
    return 0;
}
```

#34# ECPG — C##########SQL

#####PostgreSQL#####SQL############### ########C#C++########### ###Linus Tolke#<linus@epact.se>##Michael Meskes#<meskes@postgresql.org>#### #####C# ############### C++#########C++#################

################ ########################SQL#######################

34.1.

####SQL######################C###########################SQL############### ######################## (*.pgc) #####SQL############ ######################C#### # (*.c) ########C################ (################ 34.10 ##########) #####ECPG# ########libpq###############SQL##### (ecpglib) ################################## PostgreSQL##########

C#####SQL###############SQL############### #######SQL#C######################## ######## ###########SQL############################## ###C########SQL###SQL#### ############SQL######################## PostgreSQL############################## ### #######SQL#################SQL############PostgreSQL#############

############SQL###################C#################################### ##### #######################

```
EXEC SQL ...;
```

##SQL#######C############### SQL##################################

####SQL###################C###########SQL############ ###SQL#########C######## ##### ##########C#######################C######## ########SQL################## ##C########SQL######### (####4.1.2.1#4.1.1########## ECPG #standard_conforming_strings#on#####################) ###########C######C# ###########

#############SQL###########

34.2.

#####################################

34.2.1.

###SQL####################

```
EXEC SQL CONNECT TO target [AS connection-name] [USER user-name];
```

target#############

- *dbname*[@*hostname*][:*port*]
- tcp:postgresql:*//hostname*[:*port*][*/dbname*][?*options*]
- unix:postgresql:*//localhost*[:*port*][*/dbname*][?*options*]
- ############SQL########
- ####################

- DEFAULT

DEFAULT################################### ##############################

#######(##########################)###############SQL##################
##*hostname*##############SQL##
########## *options*####SQL######################### ###############SQL#######
############# ##

#####################

- *username*
- *username/password*
- *username* IDENTIFIED BY *password*
- *username* USING *password*

#######*username*#*password*##SQL####SQL##########################

#####*options*########*keyword*=*value*#########(&)########### ###########libpq#####
#######(32.1.2#########)# *keyword*#*value*########################## *value*###&##
#################

(unix:####)######################localhost########## ######################
######*options*###host###########################

1######################*connection-name*####### ######1######################
####################SQL##

###CONNECT#############

```
EXEC SQL CONNECT TO mydb@sql.mydomain.com;

EXEC SQL CONNECT TO tcp:postgresql://sql.mydomain.com/mydb AS myconnection
 USER john;

EXEC SQL BEGIN DECLARE SECTION;
const char *target = "mydb@sql.mydomain.com";
const char *user = "john";
const char *passwd = "secret";
EXEC SQL END DECLARE SECTION;
 ...
EXEC SQL CONNECT TO :target USER :user USING :passwd;

/* #### EXEC SQL CONNECT TO :target USER :user/:passwd; */
```

################################## #################SQL####C################

##########SQL##################### ##
#########################

###search_path###########
#################### ####options=-c search_path=#*options*##########EXEC
SQL SELECT pg_catalog.set_config('search_path', '', false);#########
#####ECPG################SQL##########################

34.2.2.

####SQL#######SQL################################## ######################
########3#########

1#########SQL################## ##########

```
EXEC SQL AT connection-name SELECT ...;
```

##

################################
############################

2##################SQL########### ###SQL####

```
EXEC SQL SET CONNECTION connection-name;
```

###SQL#############################

#####################################

```
#include <stdio.h>

EXEC SQL BEGIN DECLARE SECTION;
    char dbname[1024];
EXEC SQL END DECLARE SECTION;

int
main()
{
    EXEC SQL CONNECT TO testdb1 AS con1 USER testuser;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
    EXEC SQL CONNECT TO testdb2 AS con2 USER testuser;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
    EXEC SQL CONNECT TO testdb3 AS con3 USER testuser;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;


    /* ####################"testdb3"####### */
    EXEC SQL SELECT current_database() INTO :dbname;
    printf("current=%s (should be testdb3)\n", dbname);


    /* "testdb2"#############"AT"### */
    EXEC SQL AT con2 SELECT current_database() INTO :dbname;
    printf("current=%s (should be testdb2)\n", dbname);


    /* ######"testdb1"####### */
    EXEC SQL SET CONNECTION con1;

    EXEC SQL SELECT current_database() INTO :dbname;
    printf("current=%s (should be testdb1)\n", dbname);

    EXEC SQL DISCONNECT ALL;
    return 0;
}
```

##################

```
current=testdb3 (should be testdb3)
current=testdb2 (should be testdb2)
current=testdb1 (should be testdb1)
```

3################SQL#############

```
EXEC SQL AT connection-name DECLARE statement-name STATEMENT;
EXEC SQL PREPARE statement-name FROM :dyn-string;
```

##SQL##############AT#######SQL######## ##################################### #########

```
#include <stdio.h>

EXEC SQL BEGIN DECLARE SECTION;
char dbname[128];
char *dyn_sql = "SELECT current_database()";
EXEC SQL END DECLARE SECTION;

int main(){
  EXEC SQL CONNECT TO postgres AS con1;
  EXEC SQL CONNECT TO testdb AS con2;
  EXEC SQL AT con1 DECLARE stmt STATEMENT;
  EXEC SQL PREPARE stmt FROM :dyn_sql;
  EXEC SQL EXECUTE stmt INTO :dbname;
  printf("%s\n", dbname);

  EXEC SQL DISCONNECT ALL;
  return 0;
}
```

#################testdb######################

```
postgres
```

34.2.3.

###########SQL########

```
EXEC SQL DISCONNECT [connection];
```

connection##############

- *connection-name*
- CURRENT
- ALL

#########################

#######################################

34.3. SQL########

####SQL##########SQL################ ##############

34.3.1. SQL####

#######:

```
EXEC SQL CREATE TABLE foo (number integer, ascii char(16));
EXEC SQL CREATE UNIQUE INDEX num1 ON foo(number);
EXEC SQL COMMIT;
```

####:

```
EXEC SQL INSERT INTO foo (number, ascii) VALUES (9999, 'doodad');
EXEC SQL COMMIT;
```

####:

```
EXEC SQL DELETE FROM foo WHERE number = 9999;
EXEC SQL COMMIT;
```

####:

```
EXEC SQL UPDATE foo
    SET ascii = 'foobar'
    WHERE number = 9999;
EXEC SQL COMMIT;
```

#######SELECT###### EXEC SQL ##; 34.3.2 ########## (##; 34.4.4.3.1 ########)

######:

```
EXEC SQL SELECT foo INTO :FooBar FROM table1 WHERE ascii = 'doodad';
```

########### SHOW ##################:

```
EXEC SQL SHOW search_path INTO :var;
```

:*something*################ ####C################## #######34.4#######

34.3.2.

##FETCH#####################

########SELECT:

```
EXEC SQL DECLARE foo_bar CURSOR FOR
    SELECT number, ascii FROM foo
    ORDER BY ascii;
EXEC SQL OPEN foo_bar;
EXEC SQL FETCH foo_bar INTO :FooBar, DooDad;
...
EXEC SQL CLOSE foo_bar;
EXEC SQL COMMIT;
```

###############DECLARE############################FETCH##########

> **##**
>
> ECPG# DECLARE ########PostgreSQL###########SQL#########OPEN ############################DECLARE#####################

34.3.3.

##########SQL##EXEC SQL COMMIT#################### ####SQL#######
##ecpg#####ecpg#####-t################ EXEC SQL SET AUTOCOMMIT TO ON#####
#psql#################################### ################################
################################ ##########EXEC SQL SET AUTOCOMMIT TO OFF#
###################

###########################:

```
EXEC SQL COMMIT
```

#################

```
EXEC SQL ROLLBACK
```

###################

```
EXEC SQL PREPARE TRANSACTION transaction_id
```

2##########################

```
EXEC SQL COMMIT PREPARED transaction_id
```

########################

```
EXEC SQL ROLLBACK PREPARED transaction_id
```

##########################

```
EXEC SQL SET AUTOCOMMIT TO ON
```

#############

```
EXEC SQL SET AUTOCOMMIT TO OFF
```

####################

34.3.4.

SQL#########################SQL#######################

SQL##PREPARE############### ###################### #?# #####:

```
EXEC SQL PREPARE stmt1 FROM "SELECT oid, datname FROM pg_database WHERE
 oid = ?";
```

SQL###########################SQL##PREPARE####USING################## #EXECUTE############

```
EXEC SQL EXECUTE stmt1 INTO :dboid, :dbname USING 1;
```

SQL### ################### #USING###############:

```
EXEC SQL PREPARE stmt1 FROM "SELECT oid,datname FROM pg_database WHERE oid
 > ?";
EXEC SQL DECLARE foo_bar CURSOR FOR stmt1;
```

```
/* ##############while######## */
EXEC SQL WHENEVER NOT FOUND DO BREAK;

EXEC SQL OPEN foo_bar USING 100;
...
while (1)
{
    EXEC SQL FETCH NEXT FROM foo_bar INTO :dboid, :dbname;
    ...
}
EXEC SQL CLOSE foo_bar;
```

###################################:

```
EXEC SQL DEALLOCATE PREPARE name;
```

PREPARE ######## PREPARE ######### ############################ 34.5 #########

34.4.

34.3#######SQL##########SQL################## ##SQL######################### ####SQL## ####SQL######## ############## #######################C#########SQL########################## ## ####SQL########SQL#########C#################### ######C################ ####

PostgreSQL#######ECPG##########################34.7########SQL###########

34.4.1.

####SQL####C######SQL##################### ################################ ##########################SQL##############C########### #########

```
EXEC SQL INSERT INTO sometable VALUES (:v1, 'foo', :v2);
```

##SQL###v1#v2###2##C############SQL############### ###############1######## #############

SQL###C##############SQL##########################

34.4.2.

#########C########SQL##

#################

```
EXEC SQL BEGIN DECLARE SECTION;
```

#############

```
EXEC SQL END DECLARE SECTION;
```

################C##############

```
int   x = 4;
char  foo[16], bar[16];
```

################################## ################################ ######### ##################################

```
EXEC SQL int i = 4;
```

#############################

#########C##################### ######################## ###SQL############### #####################

#############DECLARE################## ##################################### #######

34.4.3.

###################SQL#################### ############################ ##### #######SQL#####SELECT#FETCH#################### ###########INTO######### ########################### SELECT ##################FETCH ################# ###############

############

```
/*

 * #############
 * CREATE TABLE test1 (a int, b varchar(50));
 */

EXEC SQL BEGIN DECLARE SECTION;
int v1;
VARCHAR v2;
EXEC SQL END DECLARE SECTION;

 ...

EXEC SQL SELECT a, b INTO :v1, :v2 FROM test;
```

INTO#######FROM######## ###########INTO################################## ##

###FETCH##############

```
EXEC SQL BEGIN DECLARE SECTION;
int v1;
VARCHAR v2;
EXEC SQL END DECLARE SECTION;

 ...

EXEC SQL DECLARE foo CURSOR FOR SELECT a, b FROM test;

 ...

do
{
    ...
    EXEC SQL FETCH NEXT FROM foo INTO :v1, :v2;
    ...
} while (...);
```

#####INTO####################

34.4.4.

ECPG#########PostgreSQL#######C##
#######SQL###############PostgreSQL####################C####################
ECPG#####################ECPG##################

########2###########: ########PostgreSQL######`integer` # `text` ###################
############# ####PostgreSQL######`timestamp` # `numeric` #######################
#########; 34.4.4.2 #########

34.1###PostgreSQL########C#################### #####PostgreSQL###############
###############C##################################

#34.1 PostgreSQL#####C########

PostgreSQL####	######
`smallint`	`short`
`integer`	`int`
`bigint`	`long long int`
`decimal`	`decimal`[a]
`numeric`	`numeric`[a]
`real`	`float`
`double precision`	`double`
`smallserial`	`short`
`serial`	`int`
`bigserial`	`long long int`
`oid`	`unsigned int`
`character(`*n*`)`, `varchar(`*n*`)`, `text`	`char[`*n*`+1]`, `VARCHAR[`*n*`+1]`
`name`	`char[NAMEDATALEN]`
`timestamp`	`timestamp`[a]
`interval`	`interval`[a]
`date`	`date`[a]
`boolean`	`bool`[b]
`bytea`	`char *`, `bytea[`*n*`]`

[a] #########################; 34.4.4.2 ####
[b] ######### `ecpglib.h` ####

34.4.4.1.

`varchar#text`#############################2##########

`char` ### `char[]` #######C##########################

```
EXEC SQL BEGIN DECLARE SECTION;
    char str[50];
EXEC SQL END DECLARE SECTION;
```

############################# #########49#################################
############

########ECPG################ `VARCHAR` ######## `VARCHAR` ####################
`struct` ######## #########:

```
VARCHAR var[180];
```

##########:

```
struct varchar_var { int len; char arr[180]; } var;
```

arr #######1############### ######## VARCHAR ############################## ################ ##### len # arr ########################### ################## ##strlen(arr) # len

VARCHAR ##

char # VARCHAR #########SQL###########################

34.4.4.2.

ECPG###PostgreSQL## ###numeric, decimal, date, timestamp, interval ############### ################################## #int, long long int, ### char[]############### ##########################pgtypes ################################## 34.6 #########pgtypes########################## ########SQL##

############################## pgtypes###############34.6##########

34.4.4.2.1. timestamp, date

####timestamp ###ECPG#######################

######### timestamp ###########################:

```
#include <pgtypes_timestamp.h>
```

########## timestamp ############:

```
EXEC SQL BEGIN DECLARE SECTION;
timestamp ts;
EXEC SQL END DECLARE SECTION;
```

##################pgtypes################## #######timestamp ### PGTYPEStimestamp_to_asc() ###### text (ASCII) #########:

```
EXEC SQL SELECT now()::timestamp INTO :ts;

printf("ts = %s\n", PGTYPEStimestamp_to_asc(ts));
```

####################

```
ts = 2010-06-27 18:03:56.949343
```

###DATE################ ###### pgtypes_date.h ############## date ###### ##PGTYPESdate_to_asc() ###### DATE ### text ######### pgtypes################34.6 ##########

34.4.4.2.2. interval

interval ###### timestamp # date ######## ###interval ######################### ### ##

##############:

```
#include <stdio.h>
#include <stdlib.h>
#include <pgtypes_interval.h>

int
main(void)
{
EXEC SQL BEGIN DECLARE SECTION;
    interval *in;
EXEC SQL END DECLARE SECTION;

    EXEC SQL CONNECT TO testdb;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;

    in = PGTYPESinterval_new();
    EXEC SQL SELECT '1 min'::interval INTO :in;
    printf("interval = %s\n", PGTYPESinterval_to_asc(in));
    PGTYPESinterval_free(in);

    EXEC SQL COMMIT;
    EXEC SQL DISCONNECT ALL;
    return 0;
}
```

34.4.4.2.3. numeric, decimal

numeric # decimal ###### interval #######: ############################pgtypes# #################### pgtypes################34.6 ##########

decimal #################### ################# pgtypes########### numeric ###### #########

numeric ### decimal

```
#include <stdio.h>
#include <stdlib.h>
#include <pgtypes_numeric.h>

EXEC SQL WHENEVER SQLERROR STOP;

int
main(void)
{
EXEC SQL BEGIN DECLARE SECTION;
    numeric *num;
    numeric *num2;
    decimal *dec;
EXEC SQL END DECLARE SECTION;

    EXEC SQL CONNECT TO testdb;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;

    num = PGTYPESnumeric_new();
    dec = PGTYPESdecimal_new();

    EXEC SQL SELECT 12.345::numeric(4,2), 23.456::decimal(4,2)
 INTO :num, :dec;
```

```
    printf("numeric = %s\n", PGTYPESnumeric_to_asc(num, 0));
    printf("numeric = %s\n", PGTYPESnumeric_to_asc(num, 1));
    printf("numeric = %s\n", PGTYPESnumeric_to_asc(num, 2));


    /* decimal########decimal#numeric###### */
    num2 = PGTYPESnumeric_new();
    PGTYPESnumeric_from_decimal(dec, num2);

    printf("decimal = %s\n", PGTYPESnumeric_to_asc(num2, 0));
    printf("decimal = %s\n", PGTYPESnumeric_to_asc(num2, 1));
    printf("decimal = %s\n", PGTYPESnumeric_to_asc(num2, 2));

    PGTYPESnumeric_free(num2);
    PGTYPESdecimal_free(dec);
    PGTYPESnumeric_free(num);

    EXEC SQL COMMIT;
    EXEC SQL DISCONNECT ALL;
    return 0;
}
```

34.4.4.2.4. bytea

bytea######VARCHAR####### bytea#####################struct######## #########:

```
bytea var[180];
```

###########:

```
struct bytea_var { int len; char arr[180]; } var;
```

#####arr################### VARCHAR######'\0'################### #####ecpglib#
##hex########hex####################

> ##
>
> bytea####bytea_output#hex#####################

34.4.4.3.

###########typedef#####################

34.4.4.3.1.

############################## ########34.4.4.1 ######### char[] ### VARCHAR[] ##
###################### ## ##########
########################### FETCH ################ ###########################
######### ##

pg_database ##############################OID##############:

```
int
main(void)
{
EXEC SQL BEGIN DECLARE SECTION;
    int dbid[8];
    char dbname[8][16];
    int i;
```

```
EXEC SQL END DECLARE SECTION;

    memset(dbname, 0, sizeof(char)* 16 * 8);
    memset(dbid, 0, sizeof(int) * 8);

    EXEC SQL CONNECT TO testdb;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;


    /* ################ */
    EXEC SQL SELECT oid,datname INTO :dbid, :dbname FROM pg_database;

    for (i = 0; i < 8; i++)
        printf("oid=%d, dbname=%s\n", dbid[i], dbname[i]);

    EXEC SQL COMMIT;
    EXEC SQL DISCONNECT ALL;
    return 0;
}

####################################


oid=1, dbname=template1
oid=11510, dbname=template0
oid=11511, dbname=postgres
oid=313780, dbname=testdb
oid=0, dbname=
oid=0, dbname=
oid=0, dbname=
```

34.4.4.3.2.

```
########################################################### ####################
#############

######pg_database###########pg_database_size()##################OID######
########       ###############SELECT###############dbinfo_t#################
#FETCH########################


EXEC SQL BEGIN DECLARE SECTION;
    typedef struct
    {
       int oid;
       char datname[65];
       long long int size;
    } dbinfo_t;

    dbinfo_t dbval;
EXEC SQL END DECLARE SECTION;

    memset(&dbval, 0, sizeof(dbinfo_t));

    EXEC SQL DECLARE cur1 CURSOR FOR SELECT oid, datname,
 pg_database_size(oid) AS size FROM pg_database;
    EXEC SQL OPEN cur1;


    /* ##############while######## */
    EXEC SQL WHENEVER NOT FOUND DO BREAK;

    while (1)
```

```
    {

        /* ####1########### */
        EXEC SQL FETCH FROM cur1 INTO :dbval;


        /* ############# */
        printf("oid=%d, datname=%s, size=%lld\n", dbval.oid,
 dbval.datname, dbval.size);
    }

    EXEC SQL CLOSE cur1;

################################


oid=1, datname=template1, size=4324580
oid=11510, datname=template0, size=4243460
oid=11511, datname=postgres, size=4324580
oid=313780, datname=testdb, size=8183012

##################################### ############################ #############
######### size ########################


EXEC SQL BEGIN DECLARE SECTION;
    typedef struct
    {
       int oid;
       char datname[65];
    } dbinfo_t;

    dbinfo_t dbval;
    long long int size;
EXEC SQL END DECLARE SECTION;

    memset(&dbval, 0, sizeof(dbinfo_t));

    EXEC SQL DECLARE cur1 CURSOR FOR SELECT oid, datname,
 pg_database_size(oid) AS size FROM pg_database;
    EXEC SQL OPEN cur1;


    /* ##############while######## */
    EXEC SQL WHENEVER NOT FOUND DO BREAK;

    while (1)
    {

        /* ####1########### */
        EXEC SQL FETCH FROM cur1 INTO :dbval, :size;


        /* ############# */
        printf("oid=%d, datname=%s, size=%lld\n", dbval.oid,
 dbval.datname, size);
    }

    EXEC SQL CLOSE cur1;
```

34.4.4.3.3. typedef

################## typedef #############

```
EXEC SQL BEGIN DECLARE SECTION;
    typedef char mychartype[40];
    typedef long serial_t;
EXEC SQL END DECLARE SECTION;
```

##################:

```
EXEC SQL TYPE serial_t IS long;
```

########################### ####typedef####C################

#####[79.670330]#typedef#######################EXEC SQL#####SQL################ ############# ######typedef################SQL#EXEC SQL######################## ###############

```
EXEC SQL BEGIN DECLARE SECTION;
    typedef int start;
EXEC SQL END DECLARE SECTION;
...
EXEC SQL START TRANSACTION;
```

ECPG##START TRANSACTION############ ###########SQL##############typedef##### ##START######## #### ######typedef############################SQL################

> ##
>
> PostgreSQL v16######typedef####SQL################################ ##typedef##########SQL################## ###########ECPG################## #######PostgreSQL####################################

34.4.4.3.4.

########################### ## ########### # 34.7 #########

```
EXEC SQL BEGIN DECLARE SECTION;
    int   *intp;
    char **charp;
EXEC SQL END DECLARE SECTION;
```

34.4.5. #######SQL########

####################SQL#####ECPG################### ########################### #############

34.4.5.1.

SQL########ECPG######################## SQL#1#####C############################ ###### ###########ecpg##############C##########SQL################### SQL####### ######ecpg################################

######### ## ################ECPG#################### ######################## ################ ######### integer ######int ################### ######## varchar ### text #### char[] ### VARCHAR[] ###################

#########################:

```
CREATE TABLE t3 (
    ii integer[]
);

testdb=> SELECT * FROM t3;
     ii
-------------
 {1,2,3,4,5}
(1 row)
```

##############4############# int ############:

```
EXEC SQL BEGIN DECLARE SECTION;
int ii;
EXEC SQL END DECLARE SECTION;

EXEC SQL DECLARE cur1 CURSOR FOR SELECT ii[4] FROM t3;
EXEC SQL OPEN cur1;

EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{
    EXEC SQL FETCH FROM cur1 INTO :ii ;
    printf("ii=%d\n", ii);
}

EXEC SQL CLOSE cur1;
```

#################:

```
ii=4
```

##
#######:

```
EXEC SQL BEGIN DECLARE SECTION;
int ii_a[8];
EXEC SQL END DECLARE SECTION;

EXEC SQL DECLARE cur1 CURSOR FOR SELECT ii[1], ii[2], ii[3], ii[4] FROM
 t3;
EXEC SQL OPEN cur1;

EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{
    EXEC SQL FETCH FROM cur1 INTO :ii_a[0], :ii_a[1], :ii_a[2], :ii_a[3];
    ...
}
```

################

```
EXEC SQL BEGIN DECLARE SECTION;
int ii_a[8];
EXEC SQL END DECLARE SECTION;

EXEC SQL DECLARE cur1 CURSOR FOR SELECT ii FROM t3;
EXEC SQL OPEN cur1;
```

```
EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{

    /* ### */
    EXEC SQL FETCH FROM cur1 INTO :ii_a;
    ...
}
```

##

################### char[] ### VARCHAR[] ################## ############## 8.15.2 ######### ## ###

34.4.5.2.

####ECPG################################ ##################################: ## ###########################

#######################:

```
CREATE TYPE comp_t AS (intval integer, textval varchar(32));
CREATE TABLE t4 (compval comp_t);
INSERT INTO t4 VALUES ( (256, 'PostgreSQL') );
```

################################## ##########comp_t############################## ##########:

```
EXEC SQL BEGIN DECLARE SECTION;
int intval;
varchar textval[33];
EXEC SQL END DECLARE SECTION;


/* SELECT################## */
EXEC SQL DECLARE cur1 CURSOR FOR SELECT (compval).intval,
 (compval).textval FROM t4;
EXEC SQL OPEN cur1;

EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{

    /* ###################### */
    EXEC SQL FETCH FROM cur1 INTO :intval, :textval;

    printf("intval=%d, textval=%s\n", intval, textval.arr);
}

EXEC SQL CLOSE cur1;
```

######### FETCH ################################### ################### 34.4.4.3.2 ########## ################################## ######## intval # textval # comp_t ############### FETCH ##########

```
EXEC SQL BEGIN DECLARE SECTION;
typedef struct
```

```
{
    int intval;
    varchar textval[33];
} comp_t;

comp_t compval;
EXEC SQL END DECLARE SECTION;


/* SELECT################# */
EXEC SQL DECLARE cur1 CURSOR FOR SELECT (compval).intval,
 (compval).textval FROM t4;
EXEC SQL OPEN cur1;

EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{

    /* SELECT#########1########### */
    EXEC SQL FETCH FROM cur1 INTO :compval;

    printf("intval=%d, textval=%s\n", compval.intval,
 compval.textval.arr);
}

EXEC SQL CLOSE cur1;
```

FETCH ################# SELECT ################ ################## *
##############

```
...
EXEC SQL DECLARE cur1 CURSOR FOR SELECT (compval).* FROM t4;
EXEC SQL OPEN cur1;

EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{

    /* SELECT#########1########### */
    EXEC SQL FETCH FROM cur1 INTO :compval;

    printf("intval=%d, textval=%s\n", compval.intval,
 compval.textval.arr);
}
...
```

#########ECPG##

####char[] ### VARCHAR[] ################################### ################
############################

34.4.5.3.

##############ECPG################## #########char[]###VARCHAR[] ##########
##############################

###36.13#####complex########### ############(%f,%f)##36.13#complex_in()####
#complex_out()########### #########a#b##complex###(1,1)###(3,3)############
#######SELECT####

```
EXEC SQL BEGIN DECLARE SECTION;
    varchar a[64];
    varchar b[64];
EXEC SQL END DECLARE SECTION;

    EXEC SQL INSERT INTO test_complex VALUES ('(1,1)', '(3,3)');

    EXEC SQL DECLARE cur1 CURSOR FOR SELECT a, b FROM test_complex;
    EXEC SQL OPEN cur1;

    EXEC SQL WHENEVER NOT FOUND DO BREAK;

    while (1)
    {
        EXEC SQL FETCH FROM cur1 INTO :a, :b;
        printf("a=%s, b=%s\n", a.arr, b.arr);
    }

    EXEC SQL CLOSE cur1;
```

###############

```
a=(1,1), b=(3,3)
```

#################ECPG############################ECPG#######################
########### ##

#######

```
CREATE FUNCTION create_complex(r double, i double) RETURNS complex
LANGUAGE SQL
IMMUTABLE
AS $$ SELECT $1 * complex '(1,0')' + $2 * complex '(0,1)' $$;
```

############

```
EXEC SQL BEGIN DECLARE SECTION;
double a, b, c, d;
EXEC SQL END DECLARE SECTION;

a = 1;
b = 2;
c = 3;
d = 4;

EXEC SQL INSERT INTO test_complex VALUES (create_complex(:a, :b),
 create_complex(:c, :d));
```

###############

```
EXEC SQL INSERT INTO test_complex VALUES ('(1,2)', '(3,4)');
```

34.4.6.

#####NULL######## ####################NULL##################### #######NULL
#################NULL## ####
################NULL################ NULL#################### ####NULL###
###############

```
EXEC SQL BEGIN DECLARE SECTION;
VARCHAR val;
int val_ind;
EXEC SQL END DECLARE SECTION:

 ...

EXEC SQL SELECT b INTO :val :val_ind FROM test1;
```

##NULL##########val_ind#0###### ##NULL########### (Oracle##############34.16#########)

############# #############NULL#####################################

#######ecpg###-r no_indicator#########no-indicator########### no-indicator###NULL###integer#####################################int###INT_MIN####

34.5. ##SQL

#########################SQL######################### ########SQL################################## ########SQL####C################### ###################SQL####################

34.5.1. ###########SQL####

###SQL################EXECUTE IMMEDIATE ############## ######:

```
EXEC SQL BEGIN DECLARE SECTION;
const char *stmt = "CREATE TABLE test1 (...);";
EXEC SQL END DECLARE SECTION;

EXEC SQL EXECUTE IMMEDIATE :stmt;
```

EXECUTE IMMEDIATE###########SQL######DDL#INSERT#UPDATE#DELETE############ #######SQL######SELECT################### #################

34.5.2. ###########SQL####

###SQL## ###SQL################################SQL############## SQL###

```
EXEC SQL BEGIN DECLARE SECTION;
const char *stmt = "INSERT INTO test1 VALUES(?, ?);";
EXEC SQL END DECLARE SECTION;

EXEC SQL PREPARE mystmt FROM :stmt;
 ...
EXEC SQL EXECUTE mystmt USING 42, 'foobar';
```

################################

```
EXEC SQL DEALLOCATE PREPARE name;
```

34.5.3. ###########SQL####

########SQL#########EXECUTE ########### ##########INTO ########

```
EXEC SQL BEGIN DECLARE SECTION;
const char *stmt = "SELECT a, b, c FROM test1 WHERE a > ?";
int v1, v2;
VARCHAR v3[50];
EXEC SQL END DECLARE SECTION;

EXEC SQL PREPARE mystmt FROM :stmt;
 ...
EXEC SQL EXECUTE mystmt INTO :v1, :v2, :v3 USING 37;
```

EXECUTE#####INTO##USING##########################

####2### ############# 34.3.2 ##########

```
EXEC SQL BEGIN DECLARE SECTION;
char dbaname[128];
char datname[128];
char *stmt = "SELECT u.usename as dbaname, d.datname "
             "  FROM pg_database d, pg_user u "
             "  WHERE d.datdba = u.usesysid";
EXEC SQL END DECLARE SECTION;

EXEC SQL CONNECT TO testdb AS con1 USER testuser;
EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC SQL
 COMMIT;

EXEC SQL PREPARE stmt1 FROM :stmt;

EXEC SQL DECLARE cursor1 CURSOR FOR stmt1;
EXEC SQL OPEN cursor1;

EXEC SQL WHENEVER NOT FOUND DO BREAK;

while (1)
{
    EXEC SQL FETCH cursor1 INTO :dbaname,:datname;
    printf("dbaname=%s, datname=%s\n", dbaname, datname);
}

EXEC SQL CLOSE cursor1;

EXEC SQL COMMIT;
EXEC SQL DISCONNECT ALL;
```

34.6. pgtypes

pgtypes######PostgreSQL########C##############C########## #############C######## ########## #######PostgreSQL########## ##############

```
EXEC SQL BEGIN DECLARE SECTION;
   date date1;
   timestamp ts1, tsout;
   interval iv1;
   char *out;
EXEC SQL END DECLARE SECTION;

PGTYPESdate_today(&date1);
EXEC SQL SELECT started, duration INTO :ts1, :iv1 FROM datetbl WHERE
 d=:date1;
PGTYPEStimestamp_add_interval(&ts1, &iv1, &tsout);
```

```
out = PGTYPEStimestamp_to_asc(&tsout);
printf("Started + duration: %s\n", out);
PGTYPESchar_free(out);
```

34.6.1.

PGTYPESnumeric_to_asc################################ #####free#### #PGTYPESchar_free############ (###Windows########Windows################## ####################)

34.6.2. numeric

numeric ################### PostgreSQL################8.1######### ########### ######################## ####PGTYPESnumeric_new#PGTYPESnumeric_free##### ########## numeric ############# decimal ## #####

numeric ############:

PGTYPESnumeric_new

numeric

```
numeric *PGTYPESnumeric_new(void);
```

PGTYPESnumeric_free

numeric ####################

```
void PGTYPESnumeric_free(numeric *var);
```

PGTYPESnumeric_from_asc

####### numeric #######

```
numeric *PGTYPESnumeric_from_asc(char *str, char **endptr);
```

############ -2#.794#+3.44#592.49E07#-32.84e-4# ###################### #### ###### NULL ########## ##ECPG##################*endptr########### ################## ####endptr#### NULL ##########

PGTYPESnumeric_to_asc

numeric #num#########malloc#####################

```
char *PGTYPESnumeric_to_asc(numeric *num, int dscale);
```

numeric################dscale##10######### ###PGTYPESchar_free()###### #######

PGTYPESnumeric_add

2## numeric #######3### numeric ########

```
int PGTYPESnumeric_add(numeric *var1, numeric *var2, numeric *result);
```

#######var1###var2#########result###### ###0######-1######

`PGTYPESnumeric_sub`

2## numeric ########3### numeric #############

```
int PGTYPESnumeric_sub(numeric *var1, numeric *var2, numeric *result);
```

#######var1####var2######## #######result####### ###0######-1######

`PGTYPESnumeric_mul`

2## numeric ########3### numeric #############

```
int PGTYPESnumeric_mul(numeric *var1, numeric *var2, numeric *result);
```

#######var1###var2######### #######result####### ###0######-1######

`PGTYPESnumeric_div`

2## numeric ########3### numeric #############

```
int PGTYPESnumeric_div(numeric *var1, numeric *var2, numeric *result);
```

#######var1###var2###### #######result####### ###0######-1######

`PGTYPESnumeric_cmp`

2##numeric##########

```
int PGTYPESnumeric_cmp(numeric *var1, numeric *var2)
```

#####2##numeric########## ######INT_MAX###### ####################:

- var1#var2#######1#
- var1#var2#######-1#
- var1#var2######0#

`PGTYPESnumeric_from_int`

int####numeric##########

```
int PGTYPESnumeric_from_int(signed int int_val, numeric *var);
```

#####signed int#########numeric###var######## ###0####-1######

`PGTYPESnumeric_from_long`

long int####numeric##########

```
int PGTYPESnumeric_from_long(signed long int long_val, numeric *var);
```

#####signed long int#########numeric###var######## ###0####-1######

`PGTYPESnumeric_copy`

numeric ###### numeric ###########

```
int PGTYPESnumeric_copy(numeric *src, numeric *dst);
```

######src#########dst############## ###0####-1######

PGTYPESnumeric_from_double

double##### numeric #########

```
int  PGTYPESnumeric_from_double(double d, numeric *dst);
```

#####double##############dst############## ###0####-1######

PGTYPESnumeric_to_double

numeric ####double########

```
int PGTYPESnumeric_to_double(numeric *nv, double *dp)
```

#####nv##### numeric ######dp#####double######### ###0###################-1## #### ########################errno#PGTYPES_NUM_OVERFLOW########

PGTYPESnumeric_to_int

numeric####int########

```
int PGTYPESnumeric_to_int(numeric *nv, int *ip);
```

#####nv#####numeric######ip#####integer######### ###0###################-1### ### ########################errno#PGTYPES_NUM_OVERFLOW########

PGTYPESnumeric_to_long

numeric ####long########

```
int PGTYPESnumeric_to_long(numeric *nv, long *lp);
```

#####[70.434783]######nv#####numeric######ip#####integer######### ###0######## ############-1###### ######################## #errno#PGTYPES_NUM_OVERFLOW######## #########nv###########lp######### ######### #######0################################-1###### ################ #errno#PGTYPES_NUM_####################errno##PGTYPES_NUM_######### ######

PGTYPESnumeric_to_decimal

numeric #### decimal ########

```
int PGTYPESnumeric_to_decimal(numeric *src, decimal *dst);
```

#####src##### numeric ######dst##### decimal ######### ###0################### #-1###### ########################errno#PGTYPES_NUM_OVERFLOW########

PGTYPESnumeric_from_decimal

decimal #### numeric ########

```
int PGTYPESnumeric_from_decimal(decimal *src, numeric *dst);
```

#####src##### decimal ######dst##### numeric ######### ###0##########-1###### decimal ###### numeric ##################################

34.6.3.

C################SQL################ PostgreSQL################8.5##########

###########################:

PGTYPESdate_from_timestamp

####################

```
date PGTYPESdate_from_timestamp(timestamp dt);
```

##

PGTYPESdate_from_asc

#################

```
date PGTYPESdate_from_asc(char *str, char **endptr);
```

#####C#char*####str#C#char*####endptr############ ##ECPG################## ##*endptr############################## ####endptr####NULL##########

#######MDY######################## ##ECPG####################

34.2###############

#34.2 ###PGTYPESdate_from_asc#####

##	##
January 8, 1999	January 8, 1999
1999-01-08	January 8, 1999
1/8/1999	January 8, 1999
1/18/1999	January 18, 1999
01/02/03	February 1, 2003
1999-Jan-08	January 8, 1999
Jan-08-1999	January 8, 1999
08-Jan-1999	January 8, 1999
99-Jan-08	January 8, 1999
08-Jan-99	January 8, 1999
08-Jan-06	January 8, 2006
Jan-08-99	January 8, 1999
19990108	ISO 8601; January 8, 1999
990108	ISO 8601; January 8, 1999
1999.008	######
J2451187	#####
January 8, 99 BC	###99#

PGTYPESdate_to_asc

##################

```
char *PGTYPESdate_to_asc(date dDate);
```

###############dDate###### #####1999-01-18####YYYY-MM-DD########### ## #PGTYPESchar_free()#############

PGTYPESdate_julmdy

#######################

```
void PGTYPESdate_julmdy(date d, int *mdy);
```

#########d##3##########mdy############ ##############mdy[0]#####mdy[1]### ###mdy[2]########

PGTYPESdate_mdyjul

#############3#####################

```
void PGTYPESdate_mdyjul(int *mdy, date *jdate);
```

######1########3##########(mdy)#2##################################

PGTYPESdate_dayofweek

##################

```
int PGTYPESdate_dayofweek(date d);
```

#############date###d#######################

- 0 - ##
- 1 - ##
- 2 - ##
- 3 - ##
- 4 - ##
- 5 - ##
- 6 - ##

PGTYPESdate_today

############

```
void PGTYPESdate_today(date *d);
```

################date###(d)################

PGTYPESdate_fmt_asc

##########date################

```
int PGTYPESdate_fmt_asc(date dDate, char *fmtstring, char *outbuf);
```

##########date#(dDate)######(fmtstring)####################(outbuf)#######

####0####################

##############################

- dd - ######
- mm - ######
- yy - #######
- yyyy - #######
- ddd - #########
- mmm - ########

#########################

34.3################### ########################### ############1959#11#23##

#34.3 ###PGTYPESdate_fmt_asc#####

##	##
mmddyy	112359
ddmmyy	231159
yymmdd	591123
yy/mm/dd	59/11/23
yy mm dd	59 11 23
yy.mm.dd	59.11.23
.mm.yyyy.dd.	.11.1959.23.
mmm. dd, yyyy	Nov. 23, 1959
mmm dd yyyy	Nov 23 1959
yyyy dd mm	1959 23 11
ddd, mmm. dd, yyyy	Mon, Nov. 23, 1959
(ddd) mmm. dd, yyyy	(Mon) Nov. 23, 1959

PGTYPESdate_defmt_asc

##########C#char*#####date#########

```
int PGTYPESdate_defmt_asc(date *d, char *fmt, char *str);
```

###########################(d)################(fmt)#############C#char*### (str)####### ################################ ##############1:1############## ## #####################yy###yyyy########mm#########dd#######

34.4###################

#34.4 ###rdefmtdate#####

##	###	##
ddmmyy	21-2-54	1954-02-21

##	###	##
ddmmyy	2-12-54	1954-12-02
ddmmyy	20111954	1954-11-20
ddmmyy	130464	1964-04-13
mmm.dd.yyyy	MAR-12-1967	1967-03-12
yy/mm/dd	1954, February 3rd	1954-02-03
mmm.dd.yyyy	041269	1969-04-12
yy/mm/dd	In the year 2525, in the month of July, mankind will be alive on the 28th day	2525-07-28
dd-mm-yy	I said on the 28th of July in the year 2525	2525-07-28
mmm.dd.yyyy	9/14/58	1958-09-14
yy/mm/dd	47/03/29	1947-03-29
mmm.dd.yyyy	oct 28 1975	1975-10-28
mmddyy	Nov 14th, 1985	1985-11-14

34.6.4. timestamp#

C####################SQL####################### PostgreSQL##############8.5#######

###########################:

PGTYPEStimestamp_from_asc

#################################

```
timestamp PGTYPEStimestamp_from_asc(char *str, char **endptr);
```

#############(str)#C char*######(endptr)####### ##ECPG#################*endptr########################### ####endptr#### NULL ###########

######################### #####PGTYPESInvalidTimestamp#####errno#PGTYPES_TS_BAD_TIMESTAMP####### #################PGTYPESInvalidTimestamp#########

####ECPG############### ####################PostgreSQL#########################

34.5#################

#34.5 ###PGTYPEStimestamp_from_asc#####

##	##
1999-01-08 04:05:06	1999-01-08 04:05:06
January 8 04:05:06 1999 PST	1999-01-08 04:05:06
1999-Jan-08 04:05:06.789-8	1999-01-08 04:05:06.789 (###########)

##	##
J2451187 04:05-08:00	1999-01-08 04:05:00 (##########)

PGTYPEStimestamp_to_asc

date##C char*##########

```
char *PGTYPEStimestamp_to_asc(timestamp tstamp);
```

#####timestamp##tstamp############timestamp########################## ## #PGTYPESchar_free()#############

PGTYPEStimestamp_current

#################

```
void PGTYPEStimestamp_current(timestamp *ts);
```

####################ts#####timestamp##########

PGTYPEStimestamp_fmt_asc

##########timestamp####C char*#######

```
int PGTYPEStimestamp_fmt_asc(timestamp *ts, char *output, int str_len,
 char *fmtstr);
```

##################timestamp# (ts)############(output)################ (str_len)############(fmtstr)#######

##########0###### ###################

########################## #####libc#strftime############### ############## #######

- %A - ###############
- %a - ###############
- %B - ##############
- %b - ##############
- %C - ##100####10#########1########0########
- %c - #################
- %D - %m/%d/%y######
- %d - ###10###(01–31)########
- %E* %O* - POSIX######## %Ec %EC %Ex %EX %Ey %EY %Od %Oe %OH %OI %Om %OM %OS %Ou %OU %OV %Ow %OW %Oy ##########################

  ####%OB##

- %e - ##10###(1–31)########1################

- %F - %Y-%m-%d######
- %G - ####10############## ################################
- %g -%G####################10##(00–99)######
- %H - 10####(24####)#######(00–23)#
- %h - %b######
- %I - 10#####12############(01–12)#
- %j - 10#############(001–366)#
- %k - 10####(24####)#######(0–23)#1##################
- %l - 10#####12############(1–12)#1##################
- %M - 10###########(00–59)#
- %m -10###########(01–12)#
- %n - ##########
- %O* - %E*######
- %p - #########################
- %R - %H:%M######
- %r - %I:%M:%S %p######
- %S - 10###########(00–60)#
- %s - #####UTC#############
- %T - %H:%M:%S######
- %t - ##########
- %U - 10#######################(00–53)#
- %u - 10#######################(1–7)#
- %V - 10##########################(01–53)# ######1#1######4##############1#### ## #####################1######
- %v - %e-%b-%Y######
- %W - 10##########################(00–53)#
- %w - 10#######################(0–6)#
- %X - ################
- %x - ################
- %Y - 10##############
- %y - 10##############(00–99)#
- %Z - #############

- `%z` - UTC################## UTC########################## ####2########### ########################RFC 822[1]################
- `%+` - #################
- `%-*` - GNU libc########################
- `$_*` - GNU libc##########################
- `%0*` - GNU libc#########0############
- `%%` - %########

`PGTYPEStimestamp_sub`

##################interval###########

```
int PGTYPEStimestamp_sub(timestamp *ts1, timestamp *ts2, interval *iv);
```

#####`ts1`################`ts2`####################`iv`#####interval#############

###########0#######################

`PGTYPEStimestamp_defmt_asc`

##################timestamp########

```
int PGTYPEStimestamp_defmt_asc(char *str, char *fmt, timestamp *d);
```

#####`str`#######################`fmt`######################### ###`d`########## ######

#####`fmt`# NULL ###################`%Y-%m-%d %H:%M:%S`############

###`PGTYPEStimestamp_fmt_asc`####### ##############################

`PGTYPEStimestamp_add_interval`

timestamp####interval##########

```
int PGTYPEStimestamp_add_interval(timestamp *tin, interval *span,
 timestamp *tout);
```

#####timestamp###`tin`#######interval###`span`############ ####interval##timestamp## #########timestamp##`tout`#############

##########0###### ###################

`PGTYPEStimestamp_sub_interval`

timestamp#####interval###########

```
int PGTYPEStimestamp_sub_interval(timestamp *tin, interval *span,
 timestamp *tout);
```

#####`tin`#####timestamp#####`span`#####interval######### ###`tout`##############

##########0#####################

[1] https://datatracker.ietf.org/doc/html/rfc822

34.6.5. interval#

C####interval##################SQL#interval################ PostgreSQL##############
###8.5##########

###########interval############

`PGTYPESinterval_new`

########interval###############

```
interval *PGTYPESinterval_new(void);
```

`PGTYPESinterval_free`

##########interval##############

```
void PGTYPESinterval_free(interval *intvl);
```

`PGTYPESinterval_from_asc`

########interval########

```
interval *PGTYPESinterval_from_asc(char *str, char **endptr);
```

##########`str`###########interval############ ##ECPG##################
#`*endptr`############################## ####`endptr`#### NULL ###########

`PGTYPESinterval_to_asc`

interval################

```
char *PGTYPESinterval_to_asc(interval *span);
```

#####`span`#####interval####C char*####### ###`@ 1 day 12 hours 59 mins 10 secs`######## ###`PGTYPESchar_free()`#############

`PGTYPESinterval_copy`

interval###########

```
int PGTYPESinterval_copy(interval *intvlsrc, interval *intvldest);
```

######`intvlsrc`#####interval#### `intvldest`############## ##################
##################

34.6.6. decimal#

decimal##numeric######## ###########30############# #############numeric##
###decimal########################## #######`PGTYPESdecimal_new`##
#`PGTYPESdecimal_free`########## 34.15#####Informix#######decimal##############
##

##########decimal########### ####`libcompat`####################

`PGTYPESdecimal_new`

##########decimal################

```
decimal *PGTYPESdecimal_new(void);
```

PGTYPESdecimal_free

decimal####################

```
void PGTYPESdecimal_free(decimal *var);
```

34.6.7. pgtypeslib#errno#

PGTYPES_NUM_BAD_NUMERIC

###numeric#######numeric#####################################

PGTYPES_NUM_OVERFLOW

############## numeric#####################numeric#########################
########

PGTYPES_NUM_UNDERFLOW

############## numeric#####################numeric#########################
########

PGTYPES_NUM_DIVIDE_ZERO

##############

PGTYPES_DATE_BAD_DATE

PGTYPESdate_from_asc##################

PGTYPES_DATE_ERR_EARGS

PGTYPESdate_defmt_asc###############

PGTYPES_DATE_ERR_ENOSHORTDATE

PGTYPESdate_defmt_asc###########################

PGTYPES_INTVL_BAD_INTERVAL

PGTYPESinterval_from_asc################## ###
#PGTYPESinterval_to_asc################

PGTYPES_DATE_ERR_ENOTDMY

PGTYPESdate_defmt_asc#####/#/##################

PGTYPES_DATE_BAD_DAY

PGTYPESdate_defmt_asc####################

PGTYPES_DATE_BAD_MONTH

PGTYPESdate_defmt_asc####################

PGTYPES_TS_BAD_TIMESTAMP

PGTYPEStimestamp_from_asc######################## ###
#PGTYPEStimestamp_to_asc######timestamp#########

PGTYPES_TS_ERR_EINFTIME

##################################

34.6.8. pgtypeslib######

PGTYPESInvalidTimestamp

#############timestamp##### ###########PGTYPEStimestamp_from_asc### ######## timestamp############PGTYPESInvalidTimestamp############ ######### ###1899-12-31 23:59:59####### ########## ##PGTYPEStimestamp_from_asc##########PGTYPESInvalidTimestamp##### ######errno != 0#########

34.7.

SQL#####SELECT#FETCH#DESCRIBE#################### SQL#####1######### #######1#################### ####SQL#################################### PostgreSQL##############2########SQL######C###SQLDA########

34.7.1. ####SQL#####

####SQL##################################1########1####################

SQL##

```
EXEC SQL ALLOCATE DESCRIPTOR identifier;
```

########################## ##########################

```
EXEC SQL DEALLOCATE DESCRIPTOR identifier;
```

#############INTO##

```
EXEC SQL FETCH NEXT FROM mycursor INTO SQL DESCRIPTOR mydesc;
```

##

##DESCRIBE#############

```
EXEC SQL BEGIN DECLARE SECTION;
char *sql_stmt = "SELECT * FROM table1";
EXEC SQL END DECLARE SECTION;

EXEC SQL PREPARE stmt1 FROM :sql_stmt;
EXEC SQL DESCRIBE stmt1 INTO SQL DESCRIPTOR mydesc;
```

PostgreSQL 9.0######SQL############# ####DESCRIPTOR###SQL DESCRIPTOR##### SQL############ ##########SQL###########SQLDA########### 34.7.2#########

DESCRIBE###FETCH####INTO###USING#################### ################## #########

############################## ################################## ############## ################################

```
EXEC SQL GET DESCRIPTOR name :hostvar = field;
```

######*COUNT*###########1########### ######################## ################ ### ########integer########### ##################################

EXEC SQL GET DESCRIPTOR *name* VALUE *num* :*hostvar* = *field*;

num################################ ##################

CARDINALITY (##)

###########

DATA

########## ###############################

DATETIME_INTERVAL_CODE (##)

TYPE#9####DATETIME_INTERVAL_CODE##DATE##1#TIME##2#TIMESTAMP#
#3#TIME WITH TIME ZONE##4#TIMESTAMP WITH TIME ZONE##5#########

DATETIME_INTERVAL_PRECISION (##)

######

INDICATOR (##)

#NULL#################

KEY_MEMBER (##)

##########

LENGTH (##)

#############

NAME (###)

#####

NULLABLE (##)

##########

OCTET_LENGTH (##)

################

PRECISION (##)

#numeric########

RETURNED_LENGTH (##)

##########

RETURNED_OCTET_LENGTH (##)

################

SCALE (##)

#numeric########

TYPE (##)

##############

EXECUTE#DECLARE###OPEN####INTO###USING######### ######################### ######################### USING SQL DESCRIPTOR *name*###################### ###### ####SQL#########SQL#########

```
EXEC SQL SET DESCRIPTOR name VALUE num field = :hostvar;
```

PostgreSQL##1##FETCH###1################################ ##################### ##

```
EXEC SQL BEGIN DECLARE SECTION;
int id[5];
EXEC SQL END DECLARE SECTION;

EXEC SQL FETCH 5 FROM mycursor INTO SQL DESCRIPTOR mydesc;

EXEC SQL GET DESCRIPTOR mydesc VALUE 1 :id = DATA;
```

34.7.2. SQLDA#####

SQLDA#####################################C######## 1############1######### ##

```
EXEC SQL include sqlda.h;
sqlda_t         *mysqlda;

EXEC SQL FETCH 3 FROM mycursor INTO DESCRIPTOR mysqlda;
```

SQL######################### 34.7.1#INTO###USING######################### DESCRIBE####INTO#########DESCRIPTOR###############

```
EXEC SQL DESCRIBE prepared_statement INTO mysqlda;
```

SQLDA##########################

1. ########################
2. ######SQLDA######
3. ########SQLDA############################
4. ###SQLDA##########
5. ###############SQLDA######
6. ###SQLDA############################
7. #########
8. ###SQLDA###################

34.7.2.1. SQLDA######

SQLDA#sqlda_t#sqlvar_t#struct sqlname####################

###

PostgreSQL#SQLDA#IBM DB2########################### #####DB2#SQLDA## ##########PostgreSQL#SQLDA#################

34.7.2.1.1. sqlda_t###

sqlda_t######SQLDA##### ################ #######sqlda_t####desc_next#### ################################ ################## ######################### ##sqlda_t####desc_next#########################

sqlda_t###########

```
struct sqlda_struct
{
    char            sqldaid[8];
    long            sqldabc;
    short           sqln;
    short           sqld;
    struct sqlda_struct *desc_next;
    struct sqlvar_struct sqlvar[1];
};

typedef struct sqlda_struct sqlda_t;
```

################

sqldaid

####"SQLDA "#############

sqldabc

#############################

sqln

USING##########OPEN#DECLARE#EXECUTE#################################### #### SELECT#EXECUTE#FETCH##################sqld#######

sqld

######################

desc_next

##########################SQLDA########## desc_next###################### ####

sqlvar

################

34.7.2.1.2. sqlvar_t###

sqlvar_t########################## ###############

```
struct sqlvar_struct
```

```
{
    short          sqltype;
    short          sqllen;
    char          *sqldata;
    short         *sqlind;
    struct sqlname sqlname;
};

typedef struct sqlvar_struct sqlvar_t;
```

################

sqltype

####################### ######ecpgtype.h#enum ECPGttype#########

sqllen

################## ###ECPGt_int##4######

sqldata

############### ######34.4.4#######

sqlind

####NULL########### 0##NULL##-1#NULL#######

sqlname

###########

34.7.2.1.3. struct sqlname###

struct sqlname############ sqlvar_t################ ##############

```
#define NAMEDATALEN 64

struct sqlname
{
        short           length;
        char            data[NAMEDATALEN];
};
```

################

length

################

data

################

34.7.2.2. SQLDA###############

SQLDA##############################

1. ############sqlda_t#########

2. ####SQLDA#################FETCH/EXECUTE/DESCRIBE#######

3. sqlda_t######sqln##########################
4. sqlda_t######sqlvar[0]#sqlvar[1]##############
5. sqlda_t######desc_next###########sqlda_t#########
6. #############

###SQLDA####################

############sqlda_t#########

```
sqlda_t *sqlda1;
```

########SQLDA####### ###FETCH########

```
EXEC SQL FETCH NEXT FROM cur1 INTO DESCRIPTOR sqlda1;
```

###########################

```
sqlda_t *cur_sqlda;

for (cur_sqlda = sqlda1;
     cur_sqlda != NULL;
     cur_sqlda = cur_sqlda->desc_next)
{
    ...
}
```

###############sqlvar_t#######################

```
for (i = 0; i < cur_sqlda->sqld; i++)
{
    sqlvar_t v = cur_sqlda->sqlvar[i];
    char *sqldata = v.sqldata;
    short sqllen  = v.sqllen;
    ...
}
```

###########sqlvar_t######sqltype######## ###########sqlvar##############
######################

```
char var_buf[1024];

switch (v.sqltype)
{
    case ECPGt_char:
        memset(&var_buf, 0, sizeof(var_buf));
        memcpy(&var_buf, sqldata, (sizeof(var_buf) <= sqllen ?
 sizeof(var_buf) - 1 : sqllen));
        break;


    case ECPGt_int: /* ## */
        memcpy(&intval, sqldata, sqllen);
        snprintf(var_buf, sizeof(var_buf), "%d", intval);
        break;

    ...
}
```

34.7.2.3. SQLDA##################

#######################SQLDA####################

1. ########################
2. ###SQLDA###sqlda_t##########
3. ###SQLDA#########sqlda_t##############
4. #######################
5. ###SQLDA##############

##########

###############

```
EXEC SQL BEGIN DECLARE SECTION;
char query[1024] = "SELECT d.oid, * FROM pg_database d, pg_stat_database s
 WHERE d.oid = s.datid AND (d.datname = ? OR d.oid = ?)";
EXEC SQL END DECLARE SECTION;

EXEC SQL PREPARE stmt1 FROM :query;
```

##SQLDA###########sqlda_t########sqln################ ##################### #################(###### - 1) * sizeof(sqlvar_t)############################ ######### #######################

```
sqlda_t *sqlda2;

sqlda2 = (sqlda_t *) malloc(sizeof(sqlda_t) + sizeof(sqlvar_t));
memset(sqlda2, 0, sizeof(sqlda_t) + sizeof(sqlvar_t));


sqlda2->sqln = 2; /* ####### */
```

###########sqlvar[]################ #####SQLDA################################ #### ####################"postgres"#integer####1###

```
sqlda2->sqlvar[0].sqltype = ECPGt_char;
sqlda2->sqlvar[0].sqldata = "postgres";
sqlda2->sqlvar[0].sqllen  = 8;

int intval = 1;
sqlda2->sqlvar[1].sqltype = ECPGt_int;
sqlda2->sqlvar[1].sqldata = (char *) &intval;
sqlda2->sqlvar[1].sqllen  = sizeof(intval);
```

#########SQLDA#####################################

```
EXEC SQL OPEN cur1 USING DESCRIPTOR sqlda2;
```

######################SQLDA#########SQLDA################################

```
free(sqlda2);
```

34.7.2.4. SQLDA##################

###

###########pg_database#pg_stat_database##############OID####################
#postgres#OID1##############################

#######SQLDA#####SQLDA#######

```
EXEC SQL include sqlda.h;


sqlda_t *sqlda1; /* ##### */
sqlda_t *sqlda2; /* ##### */
```

##

```
int
main(void)
{
    EXEC SQL BEGIN DECLARE SECTION;
    char query[1024] = "SELECT d.oid,* FROM pg_database d,
 pg_stat_database s WHERE d.oid=s.datid AND ( d.datname=? OR d.oid=? )";
    EXEC SQL END DECLARE SECTION;

    EXEC SQL CONNECT TO testdb AS con1 USER testuser;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;

    EXEC SQL PREPARE stmt1 FROM :query;
    EXEC SQL DECLARE cur1 CURSOR FOR stmt1;
```

################SQLDA############## ###SQLDA######################sqln###
########sqlvar#####sqltype#sqldata#sqllen######

```
    /* #########SQLDA######### */
    sqlda2 = (sqlda_t *) malloc(sizeof(sqlda_t) + sizeof(sqlvar_t));
    memset(sqlda2, 0, sizeof(sqlda_t) + sizeof(sqlvar_t));

    sqlda2->sqln = 2; /* ###### */

    sqlda2->sqlvar[0].sqltype = ECPGt_char;
    sqlda2->sqlvar[0].sqldata = "postgres";
    sqlda2->sqlvar[0].sqllen  = 8;

    intval = 1;
    sqlda2->sqlvar[1].sqltype = ECPGt_int;
    sqlda2->sqlvar[1].sqldata = (char *)&intval;
    sqlda2->sqlvar[1].sqllen  = sizeof(intval);
```

###SQLDA############SQLDA#############

```
    /* ################## */
    EXEC SQL OPEN cur1 USING DESCRIPTOR sqlda2;
```

############SQLDA########## ##############################FETCH#############
#######

```
    while (1)
    {
        sqlda_t *cur_sqlda;
```

```
        /* ############## */
        EXEC SQL FETCH NEXT FROM cur1 INTO DESCRIPTOR sqlda1;
```

###sqlda_t################SQLDA##################

```
    for (cur_sqlda = sqlda1 ;
         cur_sqlda != NULL ;
         cur_sqlda = cur_sqlda->desc_next)
    {
        ...
```

################## ###sqld###########sqlvar[0]######### ####sqlda_t#######
###

```
        /* 1############ */
        for (i = 0; i < sqlda1->sqld; i++)
        {
            sqlvar_t v = sqlda1->sqlvar[i];
            char *sqldata = v.sqldata;
            short sqllen  = v.sqllen;

            strncpy(name_buf, v.sqlname.data, v.sqlname.length);
            name_buf[v.sqlname.length] = '\0';
```

#########v########### #######v.sqltype###########################

```
            switch (v.sqltype) {
                int intval;
                double doubleval;
                unsigned long long int longlongval;

                case ECPGt_char:
                    memset(&var_buf, 0, sizeof(var_buf));
                    memcpy(&var_buf, sqldata, (sizeof(var_buf) <= sqllen ?
 sizeof(var_buf)-1 : sqllen));
                    break;


                case ECPGt_int: /* ## */
                    memcpy(&intval, sqldata, sqllen);
                    snprintf(var_buf, sizeof(var_buf), "%d", intval);
                    break;

                ...

                default:
                    ...
            }

            printf("%s = %s (type: %d)\n", name_buf, var_buf, v.sqltype);
        }
```

#####################################

```
    EXEC SQL CLOSE cur1;
    EXEC SQL COMMIT;

    EXEC SQL DISCONNECT ALL;
```

######### 34.1######

#34.1 SQLDA#######

```
#include <stdlib.h>
#include <string.h>
#include <stdlib.h>
#include <stdio.h>
#include <unistd.h>

EXEC SQL include sqlda.h;


sqlda_t *sqlda1; /* ##### */
sqlda_t *sqlda2; /* ##### */

EXEC SQL WHENEVER NOT FOUND DO BREAK;
EXEC SQL WHENEVER SQLERROR STOP;

int
main(void)
{
    EXEC SQL BEGIN DECLARE SECTION;
    char query[1024] = "SELECT d.oid,* FROM pg_database d,
 pg_stat_database s WHERE d.oid=s.datid AND ( d.datname=? OR d.oid=? )";

    int intval;
    unsigned long long int longlongval;
    EXEC SQL END DECLARE SECTION;

    EXEC SQL CONNECT TO uptimedb AS con1 USER uptime;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;

    EXEC SQL PREPARE stmt1 FROM :query;
    EXEC SQL DECLARE cur1 CURSOR FOR stmt1;


    /* #########SQLDA######## */
    sqlda2 = (sqlda_t *)malloc(sizeof(sqlda_t) + sizeof(sqlvar_t));
    memset(sqlda2, 0, sizeof(sqlda_t) + sizeof(sqlvar_t));

    sqlda2->sqln = 2; /* ###### */

    sqlda2->sqlvar[0].sqltype = ECPGt_char;
    sqlda2->sqlvar[0].sqldata = "postgres";
    sqlda2->sqlvar[0].sqllen  = 8;

    intval = 1;
    sqlda2->sqlvar[1].sqltype = ECPGt_int;
    sqlda2->sqlvar[1].sqldata = (char *) &intval;
    sqlda2->sqlvar[1].sqllen  = sizeof(intval);


    /* ################### */
    EXEC SQL OPEN cur1 USING DESCRIPTOR sqlda2;

    while (1)
    {
        sqlda_t *cur_sqlda;


        /* ############## */
```

```
        EXEC SQL FETCH NEXT FROM cur1 INTO DESCRIPTOR sqlda1;

        for (cur_sqlda = sqlda1 ;
             cur_sqlda != NULL ;
             cur_sqlda = cur_sqlda->desc_next)
        {
            int i;
            char name_buf[1024];
            char var_buf[1024];


            /* 1############ */
            for (i=0 ; i<cur_sqlda->sqld ; i++)
            {
                sqlvar_t v = cur_sqlda->sqlvar[i];
                char *sqldata = v.sqldata;
                short sqllen  = v.sqllen;

                strncpy(name_buf, v.sqlname.data, v.sqlname.length);
                name_buf[v.sqlname.length] = '\0';

                switch (v.sqltype)
                {
                    case ECPGt_char:
                        memset(&var_buf, 0, sizeof(var_buf));
                        memcpy(&var_buf, sqldata,
(sizeof(var_buf)<=sqllen ? sizeof(var_buf)-1 : sqllen) );
                        break;


                    case ECPGt_int: /* ## */
                        memcpy(&intval, sqldata, sqllen);
                        snprintf(var_buf, sizeof(var_buf), "%d", intval);
                        break;

                    case ECPGt_long_long: /* bigint */
                        memcpy(&longlongval, sqldata, sqllen);
                        snprintf(var_buf, sizeof(var_buf), "%lld",
longlongval);
                        break;

                    default:
                    {
                        int i;
                        memset(var_buf, 0, sizeof(var_buf));
                        for (i = 0; i < sqllen; i++)
                        {
                            char tmpbuf[16];
                            snprintf(tmpbuf, sizeof(tmpbuf), "%02x ",
(unsigned char) sqldata[i]);
                            strncat(var_buf, tmpbuf, sizeof(var_buf));
                        }
                    }
                        break;
                }

                printf("%s = %s (type: %d)\n", name_buf, var_buf,
v.sqltype);
            }

            printf("\n");
        }
    }
```

```
    EXEC SQL CLOSE cur1;
    EXEC SQL COMMIT;

    EXEC SQL DISCONNECT ALL;

    return 0;
}

####################################


oid = 1 (type: 1)
datname = template1 (type: 1)
datdba = 10 (type: 1)
encoding = 0 (type: 5)
datistemplate = t (type: 1)
datallowconn = t (type: 1)
dathasloginevt = f (type: 1)
datconnlimit = -1 (type: 5)
datfrozenxid = 379 (type: 1)
dattablespace = 1663 (type: 1)
datconfig =  (type: 1)
datacl = {=c/uptime,uptime=CTc/uptime} (type: 1)
datid = 1 (type: 1)
datname = template1 (type: 1)
numbackends = 0 (type: 5)
xact_commit = 113606 (type: 9)
xact_rollback = 0 (type: 9)
blks_read = 130 (type: 9)
blks_hit = 7341714 (type: 9)
tup_returned = 38262679 (type: 9)
tup_fetched = 1836281 (type: 9)
tup_inserted = 0 (type: 9)
tup_updated = 0 (type: 9)
tup_deleted = 0 (type: 9)

oid = 11511 (type: 1)
datname = postgres (type: 1)
datdba = 10 (type: 1)
encoding = 0 (type: 5)
datistemplate = f (type: 1)
datallowconn = t (type: 1)
dathasloginevt = f (type: 1)
datconnlimit = -1 (type: 5)
datfrozenxid = 379 (type: 1)
dattablespace = 1663 (type: 1)
datconfig =  (type: 1)
datacl =  (type: 1)
datid = 11511 (type: 1)
datname = postgres (type: 1)
numbackends = 0 (type: 5)
xact_commit = 221069 (type: 9)
xact_rollback = 18 (type: 9)
blks_read = 1176 (type: 9)
blks_hit = 13943750 (type: 9)
tup_returned = 77410091 (type: 9)
tup_fetched = 3253694 (type: 9)
tup_inserted = 0 (type: 9)
tup_updated = 0 (type: 9)
tup_deleted = 0 (type: 9)
```

34.8.

#########SQL### ########################

- WHENEVER##
- ################sqlca###############

34.8.1.

################1#################################### ################

```
EXEC SQL WHENEVER condition action;
```

condition##################

SQLERROR

SQL##############################

SQLWARNING

SQL#############################

NOT FOUND

SQL##0##########0#################### ################################### ###

action##################

CONTINUE

############################## ###########

GOTO *label*
GO TO *label*

############ #C#goto########

SQLPRINT

################# ############################## ################

STOP

###########exit(1)#######

DO BREAK

C#break####### ############switch#################

DO CONTINUE

C#continue####### ##################### ########################

CALL *name* (*args*)
DO *name* (*args*)

############C########## ##########PostgreSQL####CALL###DO############

##SQL##CONTINUE#GOTO##GO TO############

############################## ##

```
EXEC SQL WHENEVER SQLWARNING SQLPRINT;
```

```
EXEC SQL WHENEVER SQLERROR STOP;
```

EXEC SQL WHENEVER##C########SQL################## ####################EXEC SQL WHENEVER#########SQL## SQL######### C################### ########2##C##########################

```
/*

 * ###
 */
int main(int argc, char *argv[])
{
    ...
    if (verbose) {
        EXEC SQL WHENEVER SQLWARNING SQLPRINT;
    }
    ...
    EXEC SQL SELECT ...;
    ...
}
```

```
/*

 * ###
 */
int main(int argc, char *argv[])
{
    ...
    set_error_handler();
    ...
    EXEC SQL SELECT ...;
    ...
}

static void set_error_handler(void)
{
    EXEC SQL WHENEVER SQLERROR STOP;
}
```

34.8.2. sqlca

###################SQL################sqlca#SQL########################

```
struct
{
    char sqlcaid[8];
    long sqlabc;
    long sqlcode;
    struct
    {
        int sqlerrml;
        char sqlerrmc[SQLERRMC_LEN];
    } sqlerrm;
    char sqlerrp[8];
    long sqlerrd[6];
    char sqlwarn[8];
    char sqlstate[5];
} sqlca;
```

#############################sqlca############ #####C#errno################ ####

sqlca################## 1##SQL######################sqlca################
##

###SQL##########sqlca.sqlcode#0##sqlca.sqlstate#"00000"###### #########
#####sqlca.sqlcode#####sqlca.sqlstate#"00000"####### #
#sqlca.sqlcode##########0##################
sqlca.sqlcode#sqlca.sqlstate#2################ ##########

###SQL#######sqlca.sqlerrd[1]#######OID############## ###############
##sqlca.sqlerrd[2]#######################

############sqlca.sqlerrm.sqlerrmc######################
sqlca.sqlerrm.sqlerrml######sqlca.sqlerrm.sqlerrmc##################
#strlen()#########C################ ############sqlerrmc##############
#############

######sqlca.sqlwarn[2]#W###### ##############W##############
sqlca.sqlwarn[1]#W################################ ##################
##sqlca.sqlwarn[0]#W########

######sqlcaid#sqlabc#sqlerrp####sqlerrd#sqlwarn###################

sqlca###SQL########### #########SQL################## ####################
#####################################

###WHENEVER#sqlca####################sqlca################## #############
##

```
EXEC SQL WHENEVER SQLERROR CALL print_sqlca();

void
print_sqlca()
{
    fprintf(stderr, "==== sqlca ====\n");
    fprintf(stderr, "sqlcode: %ld\n", sqlca.sqlcode);
    fprintf(stderr, "sqlerrm.sqlerrml: %d\n", sqlca.sqlerrm.sqlerrml);
    fprintf(stderr, "sqlerrm.sqlerrmc: %s\n", sqlca.sqlerrm.sqlerrmc);
    fprintf(stderr, "sqlerrd: %ld %ld %ld %ld %ld %ld\n",
 sqlca.sqlerrd[0],sqlca.sqlerrd[1],sqlca.sqlerrd[2],

 sqlca.sqlerrd[3],sqlca.sqlerrd[4],sqlca.sqlerrd[5]);
    fprintf(stderr, "sqlwarn: %d %d %d %d %d %d %d %d\n",
 sqlca.sqlwarn[0], sqlca.sqlwarn[1], sqlca.sqlwarn[2],

 sqlca.sqlwarn[3], sqlca.sqlwarn[4], sqlca.sqlwarn[5],

 sqlca.sqlwarn[6], sqlca.sqlwarn[7]);
    fprintf(stderr, "sqlstate: %5s\n", sqlca.sqlstate);
    fprintf(stderr, "===============\n");
}
```

#######################################

```
==== sqlca ====
sqlcode: -400
sqlerrm.sqlerrml: 49
sqlerrm.sqlerrmc: relation "pg_databasep" does not exist on line 38
sqlerrd: 0 0 0 0 0 0
sqlwarn: 0 0 0 0 0 0 0 0
sqlstate: 42P01
===============
```

34.8.3. SQLSTATE#SQLCODE

sqlca.sqlstate#sqlca.sqlcode##############2####### ####SQL########## ##SQLCODE#SQL-92######################### ##################SQLSTATE######## #######

SQLSTATE#5######### ##5####################################### SQLSTATE## ######### ###2##################3################## #####00000############ SQLSTATE##########SQL######### PostgreSQL#######SQLSTATE################ # ### #########A##########

################SQLCODE######## 0####################################### # #SQL##########0########0##########+100################ ############# #### ###################################### #####PostgreSQL#####SQL############## #SQLCODE################ ######################## ######SQL################ ###### #########SQLSTATE###################SQLSTATE######## ####2###########1 #1####1#### ########## ###########A##########SQLSTATE##############

#########SQLCODE###

0 (ECPG_NO_ERROR)

###########(SQLSTATE 00000)

100 (ECPG_NOT_FOUND)

###
(SQLSTATE 02000)

###

```
while (1)
{
    EXEC SQL FETCH ... ;
    if (sqlca.sqlcode == ECPG_NOT_FOUND)
        break;
}
```

####WHENEVER NOT FOUND DO BREAK############### ######################## #####

-12 (ECPG_OUT_OF_MEMORY)

############ #####-ENOMEM######### #SQLSTATE YE001#

-200 (ECPG_UNSUPPORTED)

##################################### ######################################
#SQLSTATE YE002#

-201 (ECPG_TOO_MANY_ARGUMENTS)

################################ #SQLSTATE 07001####07002#

-202 (ECPG_TOO_FEW_ARGUMENTS)

################################ #SQLSTATE 07001####07002#

-203 (ECPG_TOO_MANY_MATCHES)

##################SQL###1############################# ################### ### #SQLSTATE 21000#

-204 (ECPG_INT_FORMAT)

########int#########################int################### ##########
#strtol()####### #SQLSTATE 42804#

-205 (ECPG_UINT_FORMAT)

########unsigned int#########################unsigned int###############
###########strtoul()####### #SQLSTATE 42804#

-206 (ECPG_FLOAT_FORMAT)

########float#########################float################### ##########
#strtod()####### #SQLSTATE 42804#

-207 (ECPG_NUMERIC_FORMAT)

########numeric#########################numeric###################
#SQLSTATE 42804#

-208 (ECPG_INTERVAL_FORMAT)

########interval#######################interval#####################
(SQLSTATE 42804)

-209 (ECPG_DATE_FORMAT)

########date######################date##################### (SQLSTATE
42804)

-210 (ECPG_TIMESTAMP_FORMAT)

########timestamp######################timestamp#####################
(SQLSTATE 42804)

-211 (ECPG_CONVERT_BOOL)

############bool###############'t'##'f'############### #SQLSTATE 42804#

-212 (ECPG_EMPTY)

PostgreSQL########SQL###### #######SQL#####################################
#SQLSTATE YE002#

-213 (ECPG_MISSING_INDICATOR)

NULL##########NULL#################### #SQLSTATE 22002#

-214 (ECPG_NO_ARRAY)

######################### #SQLSTATE 42804#

-215 (ECPG_DATA_NOT_ARRAY)

############################## #SQLSTATE 42804#

-216 (ECPG_ARRAY_INSERT)

############# (SQLSTATE 42804)

-220 (ECPG_NO_CONN)

########################### #SQLSTATE 08003#

-221 (`ECPG_NOT_CONN`)

################################ ############# #SQLSTATE YE002#

-230 (`ECPG_INVALID_STMT`)

########SQL################ #SQLSTATE 26000#

-239 (`ECPG_INFORMIX_DUPLICATE_KEY`)

################Informix####### (SQLSTATE 23505)

-240 (`ECPG_UNKNOWN_DESCRIPTOR`)

################## ########SQL################# #SQLSTATE 33000#

-241 (`ECPG_INVALID_DESCRIPTOR_INDEX`)

################# #SQLSTATE 07009#

-242 (`ECPG_UNKNOWN_DESCRIPTOR_ITEM`)

############################## #SQLSTATE YE002#

-243 (`ECPG_VAR_NOT_NUMERIC`)

###SQL## #SQLSTATE 07006#

-244 (`ECPG_VAR_NOT_CHAR`)

###SQL################################### #SQLSTATE 07006#

-284 (`ECPG_INFORMIX_SUBSELECT_NOT_ONE`)

####################Informix####### (SQLSTATE 21000)

-400 (`ECPG_PGSQL`)

PostgreSQL################## ########PostgreSQL####################

-401 (`ECPG_TRANS`)

PostgreSQL## #SQLSTATE 08007#

-402 (`ECPG_CONNECT`)

################### (SQLSTATE 08001)

-403 (`ECPG_DUPLICATE_KEY`)

################ (SQLSTATE 23505)

-404 (`ECPG_SUBSELECT_NOT_ONE`)

#################### (SQLSTATE 21000)

-602 (`ECPG_WARNING_UNKNOWN_PORTAL`)

################# (SQLSTATE 34000)

-603 (`ECPG_WARNING_IN_TRANSACTION`)

############### (SQLSTATE 25001)

-604 (`ECPG_WARNING_NO_TRANSACTION`)

######################## (SQLSTATE 25P01)

-605 (ECPG_WARNING_PORTAL_EXISTS)

################ (SQLSTATE 42P03)

34.9.

ecpg###

34.9.1.

####SQL################################

```
EXEC SQL INCLUDE filename;
EXEC SQL INCLUDE <filename>;
EXEC SQL INCLUDE "filename";
```

####SQL#########*filename*.h##########################C######### ############ ########SQL###########

ecpg##################################

- #########
- /usr/local/include
- #########PostgreSQL############ (####/usr/local/pgsql/include)
- /usr/include

###EXEC SQL INCLUDE "*filename*"#########################

#################################### ## #.h###########

EXEC SQL INCLUDE###################

```
#include <filename.h>
```

########SQL################### #######################C##include############ #######

> ##
>
> ###SQL###############EXEC SQL INCLUDE################################ ###########

34.9.2. define###undef###

C#####define###########SQL###############

```
EXEC SQL DEFINE name;
EXEC SQL DEFINE name value;
```

#########################

```
EXEC SQL DEFINE HAVE_FEATURE;
```

##################

```
EXEC SQL DEFINE MYNUMBER 12;
EXEC SQL DEFINE MYSTRING 'abc';
```

###########undef#######

```
EXEC SQL UNDEF MYNUMBER;
```

###C##define##undef#####SQL################## ####################EXEC SQL DEFINE#######ecpg########################## ####

```
EXEC SQL DEFINE MYNUMBER 12;
...
EXEC SQL UPDATE Tbl SET col = MYNUMBER;
```

#######ecpg##################C#######MYNUMBER##################### ####SQL################define#################### #########SQL############## #############

ecpg##################################EXEC SQL DEFINE###EXEC SQL UNDEF## ################ #############-D#####################

34.9.3. ifdef#ifndef#elif#else#endif###

#################################

EXEC SQL ifdef *name*;

name#######*name*#EXEC SQL define *name*####################

EXEC SQL ifndef *name*;

name#######*name*#EXEC SQL define *name*####################

EXEC SQL elif *name*;

EXEC SQL ifdef *name*###EXEC SQL ifndef *name*############## ########## elif#################### elif######*name*#############ifdef/ifndef…endif############################

EXEC SQL else;

EXEC SQL ifdef *name*###EXEC SQL ifndef *name*#################### ##ifdef/ifndef…endif################################

EXEC SQL endif;

ifdef/ifndef…endif######## ##############

ifdef/ifndef…endif###127############

####3##SET TIMEZONE##########1##########

```
EXEC SQL ifdef TZVAR;
EXEC SQL SET TIMEZONE TO TZVAR;
EXEC SQL elif TZNAME;
EXEC SQL SET TIMEZONE TO TZNAME;
EXEC SQL else;
EXEC SQL SET TIMEZONE TO 'GMT';
EXEC SQL endif;
```

34.10. ####SQL########

##########SQL C#################### ######################## ################ ###SQL C############ #######SQL################# ########################### ########## ##################libpq######SQL##########################

############`ecpg`########PostgreSQL################## ######SQL######## `#.pgc`##### `prog1.pgc`##

```
ecpg prog1.pgc
```

`###prog1.c`################ ##########################`-o` ##################### ##########

###################### #########

```
cc -c prog1.c
```

#####C########PostgreSQL############################## ################### PostgreSQL##########################`-I/usr/local/pgsql/include`########## ###########

####SQL######################`libecpg`################

```
cc -o myprog prog1.o prog2.o ... -lecpg
```

`##################-L/usr/local/pgsql/lib`#########################

#######################`libecpg#pg_config###pkg-config`##########

##############make####################Makefile###########

```
ECPG = ecpg

%.c: %.pgc
        $(ECPG) $<
```

`ecpg`##########ecpg########

#######ecpg########## ##
##

34.11.

`libecpg`###############SQL## ########## ############### ###########################

- `ECPGdebug(int on, FILE *stream)`##1###0###################### ###### *`#stream`*######## #######################SQL###PostgreSQL############### SQL ####################

> ##
>
> Windows###ecpg##`FILE`######### ############################## ## ##multithreaded/single-threaded#release/debug####static/dynamic################

- ECPGget_PGconn(const char *connection_name)######################### ############ connection_name####NULL################## ################# #NULL###### #######################libpq#################

  ##

  libpq########ecpg############################

- ECPGtransactionStatus(const char *connection_name)##connection_name############################ ###### ##########32.2#libpq#PQtransactionStatus#########

- ECPGstatus(int lineno, const char* connection_name)################ ############## ##############connection_name#NULL##########

34.12.

#########ECPG############## ###ECPG#########ECPGget_PGconn()######### #PGconn######libpq################################## ####ECPGget_PGconn()# #####PGconn################################ECPG########################### ####

ECPGget_PGconn()#####34.11######### #######################33##########

#################################### ################BEGIN################ #####

34.2###ECPG##

#34.2 ################ECPG#####

```
#include <stdio.h>
#include <stdlib.h>
#include <libpq-fe.h>
#include <libpq/libpq-fs.h>

EXEC SQL WHENEVER SQLERROR STOP;

int
main(void)
{
    PGconn     *conn;
    Oid         loid;
    int         fd;
    char        buf[256];
    int         buflen = 256;
    char        buf2[256];
    int         rc;

    memset(buf, 1, buflen);

    EXEC SQL CONNECT TO testdb AS con1;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;

    conn = ECPGget_PGconn("con1");
    printf("conn = %p\n", conn);
```

```
    /* ## */
    loid = lo_create(conn, 0);
    if (loid &lt; 0)
        printf("lo_create() failed: %s", PQerrorMessage(conn));

    printf("loid = %d\n", loid);


    /* ####### */
    fd = lo_open(conn, loid, INV_READ|INV_WRITE);
    if (fd &lt; 0)
        printf("lo_open() failed: %s", PQerrorMessage(conn));

    printf("fd = %d\n", fd);

    rc = lo_write(conn, fd, buf, buflen);
    if (rc &lt; 0)
        printf("lo_write() failed\n");

    rc = lo_close(conn, fd);
    if (rc &lt; 0)
        printf("lo_close() failed: %s", PQerrorMessage(conn));


    /* ####### */
    fd = lo_open(conn, loid, INV_READ);
    if (fd &lt; 0)
        printf("lo_open() failed: %s", PQerrorMessage(conn));

    printf("fd = %d\n", fd);

    rc = lo_read(conn, fd, buf2, buflen);
    if (rc &lt; 0)
        printf("lo_read() failed\n");

    rc = lo_close(conn, fd);
    if (rc &lt; 0)
        printf("lo_close() failed: %s", PQerrorMessage(conn));


    /* ## */
    rc = memcmp(buf, buf2, buflen);
    printf("memcmp() = %d\n", rc);


    /* ### */
    rc = lo_unlink(conn, loid);
    if (rc &lt; 0)
        printf("lo_unlink() failed: %s", PQerrorMessage(conn));

    EXEC SQL COMMIT;
    EXEC SQL DISCONNECT ALL;
    return 0;
}
```

34.13. C++#########

ECPG#C++########################## #################

ecpg#######C####C###########SQL#######################SQL#####C##########
###.c########## ecpg#####C############################C++#######extern
"C" { ... }###### ####C++###############

########ecpg#######C########## C++#################### #####C++############
###C++#######################SQL##

C++############SQL##############ECPG#####C#############C++##############
###C++##############################C############ 34.13.2#########

34.13.1.

ecpg#######C################### C## #
##C++## ################################ ####ecpg#############################
##

##########ecpg#######test######dbname################## #############

```
class TestCpp
{
    EXEC SQL BEGIN DECLARE SECTION;
    char dbname[1024];
    EXEC SQL END DECLARE SECTION;

  public:
    TestCpp();
    void test();
    ~TestCpp();
};

TestCpp::TestCpp()
{
    EXEC SQL CONNECT TO testdb1;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
}

void Test::test()
{
    EXEC SQL SELECT current_database() INTO :dbname;
    printf("current_database = %s\n", dbname);
}

TestCpp::~TestCpp()
{
    EXEC SQL DISCONNECT ALL;
}
```

####################

```
ecpg test_cpp.pgc
test_cpp.pgc:28: ERROR: variable "dbname" is not declared
```

##################test## #################
########################

```
void TestCpp::test()
{
    EXEC SQL BEGIN DECLARE SECTION;
    char tmp[1024];
    EXEC SQL END DECLARE SECTION;

    EXEC SQL SELECT current_database() INTO :tmp;
    strlcpy(dbname, tmp, sizeof(tmp));
```

```
    printf("current_database = %s\n", dbname);
}
```

34.13.2. ###C#########C++###########

C++####ecpg##################ECPG######C++######################C######
C++##############C++########SQL################################ ###########
##C++################SQL#################### ##############C++###
##PostgreSQL##########C###ECPG######

C#####*.pgc#########C++##########################

test_mod.pgc

C#################### #########test_mod.c#######

```
#include "test_mod.h"
#include <stdio.h>

void
db_connect()
{
    EXEC SQL CONNECT TO testdb1;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false);
 EXEC SQL COMMIT;
}

void
db_test()
{
    EXEC SQL BEGIN DECLARE SECTION;
    char dbname[1024];
    EXEC SQL END DECLARE SECTION;

    EXEC SQL SELECT current_database() INTO :dbname;
    printf("current_database = %s\n", dbname);
}

void
db_disconnect()
{
    EXEC SQL DISCONNECT ALL;
}
```

test_mod.h

C#####test_mod.pgc################## test_cpp.cpp########### ######
##C++####################extern "C"################

```
#ifdef __cplusplus
extern "C" {
#endif

void db_connect();
void db_test();
void db_disconnect();

#ifdef __cplusplus
}
#endif
```

test_cpp.cpp

main#########C++#######################

```
#include "test_mod.h"

class TestCpp
{
  public:
    TestCpp();
    void test();
    ~TestCpp();
};

TestCpp::TestCpp()
{
    db_connect();
}

void
TestCpp::test()
{
    db_test();
}

TestCpp::~TestCpp()
{
    db_disconnect();
}

int
main(void)
{
    TestCpp *t = new TestCpp();

    t->test();
    return 0;
}
```

########################### ecpg####test_mod.pgc#test_mod.c###### ###C## ######test_mod.c######test_mod.o######

```
ecpg -o test_mod.c test_mod.pgc
cc -c test_mod.c -o test_mod.o
```

##C++########test_cpp.cpp######test_cpp.o######

```
c++ -c test_cpp.cpp -o test_cpp.o
```

####C++############test_cpp.o###test_mod.o############################

```
c++ test_cpp.o test_mod.o -lecpg -o test_cpp
```

34.14. ####SQL####

########SQL###SQL############## ##############SQL############SQL######### SQL##############

ALLOCATE DESCRIPTOR

ALLOCATE DESCRIPTOR — SQL############

##

```
ALLOCATE DESCRIPTOR name
```

##

ALLOCATE DESCRIPTOR##PostgreSQL##SQL############

############DEALLOCATE DESCRIPTOR#####################

#####

name

SQL######## ############ ###SQL#####################

#

```
EXEC SQL ALLOCATE DESCRIPTOR mydesc;
```

###

ALLOCATE DESCRIPTOR###SQL#########

####

DEALLOCATE DESCRIPTOR, GET DESCRIPTOR, SET DESCRIPTOR

CONNECT

CONNECT — ###############

##

```
CONNECT TO connection_target [ AS connection_name ] [ USER connection_user
 ]
CONNECT TO DEFAULT
CONNECT connection_user
DATABASE connection_target
```

##

CONNECT###########PostgreSQL###############

#####

connection_target

connection_target ################################

[*database_name*][@*host*][:*port*]

TCP/IP#######

unix:postgresql:*//host* [:*port*] / [*database_name*] [?*connection_option*]

Unix##############

tcp:postgresql:*//host*[:*port*]/[*database_name*][?*connection_option*]

TCP/IP#######

SQL#####

#####################

#####

##################char[]###VARCHAR[]#######

connection_name

###SQL####################

connection_user

################

########*user_name/password#user_name* IDENTIFIED BY *password#user_name* USING *password*##################################

############SQL#########################

DEFAULT

libpq############################

# #

```
##########################


EXEC SQL CONNECT TO "connectdb" AS main;
EXEC SQL CONNECT TO "connectdb" AS second;
EXEC SQL CONNECT TO "unix:postgresql://200.46.204.71/connectdb" AS main
 USER connectuser;
EXEC SQL CONNECT TO "unix:postgresql://localhost/connectdb" AS main USER
 connectuser;
EXEC SQL CONNECT TO 'connectdb' AS main;
EXEC SQL CONNECT TO 'unix:postgresql://localhost/connectdb' AS main
 USER :user;
EXEC SQL CONNECT TO :db AS :id;
EXEC SQL CONNECT TO :db USER connectuser USING :pw;
EXEC SQL CONNECT TO @localhost AS main USER connectdb;
EXEC SQL CONNECT TO REGRESSDB1 as main;
EXEC SQL CONNECT TO AS main USER connectdb;
EXEC SQL CONNECT TO connectdb AS :id;
EXEC SQL CONNECT TO connectdb AS main USER connectuser/connectdb;
EXEC SQL CONNECT TO connectdb AS main;
EXEC SQL CONNECT TO connectdb@localhost AS main;
EXEC SQL CONNECT TO tcp:postgresql://localhost/ USER connectdb;
EXEC SQL CONNECT TO tcp:postgresql://localhost/connectdb USER connectuser
 IDENTIFIED BY connectpw;
EXEC SQL CONNECT TO tcp:postgresql://localhost:20/connectdb USER
 connectuser IDENTIFIED BY connectpw;
EXEC SQL CONNECT TO unix:postgresql://localhost/ AS main USER connectdb;
EXEC SQL CONNECT TO unix:postgresql://localhost/connectdb AS main USER
 connectuser;
EXEC SQL CONNECT TO unix:postgresql://localhost/connectdb USER connectuser
 IDENTIFIED BY "connectpw";
EXEC SQL CONNECT TO unix:postgresql://localhost/connectdb USER connectuser
 USING "connectpw";
EXEC SQL CONNECT TO unix:postgresql://localhost/connectdb?
connect_timeout=14 USER connectuser;

##########################################


int
main(void)
{
EXEC SQL BEGIN DECLARE SECTION;

    char *dbname     = "testdb";    /* ####### */
    char *user       = "testuser";  /* ###### */
    char *connection = "tcp:postgresql://localhost:5432/testdb";
                                    /* ##### */
    char ver[256];                  /* ################# */
EXEC SQL END DECLARE SECTION;

    ECPGdebug(1, stderr);

    EXEC SQL CONNECT TO :dbname USER :user;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
    EXEC SQL SELECT version() INTO :ver;
    EXEC SQL DISCONNECT;

    printf("version: %s\n", ver);

    EXEC SQL CONNECT TO :connection USER :user;
```

```
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
    EXEC SQL SELECT version() INTO :ver;
    EXEC SQL DISCONNECT;

    printf("version: %s\n", ver);

    return 0;
}
```

###

CONNECT###SQL################################

####

DISCONNECT, SET CONNECTION

DEALLOCATE DESCRIPTOR

DEALLOCATE DESCRIPTOR — SQL################

##

```
DEALLOCATE DESCRIPTOR name
```

##

DEALLOCATE DESCRIPTOR#####SQL################

#####

name

################# ############ ###SQL####################

#

```
EXEC SQL DEALLOCATE DESCRIPTOR mydesc;
```

###

DEALLOCATE DESCRIPTOR###SQL#########

####

ALLOCATE DESCRIPTOR, GET DESCRIPTOR, SET DESCRIPTOR

DECLARE

DECLARE — ###########

##

```
DECLARE cursor_name [ BINARY ] [ ASENSITIVE | INSENSITIVE ] [ [ NO ]
 SCROLL ] CURSOR [ { WITH | WITHOUT } HOLD ] FOR prepared_name
DECLARE cursor_name [ BINARY ] [ ASENSITIVE | INSENSITIVE ] [ [ NO ]
 SCROLL ] CURSOR [ { WITH | WITHOUT } HOLD ] FOR query
```

##

DECLARE################################## ###########DECLARESQL############ ####### ####################################SQL############################## ############## ######OPEN####################

#####

cursor_name

######## ############# ###SQL#####################

prepared_name

################ SQL##################

query

################SELECT###VALUES#######

################DECLARE##########

#

#########################

```
EXEC SQL DECLARE C CURSOR FOR SELECT * FROM My_Table;
EXEC SQL DECLARE C CURSOR FOR SELECT Item1 FROM T;
EXEC SQL DECLARE cur1 CURSOR FOR SELECT version();
```

#######################

```
EXEC SQL PREPARE stmt1 AS SELECT version();
EXEC SQL DECLARE cur1 CURSOR FOR stmt1;
```

###

DECLARE###SQL#########

####

OPEN, CLOSE, DECLARE

DECLARE STATEMENT

DECLARE STATEMENT — SQL########

##

```
EXEC SQL [ AT connection_name ] DECLARE statement_name STATEMENT
```

##

DECLARE STATEMENT#SQL########## SQL#################### ######SQL##### ##################### #####################SQL##################### ######## Informix#############SQL#############"database"#########################

#####

connection_name

CONNECT#####################

AT#########################

statement_name

SQL########SQL#################

##

```
################################
```

#

```
EXEC SQL CONNECT TO postgres AS con1;
EXEC SQL AT con1 DECLARE sql_stmt STATEMENT;
EXEC SQL DECLARE cursor_name CURSOR FOR sql_stmt;
EXEC SQL PREPARE sql_stmt FROM :dyn_string;
EXEC SQL OPEN cursor_name;
EXEC SQL FETCH cursor_name INTO :column1;
EXEC SQL CLOSE cursor_name;
```

###

DECLARE STATEMENT###SQL##########DBMS########

####

CONNECT, DECLARE, OPEN

DESCRIBE

DESCRIBE — ##########################

##

```
DESCRIBE [ OUTPUT ] prepared_name USING [ SQL ] DESCRIPTOR descriptor_name
DESCRIBE [ OUTPUT ] prepared_name INTO [ SQL ] DESCRIPTOR descriptor_name
DESCRIBE [ OUTPUT ] prepared_name INTO sqlda_name
```

##

```
DESCRIBE##################################################
```

#####

prepared_name

############ ###SQL#####################

descriptor_name

######### ############# ###SQL#####################

sqlda_name

SQLDA########

#

```
EXEC SQL ALLOCATE DESCRIPTOR mydesc;
EXEC SQL PREPARE stmt1 FROM :sql_stmt;
EXEC SQL DESCRIBE stmt1 INTO SQL DESCRIPTOR mydesc;
EXEC SQL GET DESCRIPTOR mydesc VALUE 1 :charvar = NAME;
EXEC SQL DEALLOCATE DESCRIPTOR mydesc;
```

###

```
DESCRIBE###SQL##########
```

####

ALLOCATE DESCRIPTOR, GET DESCRIPTOR

DISCONNECT

DISCONNECT — ##############

##

```
DISCONNECT connection_name
DISCONNECT [ CURRENT ]
DISCONNECT ALL
```

##

DISCONNECT###########################

#####

connection_name

CONNECT######################

CURRENT

###########SET CONNECTION################################## ##
#DISCONNECT######################

ALL

################

#

```
int
main(void)
{
    EXEC SQL CONNECT TO testdb AS con1 USER testuser;
    EXEC SQL CONNECT TO testdb AS con2 USER testuser;
    EXEC SQL CONNECT TO testdb AS con3 USER testuser;


    EXEC SQL DISCONNECT CURRENT;  /* con3####        */
    EXEC SQL DISCONNECT ALL;      /* con2#con1####  */

    return 0;
}
```

###

DISCONNECT###SQL#########

####

CONNECT, SET CONNECTION

EXECUTE IMMEDIATE

EXECUTE IMMEDIATE — SQL################

##

```
EXECUTE IMMEDIATE string
```

##

EXECUTE IMMEDIATE########SQL############################

#####

string

###############SQL###########

##

#########*string*########SQL################## ####################
##EXECUTE IMMEDIATE############SQL################

###################SQL##############C########\"#####8######(\042)##########
################ #######EXEC SQL###########ECPG######C#######SQL#########
################C################\"################################

#

```
####EXECUTE IMMEDIATE#command#########INSERT###########

sprintf(command, "INSERT INTO test (name, amount, letter) VALUES ('db:
 ''r1''', 1, 'f')");
EXEC SQL EXECUTE IMMEDIATE :command;
```

###

EXECUTE IMMEDIATE###SQL#########

GET DESCRIPTOR

GET DESCRIPTOR — SQL###############

##

```
GET DESCRIPTOR descriptor_name :cvariable = descriptor_header_item
 [, ... ]
GET DESCRIPTOR descriptor_name VALUE column_number :cvariable
 = descriptor_item [, ... ]
```

##

GET DESCRIPTOR#SQL## ##############
#######################FETCH###SELECT############

################## ## ######### #####
###############

#####

descriptor_name

#########

descriptor_header_item

######################## #############COUNT###############

column_number

############# ########

descriptor_item

########################## ##################34.7.1#########

cvariable

############################

#

####################

```
EXEC SQL GET DESCRIPTOR d :d_count = COUNT;
```

####################

```
EXEC SQL GET DESCRIPTOR d VALUE 1 :d_returned_octet_length =
 RETURNED_OCTET_LENGTH;
```

##############################

```
EXEC SQL GET DESCRIPTOR d VALUE 2 :d_data = DATA;
```

####SELECT current_database();##################################

```
int
main(void)
{
EXEC SQL BEGIN DECLARE SECTION;
    int  d_count;
    char d_data[1024];
    int  d_returned_octet_length;
EXEC SQL END DECLARE SECTION;

    EXEC SQL CONNECT TO testdb AS con1 USER testuser;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
    EXEC SQL ALLOCATE DESCRIPTOR d;


    /* ############################## */
    EXEC SQL DECLARE cur CURSOR FOR SELECT current_database();
    EXEC SQL OPEN cur;
    EXEC SQL FETCH NEXT FROM cur INTO SQL DESCRIPTOR d;


    /* ###### */
    EXEC SQL GET DESCRIPTOR d :d_count = COUNT;
    printf("d_count                 = %d\n", d_count);


    /* ########### */
    EXEC SQL GET DESCRIPTOR d VALUE 1 :d_returned_octet_length =
 RETURNED_OCTET_LENGTH;
    printf("d_returned_octet_length = %d\n", d_returned_octet_length);


    /* ################ */
    EXEC SQL GET DESCRIPTOR d VALUE 1 :d_data = DATA;
    printf("d_data                  = %s\n", d_data);


    /* ### */
    EXEC SQL CLOSE cur;
    EXEC SQL COMMIT;

    EXEC SQL DEALLOCATE DESCRIPTOR d;
    EXEC SQL DISCONNECT ALL;

    return 0;
}
```

```
########################


d_count                 = 1
d_returned_octet_length = 6
d_data                  = testdb
```

###

GET DESCRIPTOR###SQL#########

####

ALLOCATE DESCRIPTOR, SET DESCRIPTOR

OPEN

OPEN — ###########

##

```
OPEN cursor_name
OPEN cursor_name USING value [, ... ]
OPEN cursor_name USING SQL DESCRIPTOR descriptor_name
```

##

OPEN## ########DECLARE######## ############### OPEN##########################

#####

cursor_name

########### ###SQL####################

value

######################### ####SQL##############################

descriptor_name

################################## ###SQL#####################

#

```
EXEC SQL OPEN a;
EXEC SQL OPEN d USING 1, 'test';
EXEC SQL OPEN c1 USING SQL DESCRIPTOR mydesc;
EXEC SQL OPEN :curname1;
```

###

OPEN###SQL#########

####

DECLARE, CLOSE

PREPARE

PREPARE — ######SQL##########

##

```
PREPARE prepared_name FROM string
```

##

PREPARE##################SQL######### ##############################
#PREPARE SQL######## EXECUTE####################################

#####

prepared_name

#################

string

##################SQL####SELECT/INSERT/UPDATE/DELETE#1###############
###################(?)########

##

##########*string*########SQL################## ###########################
SQL PREPARE##############

###################SQL##############C########\"#####8######(\042)###########
################# #######EXEC SQL###########ECPG######C#######SQL##########
################C################\"##################################

#

```
char *stmt = "SELECT * FROM test1 WHERE a = ? AND b = ?";

EXEC SQL ALLOCATE DESCRIPTOR outdesc;
EXEC SQL PREPARE foo FROM :stmt;

EXEC SQL EXECUTE foo USING SQL DESCRIPTOR indesc INTO SQL DESCRIPTOR
 outdesc;
```

###

PREPARE###SQL##########

####

EXECUTE

SET AUTOCOMMIT

SET AUTOCOMMIT — ########################

##

```
SET AUTOCOMMIT { = | TO } { ON | OFF }
```

##

SET AUTOCOMMIT############################## ###########SQL################## ##### ####COMMIT############################## ################SQL############## ##########################

###

SET AUTOCOMMIT#PostgreSQL ECPG######

SET CONNECTION

SET CONNECTION — ##############

##

```
SET CONNECTION [ TO | = ] connection_name
```

##

SET CONNECTION###

#####

connection_name

CONNECT######################

CURRENT

##############################

#

```
EXEC SQL SET CONNECTION TO con2;
EXEC SQL SET CONNECTION = con1;
```

###

SET CONNECTION###SQL##########

####

CONNECT, DISCONNECT

SET DESCRIPTOR

SET DESCRIPTOR — SQL##############

##

```
SET DESCRIPTOR descriptor_name descriptor_header_item = value [, ... ]
SET DESCRIPTOR descriptor_name VALUE number descriptor_item = value
 [, ...]
```

##

SET DESCRIPTOR#SQL############# ## #######

################# ################################ ######################### ######

#####

descriptor_name

#########

descriptor_header_item

###################### ##########COUNT###############

number

############## ##########

descriptor_item

############################## ##################34.7.1##########

value

############## ###SQL####################

#

```
EXEC SQL SET DESCRIPTOR indesc COUNT = 1;
EXEC SQL SET DESCRIPTOR indesc VALUE 1 DATA = 2;
EXEC SQL SET DESCRIPTOR indesc VALUE 1 DATA = :val1;
EXEC SQL SET DESCRIPTOR indesc VALUE 2 INDICATOR = :val1, DATA = 'some
 string';
EXEC SQL SET DESCRIPTOR indesc VALUE 2 INDICATOR = :val2null, DATA
 = :val2;
```

###

SET DESCRIPTOR###SQL#########

####

ALLOCATE DESCRIPTOR, GET DESCRIPTOR

TYPE

TYPE — ##############

##

```
TYPE type_name IS ctype
```

##

TYPE#######C######## ##########typedef############

ecpg#-c##############################

#####

type_name

########## ######C##############

ctype

C#######

#

```
EXEC SQL TYPE customer IS
    struct
    {
        varchar name[50];
        int     phone;
    };

EXEC SQL TYPE cust_ind IS
    struct ind
    {
        short   name_ind;
        short   phone_ind;
    };

EXEC SQL TYPE c IS char reference;
EXEC SQL TYPE ind IS union { int integer; short smallint; };
EXEC SQL TYPE intarray IS int[AMOUNT];
EXEC SQL TYPE str IS varchar[BUFFERSIZ];
EXEC SQL TYPE string IS char[11];
```

###EXEC SQL TYPE################

```
EXEC SQL WHENEVER SQLERROR SQLPRINT;

EXEC SQL TYPE tt IS
    struct
    {
        varchar v[256];
        int     i;
    };

EXEC SQL TYPE tt_ind IS
    struct ind {
```

```
        short   v_ind;
        short   i_ind;
    };

int
main(void)
{
EXEC SQL BEGIN DECLARE SECTION;
    tt t;
    tt_ind t_ind;
EXEC SQL END DECLARE SECTION;

    EXEC SQL CONNECT TO testdb AS con1;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;

    EXEC SQL SELECT current_database(), 256 INTO :t:t_ind LIMIT 1;

    printf("t.v = %s\n", t.v.arr);
    printf("t.i = %d\n", t.i);

    printf("t_ind.v_ind = %d\n", t_ind.v_ind);
    printf("t_ind.i_ind = %d\n", t_ind.i_ind);

    EXEC SQL DISCONNECT con1;

    return 0;
}

######################


t.v = testdb
t.i = 256
t_ind.v_ind = 0
t_ind.i_ind = 0
```

###

TYPE#####PostgreSQL######

VAR

VAR — ########

##

```
VAR varname IS ctype
```

##

VAR########C################## ##################################

#####

varname

C#######

ctype

C#######

#

```
Exec sql begin declare section;
short a;
exec sql end declare section;
EXEC SQL VAR a IS int;
```

###

VAR#####PostgreSQL######

WHENEVER

WHENEVER — SQL##############################

##

```
WHENEVER { NOT FOUND | SQLERROR | SQLWARNING } action
```

##

SQL####################SQL#####SQL####################

#####

#############34.8.1##########

#

```
EXEC SQL WHENEVER NOT FOUND CONTINUE;
EXEC SQL WHENEVER NOT FOUND DO BREAK;
EXEC SQL WHENEVER NOT FOUND DO CONTINUE;
EXEC SQL WHENEVER SQLWARNING SQLPRINT;
EXEC SQL WHENEVER SQLWARNING DO warn();
EXEC SQL WHENEVER SQLERROR sqlprint;
EXEC SQL WHENEVER SQLERROR CALL print2();
EXEC SQL WHENEVER SQLERROR DO handle_error("select");
EXEC SQL WHENEVER SQLERROR DO sqlnotice(NULL, NONO);
EXEC SQL WHENEVER SQLERROR DO sqlprint();
EXEC SQL WHENEVER SQLERROR GOTO error_label;
EXEC SQL WHENEVER SQLERROR STOP;
```

```
#######################WHENEVER NOT FOUND BREAK####################


int
main(void)
{
    EXEC SQL CONNECT TO testdb AS con1;
    EXEC SQL SELECT pg_catalog.set_config('search_path', '', false); EXEC
 SQL COMMIT;
    EXEC SQL ALLOCATE DESCRIPTOR d;
    EXEC SQL DECLARE cur CURSOR FOR SELECT current_database(), 'hoge',
 256;
    EXEC SQL OPEN cur;


    /* ##############while######## */
    EXEC SQL WHENEVER NOT FOUND DO BREAK;

    while (1)
    {
        EXEC SQL FETCH NEXT FROM cur INTO SQL DESCRIPTOR d;
        ...
    }

    EXEC SQL CLOSE cur;
    EXEC SQL COMMIT;

    EXEC SQL DEALLOCATE DESCRIPTOR d;
    EXEC SQL DISCONNECT ALL;
```

```
    return 0;
}
```

###

WHENEVER###SQL##################PostgreSQL######

34.15. Informix#####

ecpg#*Informix*######################### ###########Informix E/SQL##Informix######## ############ ################SQL##########EXEC SQL########################## ##

```
$int j = 3;
$CONNECT TO :dbname;
$CREATE TABLE test(i INT PRIMARY KEY, j INT);
$INSERT INTO test(i, j) VALUES (7, :j);
$COMMIT;
```

> ##
>
> $######include#define#ifdef############################## ########## ######################

INFORMIX#INFORMIX_SE###2############

######################ECPG#####libcompat#################

################Informix####################E/SQL##ECPG########SQL########## ###########

Informix#####ECPG#pgtypeslib################ pgtypeslib#SQL#####C############## ##########Informix####################C#################### ############## ## ###Informix############## ####################################Informix##### ##################### ############### ####PostgreSQL#####interval###YEAR TO MINUTE##################ECPG#################

34.15.1.

##################Informix####"string"####typedef######Informix############### ### ##Informix#####ECPG#typedef sometype string;##################

```
EXEC SQL BEGIN DECLARE SECTION;

string userid; /* ######################## */
EXEC SQL END DECLARE SECTION;

EXEC SQL FETCH MYCUR INTO :userid;
```

34.15.2. ##############SQL#

CLOSE DATABASE

##SQL############ #####ECPG#DISCONNECT CURRENT#####

```
$CLOSE DATABASE;                /* ######### */
EXEC SQL CLOSE DATABASE;
```

FREE cursor_name

Informix#ESQL/C###ECPG#############(#######################################
#####)###ECPG##FREE cursor_name######## ####ECPG#####DECLARE CURSOR#
#################################### ###ECPG###########SQL################
##PostgreSQL#####################

FREE statement_name

FREE statement_name#DEALLOCATE PREPARE statement_name#######

34.15.3. Informix##SQLDA#####

Informix######34.7.2################### ############

```
struct sqlvar_compat
{
    short   sqltype;
    int     sqllen;
    char   *sqldata;
    short  *sqlind;
    char   *sqlname;
    char   *sqlformat;
    short   sqlitype;
    short   sqlilen;
    char   *sqlidata;
    int     sqlxid;
    char   *sqltypename;
    short   sqltypelen;
    short   sqlownerlen;
    short   sqlsourcetype;
    char   *sqlownername;
    int     sqlsourceid;
    char   *sqlilongdata;
    int     sqlflags;
    void   *sqlreserved;
};

struct sqlda_compat
{
    short  sqld;
    struct sqlvar_compat *sqlvar;
    char   desc_name[19];
    short  desc_occ;
    struct sqlda_compat *desc_next;
    void  *reserved;
};

typedef struct sqlvar_compat    sqlvar_t;
typedef struct sqlda_compat     sqlda_t;
```

###############

sqld

SQLDA##############

sqlvar

###################

desc_name

############

desc_occ

################

desc_next

######################SQLDA###########

reserved

#########NULL####### Informix############

################# ####sqlvar##########

sqltype

########## ###sqltypes.h##########

sqllen

############

sqldata

################ #######char *#### ################## ##########

```
int intval;

switch (sqldata->sqlvar[i].sqltype)
{
    case SQLINTEGER:
        intval = *(int *)sqldata->sqlvar[i].sqldata;
        break;
  ...
}
```

sqlind

NULL########### DESCRIBE###FETCH################### EXECUTE ... USING sqlda;###############NULL################NULL########### ####### #######sqlitype################ ##########

```
if (*(int2 *)sqldata->sqlvar[i].sqlind != 0)
    printf("value is NULL\n");
```

sqlname

######### ###########

sqlformat

Informix########### ########PQfformat#####

sqlitype

NULL########### ##################SQLSMINT### #############SQLDA########## #################

sqlilen

NULL###########

sqlxid

###########PQftype######

sqltypename
sqltypelen
sqlownerlen
sqlsourcetype
sqlownername
sqlsourceid
sqlflags
sqlreserved

######

sqlilongdata

sqllen#32###########sqldata######

##########

```
EXEC SQL INCLUDE sqlda.h;


    sqlda_t        *sqlda; /* #########DECLARE SECTION########## */

    EXEC SQL BEGIN DECLARE SECTION;
    char *prep_stmt = "select * from table1";
    int i;
    EXEC SQL END DECLARE SECTION;

    ...

    EXEC SQL PREPARE mystmt FROM :prep_stmt;

    EXEC SQL DESCRIBE mystmt INTO sqlda;

    printf("# of fields: %d\n", sqlda->sqld);
    for (i = 0; i < sqlda->sqld; i++)
      printf("field %d: \"%s\"\n", sqlda->sqlvar[i]->sqlname);

    EXEC SQL DECLARE mycursor CURSOR FOR mystmt;
    EXEC SQL OPEN mycursor;
    EXEC SQL WHENEVER NOT FOUND GOTO out;

    while (1)
    {
      EXEC SQL FETCH mycursor USING sqlda;
    }

    EXEC SQL CLOSE mycursor;


    free(sqlda); /* #########free()####
                  * sqlda#sqlda->sqlvar#1############ */
```

#########sqlda.h########src/interfaces/ecpg/test/compat_informix/sqlda.pgc###################

34.15.4.

`decadd`

2##decimal#########

```
int decadd(decimal *arg1, decimal *arg2, decimal *sum);
```

######decimal########(arg1)#######decimal##2#######(arg2)###############
decimal##(sum)############ ###########0###### ##############
#ECPG_INFORMIX_NUM_OVERFLOW###########
#ECPG_INFORMIX_NUM_UNDERFLOW###### ##############-1####errno##pgtypeslib#
#######errno#########

`deccmp`

2##decimal#########

```
int deccmp(decimal *arg1, decimal *arg2);
```

#########decimal#(arg1)#######2###decimal#(arg2)#####################int#####
##

- arg1#######arg2############1#
- arg1#######arg2############-1#
- arg1#######arg2###########0#

`deccopy`

decimal########

```
void deccopy(decimal *src, decimal *target);
```

##################decimal#(src)#######2########decimal#########(target)#####
########

`deccvasc`

ASCII####decimal#########

```
int deccvasc(char *cp, int len, decimal *np);
```

####################(cp)############len####### np###########decimal######
######

-2# .794# +3.44# 592.49E07# -32.84e-4#

########0###### ####################
#ECPG_INFORMIX_NUM_OVERFLOW#ECPG_INFORMIX_NUM_UNDERFLOW######
ASCII##############ECPG_INFORMIX_BAD_NUMERIC###############
#ECPG_INFORMIX_BAD_EXPONENT######

`deccvdbl`

double####decimal#########

```
int deccvdbl(double dbl, decimal *np);
```

##################double####(dbl)####### 2#####(np)##################decimal#######################

#########0###### ##################

deccvint

int####decimal##########

```
int deccvint(int in, decimal *np);
```

##################int###(in)####### 2#####(np)##################decimal#########################

#########0###### ##################

deccvlong

long####decimal##########

```
int deccvlong(long lng, decimal *np);
```

##################long###(lng)####### 2#####(np)##################decimal###############################

#########0###### ##################

decdiv

2##decimal###########

```
int decdiv(decimal *n1, decimal *n2, decimal *result);
```

######1#######(n1)#2#######(n2)###############n1/n2####### result###

###0####################### ##############################
#ECPG_INFORMIX_NUM_OVERFLOW#ECPG_INFORMIX_NUM_UNDERFLOW###### 0#######ECPG_INFORMIX_DIVIDE_ZERO#######

decmul

2##decimal#########

```
int decmul(decimal *n1, decimal *n2, decimal *result);
```

######1#######(n1)#2#######(n2)###############n1*n2####### result###

###0####################### ##############################
#ECPG_INFORMIX_NUM_OVERFLOW#ECPG_INFORMIX_NUM_UNDERFLOW######

decsub

decimal##############

```
int decsub(decimal *n1, decimal *n2, decimal *result);
```

######1#######(n1)#2#######(n2)###############n1-n2####### result###

###0####################### ##############################
#ECPG_INFORMIX_NUM_OVERFLOW#ECPG_INFORMIX_NUM_UNDERFLOW######

dectoasc

decimal####C char* ####ASCII#########

```
int dectoasc(decimal *np, char *cp, int len, int right)
```

#####decimal###(np)####################### cp################### right#####
##10####################### #####10########## right#-1#####################
######## len###############NULL################################*###1######
##-1#######

######cp#############-1###### #########ECPG_INFORMIX_OUT_OF_MEMORY######

dectodbl

decimal####double#######

```
int dectodbl(decimal *np, double *dblp);
```

##########decimal###(np)###############double##(dblp)##############

###0#################

dectoint

decimal####integer########

```
int dectoint(decimal *np, int *ip);
```

##########decimal###(np)###############integer###(ip)##############

###0################# ###############ECPG_INFORMIX_NUM_OVERFLOW#######

##ECPG####Informix################### Informix###integer###-32767##32767########
#######ECPG###############(INT_MIN .. INT_MAX)####

dectolong

decimal####long integer########

```
int dectolong(decimal *np, long *lngp);
```

##########decimal###(np)###############long##(lngp)##############

###0################# ###############ECPG_INFORMIX_NUM_OVERFLOW#######

##ECPG####Informix################### Informix###long integer###-2,147,483,647##
2,147,483,647###############ECPG###############(-LONG_MAX .. LONG_MAX)####

rdatestr

date##C char*##########

```
int rdatestr(date d, char *str);
```

#####2########### ###########date#(d)#2################### #######yyyy-mm-dd#
#########11###NULL##########################

########0##############

##ECPG####Informix################## Informix####################ECPG###### #############

rstrdate

date##############

```
int rstrdate(char *str, date *d);
```

###########date########(str)#date########(d)####### ######################### Informix##############mm/dd/yyyy####### ##########rdefmtdate########### ## #####rstrdate######## ################################rdefmtdate#### #####

#####rdefmtdate##########

rtoday

#######date#########

```
void rtoday(date *d);
```

#####date###(d)#########################

###########PGTYPESdate_today########

rjulmdy

date####################

```
int rjulmdy(date d, short mdy[3]);
```

#######d#3##short integer########mdy############ ##############mdy[0]### ##mdy[1]######mdy[2]########

#########0######

##########PGTYPESdate_julmdy########

rdefmtdate

##############date#########

```
int rdefmtdate(date *d, char *fmt, char *str);
```

##################date#######(d)################(fmt)#date##########C#char*### (str)####### ################################ ##############1:1############## ## #####################yy###yyyy########mm#########dd#######

###############

- 0 - ############
- ECPG_INFORMIX_ENOSHORTDATE - ######################### ###########6####8 ################################
- ECPG_INFORMIX_ENOTDMY - #######################

- ECPG_INFORMIX_BAD_DAY - ###################
- ECPG_INFORMIX_BAD_MONTH - ###################
- ECPG_INFORMIX_BAD_YEAR - ###################

###########PGTYPESdate_defmt_asc############## ########################
###########

rfmtdate

##########date################

```
int rfmtdate(date d, char *fmt, char *str);
```

############(d)######(fmt)#################(str)#######

###0#####################

##########PGTYPESdate_fmt_asc######## ########################

rmdyjul

###########3##short integer####################

```
int rmdyjul(short mdy[3], date *d);
```

#####3##short integer#######(mdy)##########date###############

#########0######

##########PGTYPESdate_mdyjul##############

rdayofweek

###################

```
int rdayofweek(date d);
```

#####date###d##################################

- 0 - ##
- 1 - ##
- 2 - ##
- 3 - ##
- 4 - ##
- 5 - ##
- 6 - ##

########## PGTYPESdate_dayofweek##############

dtcurrent

##################

```
void dtcurrent(timestamp *ts);
```

####################ts####################

dtcvasc

########timestamp################

```
int dtcvasc(char *str, timestamp *ts);
```

###########(str)##########timestamp###(ts)############

########0################

##########PGTYPEStimestamp_from_asc######## ########################

dtcvfmtasc

########################timestamp##########

```
dtcvfmtasc(char *inbuf, char *fmtstr, timestamp *dtvalue)
```

##############(inbuf)##########(fmtstr)##########timestamp##(dtvalue)#######
######

#####PGTYPEStimestamp_defmt_asc############### ######################
##########

#########0##############

dtsub

timestamp##########interval########

```
int dtsub(timestamp *ts1, timestamp *ts2, interval *iv);
```

#####ts1#####timestamp#####ts2#####timestamp######## ###iv#####interval#######
####

########0################

dttoasc

timestamp####C char*##########

```
int dttoasc(timestamp *ts, char *output);
```

########timestamp###(ts)##################(output)####### ###ts###SQL######
####YYYY-MM-DD HH:MM:SS############

########0################

dttofmtasc

##########timestamp####C char*#######

```
int dttofmtasc(timestamp *ts, char *output, int str_len, char *fmtstr);
```

#########################(ts)############(output)################
(str_len)############(fmtstr)#######

########0#########################

##########PGTYPEStimestamp_fmt_asc######## ##########################
########

intoasc

interval####C char*##########

```
int intoasc(interval *i, char *str);
```

###########interval###(i)##################(str)####### ###i###SQL########
#YYYY-MM-DD HH:MM:SS############

#########0###### ###################

rfmtlong

long integer########################

```
int rfmtlong(long lng_val, char *fmt, char *outbuf);
```

######long###lng_val#####fmt######outbuf############ ###long###########
##############

#############################

- * (######) - ######################
- & (######) - ##########0######
- # - ##############
- < - ###############
- , (###) - 4##############3##########
- . (####) - ###################
- - (####) - ###################
- + (###) - #################
- (- ######################## #############
-) - ############################
- $ - ####

rupshift

##############

```
void rupshift(char *str);
```

####################################

byleng

############## ###############

```
int byleng(char *str, int len);
```

####################(str)##2############# (len)####### ################## ##############

ldchar

########NULL#############

```
void ldchar(char *src, int len, char *dest);
```

#################(src)#####(len)#######(dest)############ dest########### ###len+1######################### ########len################ ########## ################## ####NULL########

rgetmsg

```
int rgetmsg(int msgnum, char *s, int maxsize);
```

#######################

rtypalign

```
int rtypalign(int offset, int type);
```

#######################

rtypmsize

```
int rtypmsize(int type, int len);
```

#######################

rtypwidth

```
int rtypwidth(int sqltype, int sqllen);
```

#######################

rsetnull

###NULL######

```
int rsetnull(int t, char *ptr);
```

################C char*#######################

############

- CCHARTYPE - char### char*#####
- CSHORTTYPE - short int#####
- CINTTYPE - int#####
- CBOOLTYPE - boolean#####
- CFLOATTYPE - float#####

- CLONGTYPE - long#####
- CDOUBLETYPE - double#####
- CDECIMALTYPE - decimal#####
- CDATETYPE - date#####
- CDTIMETYPE - timestamp#####

##################

```
$char c[] = "abc       ";
$short s = 17;
$int i = -74874;

rsetnull(CCHARTYPE, (char *) c);
rsetnull(CSHORTTYPE, (char *) &s);
rsetnull(CINTTYPE, (char *) &i);
```

risnull

###NULL#######

```
int risnull(int t, char *ptr);
```

##############(t)###(ptr)############ ###char*######################## ####### ###### rsetnull###########

################

```
$char c[] = "abc       ";
$short s = 17;
$int i = -74874;

risnull(CCHARTYPE, (char *) c);
risnull(CSHORTTYPE, (char *) &s);
risnull(CINTTYPE, (char *) &i);
```

34.15.5.

################################

ECPG_INFORMIX_NUM_OVERFLOW

############################## #####-1200#Informix##############

ECPG_INFORMIX_NUM_UNDERFLOW

############################## #####-1201#Informix##############

ECPG_INFORMIX_DIVIDE_ZERO

########################### #####-1202#Informix##############

ECPG_INFORMIX_BAD_YEAR

############################## #####-1204#Informix##############

`ECPG_INFORMIX_BAD_MONTH`

################################ #####-1205#Informix##############

`ECPG_INFORMIX_BAD_DAY`

################################ #####-1206#Informix##############

`ECPG_INFORMIX_ENOSHORTDATE`

#####-1209#Informix#########################

`ECPG_INFORMIX_DATE_CONVERT`

################################ #####-1210#Informix##############

`ECPG_INFORMIX_OUT_OF_MEMORY`

########################### #####-1211#Informix##############

`ECPG_INFORMIX_ENOTDMY`

###########mmddyy###-1212#Informix##############

`ECPG_INFORMIX_BAD_NUMERIC`

##1## #####-1213#Informix##############

`ECPG_INFORMIX_BAD_EXPONENT`

################################### #####-1216#Informix##############

`ECPG_INFORMIX_BAD_DATE`

############################### #####-1218#Informix##############

`ECPG_INFORMIX_EXTRA_CHARS`

################################### #####-1264#Informix##############

34.16. Oracle#####

`ecpg`#####*Oracle*############ #############Oracle Pro*C#################

###########3####`ecpg`######

- ##################################
- #######################################
- ######################NULL####`-1`######

34.17.

########ECPG########## #####ECPG##########################

`ecpg`################4########### 2#########2################################ ################################ ######################

EXEC SQL################### #######EXEC SQL#####;####### ########SQL######## ###############

###############:############### #########EXEC SQL DECLARE################### #######

###############ECPGdo### ###################### ########### ################### ###############50################

##########

###

##########

###

#####SQL####### ## ############ ####?########

####

#########10###############

ECPGt_EOIT

#############enum###

####

#########10############### ###################

ECPGt_EORT

###########enum###

SQL############################10###########

1. ###########
2. ######################
3. ###char#varchar#########
4. ###############
5. ######################
6. #################
7. ###########
8. 0
9. ###################
10. ##########################

####SQL############################## ##############SQL##############

EXEC SQL OPEN *cursor*;

###########DECLARE#####OPEN############## ######################

```
####foo.pgc#################################   ##################################
###


EXEC SQL BEGIN DECLARE SECTION;
int index;
int result;
EXEC SQL END DECLARE SECTION;
...
EXEC SQL SELECT res INTO :result FROM mytable WHERE index = :index;

#############


/* Processed by ecpg (2.6.0) */
/* These two include files are added by the preprocessor */
#include <ecpgtype.h>;
#include <ecpglib.h>;

/* exec sql begin declare section */

#line 1 "foo.pgc"

 int index;
 int result;
/* exec sql end declare section */
...
ECPGdo(__LINE__, NULL, "SELECT res FROM mytable WHERE index = ?     ",
        ECPGt_int,&(index),1L,1L,sizeof(int),
        ECPGt_NO_INDICATOR, NULL , 0L, 0L, 0L, ECPGt_EOIT,
        ECPGt_int,&(result),1L,1L,sizeof(int),
        ECPGt_NO_INDICATOR, NULL , 0L, 0L, 0L, ECPGt_EORT);
#line 147 "foo.pgc"


######################### ####################
```

#35#

########SQL##########
#####PostgreSQL## #
###############PostgreSQL#################################### ###########################
PostgreSQL#####################################

##

########
############SQL##############PostgreSQL############### PostgreSQL######
##

####
##check_constraint_routine_usage#check_constraints#domain_constraints#refer
################################## ##
##########
#constraint_column_usage#constraint_table_usage#table_constraints#
####

35.1.

##########information_schema############ ########################### ######
#############
##############

################################## ##
##
######

35.2.

##################################
################################## ##
#####

#############

cardinal_number

########

character_data

##################

sql_identifier

####SQL##########character_data#####################

time_stamp

timestamp with time zone##########

yes_or_no

YES#NO################# ##########(#/#)################ ########boolean####
SQL################ ####################################

###############5#############

35.3. information_schema_catalog_name

information_schema_catalog_name##############SQL##############1#1#####
######

#35.1 **information_schema_catalog_name##**

##
catalog_name sql_identifier ######################

35.4. administrable_role_authorizations

administrable_role_authorizations##################################

#35.2 **administrable_role_authorizations##**

##
grantee sql_identifier ######################## ## ###
role_name sql_identifier ######
is_grantable yes_or_no ##YES###

35.5. applicable_roles

applicable_roles#################################### ####################
###################### ################### ######################

#35.3 **applicable_roles##**

##
grantee sql_identifier ######################## ## ###
role_name sql_identifier ######
is_grantable yes_or_no #######################YES######NO###

35.6. attributes

attributes#################################### ########PostgreSQL#########
#################################### ####################################
##################

#35.4 attributes##

# #	##
udt_catalog sql_identifier	##################################
udt_schema sql_identifier	################
udt_name sql_identifier	#########
attribute_name sql_identifier	#######
ordinal_position cardinal_number	##############1########
attribute_default character_data	###########
is_nullable yes_or_no	###NULL###########YES######NO###
data_type character_data	######################## ##########ARRAY### ######element_types#### ######### #####USER-DEFINED### #######attribute_udt_name####### ########
character_maximum_length cardinal_number	data_type################################ ######################### NULL###
character_octet_length cardinal_number	data_type##################################### ########NULL### #### ###############(##)###############
character_set_catalog sql_identifier	PostgreSQL###################
character_set_schema sql_identifier	PostgreSQL###################
character_set_name sql_identifier	PostgreSQL###################
collation_catalog sql_identifier	#############(###########)################################NULL###
collation_schema sql_identifier	##NULL###
collation_name sql_identifier	####################################NULL###
numeric_precision cardinal_number	data_type## ############### numeric_precision_radix########(10#####)10######(2#####)2######### ## ###############NULL###
numeric_precision_radix cardinal_number	data_type################numeric_precision###numeric_scale### ############## ####2###10### ###############NULL###
numeric_scale cardinal_number	data_type## ######### ########## numeric_precision_radix########(10#####)10######(2#### #)2########### ###############NULL###

##
datetime_precision cardinal_number   data_type##   ###### ##################10###### ################NULL###
interval_type character_data   data_type###YEAR TO MONTH#DAY TO SECOND##### ##################(###################)#   ########################NULL###
interval_precision cardinal_number   PostgreSQL####################(########################   #datetime_precision#########)
attribute_udt_catalog sql_identifier   ######################### ################
attribute_udt_schema sql_identifier   ########################
attribute_udt_name sql_identifier   #############
scope_catalog sql_identifier   PostgreSQL####################
scope_schema sql_identifier   PostgreSQL####################
scope_name sql_identifier   PostgreSQL####################
maximum_cardinality cardinal_number   ##NULL###PostgreSQL################
dtd_identifier sql_identifier   ## ########################### ###   ##
is_derived_reference_attribute yes_or_no   PostgreSQL####################

###35.17########## ##################################

35.7. character_sets

##character_sets############################## PostgreSQL#################### ##

######SQL##################

####

###UNICODE#UCS#LATIN1###############SQL##############################

#######

############# ##(### ##LATIN2#LATIN2###############)# ######Unicode##UTF8#UTF16############# #PostgreSQL################ ######SQL#############################

#####

################################SQL## ################## #########UTF8#########UCS######UTF8##################

PostgreSQL#####encoding###
######

#35.5 character_sets##

# #	##
character_set_catalog sql_identifier	################################NULL###
character_set_schema sql_identifier	################################NULL###
character_set_name sql_identifier	###
character_repertoire sql_identifier	##############UTF8####UCS###########################
form_of_use sql_identifier	#######, ###################
default_collate_catalog sql_identifier	###############(##########################)######
default_collate_schema sql_identifier	###################
default_collate_name sql_identifier	##############################COLLATE#CTYPE####################### ##############################NULL###

35.8. check_constraint_routine_usage

check_constraint_routine_usage############################## ############
############

#35.6 check_constraint_routine_usage##

# #	##
constraint_catalog sql_identifier	################################
constraint_schema sql_identifier	################
constraint_name sql_identifier	#######
specific_catalog sql_identifier	################################
specific_schema sql_identifier	################
specific_name sql_identifier	########### ###35.45#########

35.9. check_constraints

check_constraints###
############################

######SQL######null##########_##IS NOT null############### #######null#### ##################

#35.7 check_constraints##

##	##
constraint_catalog	sql_identifier ################################
constraint_schema	sql_identifier ################
constraint_name	sql_identifier #######
check_clause	character_data ############

35.10. collations

collations#######################

#35.8 collations##

##	##
collation_catalog	sql_identifier ##############################
collation_schema	sql_identifier ##############
collation_name	sql_identifier ############
pad_attribute	character_data ##NO PAD###(#####PAD SPACE#PostgreSQL############)

35.11. collation_character_set_applicability

collation_character_set_applicability##############################
PostgreSQL#########################(35.7###########)####################

#35.9 collation_character_set_applicability##

##	##
collation_catalog	sql_identifier ##############################
collation_schema	sql_identifier ##############
collation_name	sql_identifier ############
character_set_catalog	sql_identifier

##
################################NULL###
character_set_schema sql_identifier   ################################NULL###
character_set_name sql_identifier   ##########

35.12. column_column_usage

column_column_usage################################ ################################

#35.10 column_column_usage##

##
table_catalog sql_identifier   ################################
table_schema sql_identifier   ################
table_name sql_identifier   #########
column_name sql_identifier   ##################
dependent_column sql_identifier   ###########

35.13. column_domain_usage

column_domain_usage##

#35.11 column_domain_usage##

##
domain_catalog sql_identifier   ################################
domain_schema sql_identifier   ################
domain_name sql_identifier   #########
table_catalog sql_identifier   ################################
table_schema sql_identifier   ################
table_name sql_identifier   #########
column_name sql_identifier   ######

35.14. column_options

column_options## #######(#########
#######)##########################

#35.12 column_options##

##	
table_catalog sql_identifier	#####################################
table_schema sql_identifier	##################
table_name sql_identifier	###########
column_name sql_identifier	#######
option_name sql_identifier	###########
option_value character_data	#########

35.15. column_privileges

column_privileges## ########
#################1#######

##
##SELECT#INSERT#UPDATE#REFERENCES###########################

#35.13 column_privileges##

##	
grantor sql_identifier	##############
grantee sql_identifier	################
table_catalog sql_identifier	##
table_schema sql_identifier	######################
table_name sql_identifier	###############
column_name sql_identifier	#######
privilege_type character_data	######## SELECT#INSERT#UPDATE#####REFERENCES###
is_grantable yes_or_no	############YES######NO###

35.16. column_udt_usage

column_udt_usage################################### PostgreSQL############## ################################## ###35.17##########

#35.14 column_udt_usage##

##
udt_catalog sql_identifier ##
udt_schema sql_identifier ###
udt_name sql_identifier ##################################
table_catalog sql_identifier ##################################
table_schema sql_identifier ################
table_name sql_identifier #########
column_name sql_identifier #######

35.17. columns

columns## ######ctid########### ## ##

#35.15 columns##

##
table_catalog sql_identifier ##################################
table_schema sql_identifier ################
table_name sql_identifier #########
column_name sql_identifier #######
ordinal_position cardinal_number ###########1###########
column_default character_data ###########
is_nullable yes_or_no ##NULL############YES#NULL############NO### #NULL######NULL###### ########1###############
data_type character_data

##	##
	####################ARRAY######element_types################ ##USER-DEFINED#########udt_name############## ################### ################ #########domain_name################
character_maximum_length	cardinal_number data_type############################### ####################### NULL###
character_octet_length	cardinal_number data_type##################################### ########NULL### #### ##############(##)###############
numeric_precision	cardinal_number data_type## ######### ###### numeric_precision_radix########10###10########2###2######### ## ###############NULL###
numeric_precision_radix	cardinal_number data_type#################numeric_precision###numeric_scale### ############## ####2###10### ###############NULL###
numeric_scale	cardinal_number data_type### ####### ############## numeric_precision_radix########10###10########2###2 ########## ###############NULL###
datetime_precision	cardinal_number data_type## ###########10###### ###############NULL###
interval_type	character_data data_type###YEAR TO MONTH#DAY TO SECOND##### #################(##################)# #######################NULL###
interval_precision	cardinal_number PostgreSQL##################(################## #datetime_precision########)
character_set_catalog	sql_identifier PostgreSQL###################
character_set_schema	sql_identifier PostgreSQL###################
character_set_name	sql_identifier PostgreSQL###################
collation_catalog	sql_identifier ############(##########)################################NULL###
collation_schema	sql_identifier ##NULL###
collation_name	sql_identifier ##################################NULL###
domain_catalog	sql_identifier ################################ ############### ######NULL###
domain_schema	sql_identifier ############################## ######NULL###
domain_name	sql_identifier ###################### ######NULL###
udt_catalog	sql_identifier

##
##
udt_schema sql_identifier   ###
udt_name sql_identifier   ##################################
scope_catalog sql_identifier   PostgreSQL###################
scope_schema sql_identifier   PostgreSQL###################
scope_name sql_identifier   PostgreSQL###################
maximum_cardinality cardinal_number   ##NULL###PostgreSQL################
dtd_identifier sql_identifier   ################################## ######################### ######   ######################################
is_self_referencing yes_or_no   PostgreSQL###################
is_identity yes_or_no   ########YES#######NO###
identity_generation character_data   ##################ALWAYS###BY DEFAULT###
identity_start character_data   ##########################NULL###
identity_increment character_data   #########################NULL###
identity_maximum character_data   ##########################NULL###
identity_minimum character_data   ##########################NULL###
identity_cycle yes_or_no   ########################YES########NO########NULL###
is_generated character_data   ###########ALWAYS#######NEVER###
generation_expression character_data   ######################NULL###
is_updatable yes_or_no   ########YES######NO### ##############################

###########SQL##########PostgreSQL## ######### data_type################################### PostgreSQL#### #pg_catalog########################## ##################################### ## udt_name#udt_schema#udt_catalog## #PostgreSQL##SQL######## ########## ################################ ##################################### ####### ##############domain_name#domain_schema#domain_catalog######## ####################################coalesce(domain_name, udt_name)###### ######

35.18. constraint_column_usage

#####[80.769231]#constraint_column_usage############################## #### ######################### ############################ ######################## ####### ################################## ##########_###_usage############## ################## ################################ ######################## ######## null################################### ############################ #### ################################

#35.16 constraint_column_usage##

##	##
table_catalog	sql_identifier ######################## ################
table_schema	sql_identifier ##############################
table_name	sql_identifier ########################
column_name	sql_identifier ################
constraint_catalog	sql_identifier ##############################
constraint_schema	sql_identifier ##############
constraint_name	sql_identifier ########

35.19. constraint_table_usage

constraint_table_usage## ## ##############################table_constraints######## ################ ################ ## ######NULL########## #######

#35.17 constraint_table_usage##

##	##
table_catalog	sql_identifier ########################## ################
table_schema	sql_identifier ##########################
table_name	sql_identifier ###################
constraint_catalog	sql_identifier ##############################
constraint_schema	sql_identifier ##############
constraint_name	sql_identifier ########

35.20. data_type_privileges

data_type_privileges## ################## ##1## ##

#35.18 data_type_privileges##

# #	##
object_catalog sql_identifier	##
object_schema sql_identifier	#########################
object_name sql_identifier	##################
object_type character_data	################## TABLE#######################DOMAIN################ ######ROUTINE##
dtd_identifier sql_identifier	############# ################################

35.21. domain_constraints

domain_constraints#################################### ################### ############################

#35.19 domain_constraints##

# #	##
constraint_catalog sql_identifier	###############################
constraint_schema sql_identifier	##############
constraint_name sql_identifier	#######
domain_catalog sql_identifier	##################################
domain_schema sql_identifier	################
domain_name sql_identifier	#########
is_deferrable yes_or_no	#########YES######NO#
initially_deferred yes_or_no	##################YES######NO#

35.22. domain_udt_usage

domain_udt_usage## PostgreSQL##########
######################################

#35.20 domain_udt_usage##

##	##
udt_catalog	sql_identifier ######################################
udt_schema	sql_identifier ######################
udt_name	sql_identifier #############
domain_catalog	sql_identifier ##################################
domain_schema	sql_identifier ################
domain_name	sql_identifier #########

35.23. domains

domains################################## ################################
#############

#35.21 domains##

##	##
domain_catalog	sql_identifier ##################################
domain_schema	sql_identifier ################
domain_name	sql_identifier #########
data_type	character_data ############################ARRAY## ######element_types######### #### #####USER-DEFINED## #########udt_name##############
character_maximum_length	cardinal_number ################################## ######################NULL###
character_octet_length	cardinal_number ###########1######################### #########NULL### ########### ######(##)###############
character_set_catalog	sql_identifier PostgreSQL###################
character_set_schema	sql_identifier PostgreSQL###################
character_set_name	sql_identifier PostgreSQL###################

# #	##
collation_catalog	sql_identifier ###############(##########)################################NULL###
collation_schema	sql_identifier ##NULL###
collation_name	sql_identifier ##NULL###
numeric_precision	cardinal_number ## ################ numeric_precision_radix ######10###10#####2###2########### ####### ########NULL###
numeric_precision_radix	cardinal_number ################numeric_precision#numeric_scale################ ####### 2####10####### ################NULL###
numeric_scale	cardinal_number ### ################ ####### numeric_precision_radix#######10###10#######2###2####### #### ################NULL###
datetime_precision	cardinal_number data_type### ####################10####### ################NULL###
interval_type	character_data data_type###YEAR TO MONTH#DAY TO SECOND##### ##################(###################)# #######################NULL###
interval_precision	cardinal_number PostgreSQL####################(#################### #datetime_precision########)
domain_default	character_data ##############
udt_catalog	sql_identifier ##
udt_schema	sql_identifier #######################
udt_name	sql_identifier ##############
scope_catalog	sql_identifier PostgreSQL####################
scope_schema	sql_identifier PostgreSQL####################
scope_name	sql_identifier PostgreSQL####################
maximum_cardinality	cardinal_number ##NULL###PostgreSQL################
dtd_identifier	sql_identifier ################################### #######1########################### ### ################################ ################################ ##################

35.24. element_types

element_types###################### ######################################
######## #############data_type##ARRAY######## ###########################
############### ##

```
SELECT c.column_name, c.data_type, e.data_type AS element_type
FROM information_schema.columns c LEFT JOIN
 information_schema.element_types e
     ON ((c.table_catalog, c.table_schema, c.table_name, 'TABLE',
 c.dtd_identifier)
       = (e.object_catalog, e.object_schema, e.object_name, e.object_type,
 e.collection_type_identifier))
WHERE c.table_schema = '...' AND c.table_name = '...'
ORDER BY c.ordinal_position;
```

##

#35.22 element_types##

##	##
object_catalog	sql_identifier ############################## ###############
object_schema	sql_identifier ##############################
object_name	sql_identifier #######################
object_type	character_data ####################### TABLE#####################USER-DEFINED TYPE#########################DOMAIN####################ROUTINE#### #####################################
collection_type_identifier	sql_identifier ################################dtd_identifier################
data_type	character_data #########################USER-DEFINED## #######udt_name######### #####
character_maximum_length	cardinal_number ##NULL### ######PostgreSQL#########################
character_octet_length	cardinal_number ##NULL### ######PostgreSQL#########################
character_set_catalog	sql_identifier PostgreSQL###################
character_set_schema	sql_identifier PostgreSQL###################
character_set_name	sql_identifier PostgreSQL###################
collation_catalog	sql_identifier ################(###########)##############################NULL###
collation_schema	sql_identifier ##NULL###
collation_name	sql_identifier ######################################NULL###

##
numeric_precision cardinal_number ##NULL### ######PostgreSQL##########################
numeric_precision_radix cardinal_number ##NULL### ######PostgreSQL##########################
numeric_scale cardinal_number ##NULL### ######PostgreSQL##########################
datetime_precision cardinal_number ##NULL### ######PostgreSQL##########################
interval_type character_data ##NULL### ######PostgreSQL##########################
interval_precision cardinal_number ##NULL### ######PostgreSQL##########################
udt_catalog sql_identifier ###################### ###############
udt_schema sql_identifier #####################
udt_name sql_identifier ############
scope_catalog sql_identifier PostgreSQL###################
scope_schema sql_identifier PostgreSQL###################
scope_name sql_identifier PostgreSQL###################
maximum_cardinality cardinal_number ##NULL###PostgreSQL###############
dtd_identifier sql_identifier ###########################

35.25. enabled_roles

enabled_roles################### ## ############ ##

######################### ######################### ################ #applicable_roles############### applicable_roles##########35.5######## ##

#35.23 enabled_roles##

##
role_name sql_identifier ######

35.26. foreign_data_wrapper_options

foreign_data_wrapper_options## ## ##

#35.24 foreign_data_wrapper_options##

##
foreign_data_wrapper_catalog sql_identifier ##
foreign_data_wrapper_name sql_identifier ##############
option_name sql_identifier ##########
option_value character_data #########

35.27. foreign_data_wrappers

foreign_data_wrappers## ############ ##

#35.25 foreign_data_wrappers##

##
foreign_data_wrapper_catalog sql_identifier ##################################
foreign_data_wrapper_name sql_identifier ##############
authorization_identifier sql_identifier ##############
library_name character_data #########################
foreign_data_wrapper_language character_data ###########################

35.28. foreign_server_options

foreign_server_options## ###### ##

#35.26 foreign_server_options##

##
foreign_server_catalog sql_identifier ##################################
foreign_server_name sql_identifier #######
option_name sql_identifier ##########
option_value character_data #########

35.29. foreign_servers

foreign_servers################################### ####################### ############################

#35.27 foreign_servers##

##	##
foreign_server_catalog	sql_identifier ##################################
foreign_server_name	sql_identifier #######
foreign_data_wrapper_catalog	sql_identifier ##
foreign_data_wrapper_name	sql_identifier ##########################
foreign_server_type	character_data ############################
foreign_server_version	character_data ##############################
authorization_identifier	sql_identifier ##############

35.30. foreign_table_options

foreign_table_options##(############ ######)############################

#35.28 foreign_table_options##

##	##
foreign_table_catalog	sql_identifier #####################################
foreign_table_schema	sql_identifier ##################
foreign_table_name	sql_identifier ###########
option_name	sql_identifier ##########
option_value	character_data #########

35.31. foreign_tables

foreign_tables####################################(##################)###### #######################

#35.29 foreign_tables##

##
foreign_table_catalog sql_identifier ##################################
foreign_table_schema sql_identifier ##################
foreign_table_name sql_identifier ###########
foreign_server_catalog sql_identifier ##################################
foreign_server_name sql_identifier #######

35.32. key_column_usage

key_column_usage## ################## ##

#35.30 key_column_usage##

##
constraint_catalog sql_identifier ##############################
constraint_schema sql_identifier ##############
constraint_name sql_identifier #######
table_catalog sql_identifier ################################## ###############
table_schema sql_identifier ##################################
table_name sql_identifier ###########################
column_name sql_identifier ###################
ordinal_position cardinal_number ###########1################
position_in_unique_constraint cardinal_number ###########################1########## #######NULL###

35.33. parameters

parameters## ##################### ##################################

#35.31 parameters##

specific_catalog sql_identifier
specific_schema sql_identifier ################
specific_name sql_identifier ########### ###35.45#########
ordinal_position cardinal_number #######################1########
parameter_mode character_data ########IN#########OUT##########INOUT ###
is_result yes_or_no PostgreSQL###################
as_locator yes_or_no PostgreSQL###################
parameter_name sql_identifier ########### ###########NULL###
data_type character_data #################### #########ARRAY## ######element_types######## ##### ######USER-DEFINED## ########udt_name ##############
character_maximum_length cardinal_number ##NULL### PostgreSQL###########################
character_octet_length cardinal_number ##NULL### PostgreSQL###########################
character_set_catalog sql_identifier PostgreSQL###################
character_set_schema sql_identifier PostgreSQL###################
character_set_name sql_identifier PostgreSQL###################
collation_catalog sql_identifier ##NULL### PostgreSQL############################
collation_schema sql_identifier ##NULL### PostgreSQL###########################
collation_name sql_identifier ##NULL### PostgreSQL###########################
numeric_precision cardinal_number ##NULL### PostgreSQL###########################
numeric_precision_radix cardinal_number ##NULL### PostgreSQL###########################
numeric_scale cardinal_number ##NULL### PostgreSQL###########################
datetime_precision cardinal_number ##NULL### PostgreSQL###########################
interval_type character_data ##NULL### PostgreSQL###########################

##	
	##
interval_precision cardinal_number	
	##NULL### PostgreSQL###########################
udt_catalog sql_identifier	
	######################### ################
udt_schema sql_identifier	
	#########################
udt_name sql_identifier	
	################
scope_catalog sql_identifier	
	PostgreSQL####################
scope_schema sql_identifier	
	PostgreSQL####################
scope_name sql_identifier	
	PostgreSQL####################
maximum_cardinality cardinal_number	
	##NULL###PostgreSQL################
dtd_identifier sql_identifier	
	##################################### ################################## ###
parameter_default character_data	
	##NULL###

35.34. referential_constraints

referential_constraints###
#######SELECT##

#35.32 referential_constraints##

##	
	##
constraint_catalog sql_identifier	
	##################################
constraint_schema sql_identifier	
	################
constraint_name sql_identifier	
	########
unique_constraint_catalog sql_identifier	
	## ################
unique_constraint_schema sql_identifier	
	##
unique_constraint_name sql_identifier	
	###############################
match_option character_data	
	################ FULL#PARTIAL#NONE########
update_rule character_data	
	############# CASCADE#SET NULL#SET DEFAULT#RESTRICT#NO ACTION# #######

##
delete_rule character_data ############# CASCADE#SET NULL#SET DEFAULT#RESTRICT#NO ACTION# #######

35.35. role_column_grants

role_column_grants## #####
#column_privileges####### #####column_privileges#####################
##PUBLIC#####################################

#35.33 role_column_grants##

##
grantor sql_identifier ##############
grantee sql_identifier ################
table_catalog sql_identifier ######################################
table_schema sql_identifier ######################
table_name sql_identifier ###############
column_name sql_identifier ######
privilege_type character_data ####### SELECT#INSERT#UPDATE#####REFERENCES###
is_grantable yes_or_no ############YES######NO###

35.36. role_routine_grants

role_routine_grants## #####
#routine_privileges####### #####routine_privileges#####################
####PUBLIC#####################################

#35.34 role_routine_grants##

##
grantor sql_identifier ##############
grantee sql_identifier ################
specific_catalog sql_identifier ################################
specific_schema sql_identifier ################

##
specific_name sql_identifier ########### ###35.45#########
routine_catalog sql_identifier ##################################
routine_schema sql_identifier ################
routine_name sql_identifier ###################################
privilege_type character_data ##EXECUTE###############
is_grantable yes_or_no ############YES######NO###

35.37. role_table_grants

role_table_grants## #####
#table_privileges####### ######table_privileges#######################
#PUBLIC##

#35.35 role_table_grants##

##
grantor sql_identifier ##############
grantee sql_identifier ################
table_catalog sql_identifier ##################################
table_schema sql_identifier ################
table_name sql_identifier #########
privilege_type character_data ####### SELECT#INSERT#UPDATE#DELETE#TRUNCATE#REFERENCES### #TRIGGER#######
is_grantable yes_or_no ############YES######NO###
with_hierarchy yes_or_no SQL#####WITH HIERARCHY OPTION################################# PostgreSQL######SELECT##################SELECT####YES#########NO###

35.38. role_udt_grants

role_udt_grants#######################################USAGE################
#########udt_privileges####### ######udt_privileges####################
#######PUBLIC## PostgreSQL###########
#####PUBLIC############################

#35.36 role_udt_grants##

##
grantor sql_identifier   ###########
grantee sql_identifier   #############
udt_catalog sql_identifier   ###########################
udt_schema sql_identifier   ##########
udt_name sql_identifier   ####
privilege_type character_data   ##TYPE USAGE
is_grantable yes_or_no   ############YES######NO###

35.39. role_usage_grants

role_usage_grants##USAGE####### ## ####usage_privileges####### ######usage_privileges#################### ######PUBLIC##

#35.37 role_usage_grants##

##
grantor sql_identifier   ###########
grantee sql_identifier   #############
object_catalog sql_identifier   ################################
object_schema sql_identifier   ###############################
object_name sql_identifier   ###########
object_type character_data   COLLATION###DOMAIN###FOREIGN DATA WRAPPER###FOREIGN SERVER##   #SEQUENCE
privilege_type character_data   ##USAGE###
is_grantable yes_or_no   ############YES######NO###

35.40. routine_column_usage

routine_column_usage#####SQL## ######## ###############SQL#################################### ############ #########################

#35.38 routine_column_usage Columns

##
specific_catalog sql_identifier ##################################
specific_schema sql_identifier ################
specific_name sql_identifier ########### ###35.45#########
routine_catalog sql_identifier ##################################
routine_schema sql_identifier ################
routine_name sql_identifier ###################################
table_catalog sql_identifier ###
table_schema sql_identifier #########################
table_name sql_identifier ##################
column_name sql_identifier ##############

35.41. routine_privileges

routine_privileges### # #####################1######

#35.39 routine_privileges##

##
grantor sql_identifier ###############
grantee sql_identifier ################
specific_catalog sql_identifier ##################################
specific_schema sql_identifier ################
specific_name sql_identifier ########### ###35.45#########
routine_catalog sql_identifier

##
##
##################################
routine_schema sql_identifier ################
routine_name sql_identifier ###################################
privilege_type character_data ##EXECUTE###############
is_grantable yes_or_no ############YES######NO###

35.42. routine_routine_usage

routine_routine_usage#####SQL## ############################## ################SQL############################ ########## ## ################ ######

##
###SQL###################### ########35.45##########

#35.40 routine_routine_usage Columns

##
##
specific_catalog sql_identifier #####################################
specific_schema sql_identifier ####################
specific_name sql_identifier ###############
routine_catalog sql_identifier ##
routine_schema sql_identifier #########################
routine_name sql_identifier ######################

35.43. routine_sequence_usage

routine_sequence_usage#####SQL## ############# ################SQL#################################### ###### ######################################

#35.41 routine_sequence_usage Columns

##
##
specific_catalog sql_identifier ##################################
specific_schema sql_identifier

##
##
################
specific_name sql_identifier
########### ###35.45#########
routine_catalog sql_identifier
##################################
routine_schema sql_identifier
################
routine_name sql_identifier
###################################
schema_catalog sql_identifier
###
sequence_schema sql_identifier
#########################
sequence_name sql_identifier
##################

35.44. routine_table_usage

routine_table_usage## ## PostgreSQL################

#35.42 routine_table_usage Columns

##
##
specific_catalog sql_identifier
##################################
specific_schema sql_identifier
################
specific_name sql_identifier
########### ###35.45#########
routine_catalog sql_identifier
##################################
routine_schema sql_identifier
################
routine_name sql_identifier
###################################
table_catalog sql_identifier
###
table_schema sql_identifier
#########################
table_name sql_identifier
#################

35.45. routines

routines################################## ################################### ############################

#35.43 routines##

specific_catalog sql_identifier
specific_schema sql_identifier ################
specific_name sql_identifier ########### ## ##### ##
routine_catalog sql_identifier ################################
routine_schema sql_identifier ################
routine_name sql_identifier ###################################
routine_type character_data #######FUNCTION###########PROCEDURE###
module_catalog sql_identifier PostgreSQL###################
module_schema sql_identifier PostgreSQL###################
module_name sql_identifier PostgreSQL###################
udt_catalog sql_identifier PostgreSQL###################
udt_schema sql_identifier PostgreSQL###################
udt_name sql_identifier PostgreSQL###################
data_type character_data ######################## ##########ARRAY## #######element_types##### ######## ######USER-DEFINED## #########type_udt_name############ ##### ###########NULL###
character_maximum_length cardinal_number ##NULL###PostgreSQL###########################
character_octet_length cardinal_number ##NULL###PostgreSQL###########################
character_set_catalog sql_identifier PostgreSQL###################
character_set_schema sql_identifier PostgreSQL###################
character_set_name sql_identifier PostgreSQL###################
collation_catalog sql_identifier ##NULL###PostgreSQL###########################
collation_schema sql_identifier ##NULL###PostgreSQL###########################
collation_name sql_identifier

##
##NULL###PostgreSQL##########################
numeric_precision cardinal_number   ##NULL###PostgreSQL##########################
numeric_precision_radix cardinal_number   ##NULL###PostgreSQL##########################
numeric_scale cardinal_number   ##NULL###PostgreSQL##########################
datetime_precision cardinal_number   ##NULL###PostgreSQL##########################
interval_type character_data   ##NULL###PostgreSQL##########################
interval_precision cardinal_number   ##NULL###PostgreSQL##########################
type_udt_catalog sql_identifier   ########################## ############### ##########NULL###
type_udt_schema sql_identifier   ######################### ##########NULL###
type_udt_name sql_identifier   ############### ##########NULL###
scope_catalog sql_identifier   PostgreSQL###################
scope_schema sql_identifier   PostgreSQL###################
scope_name sql_identifier   PostgreSQL###################
maximum_cardinality cardinal_number   ##NULL###PostgreSQL###############
dtd_identifier sql_identifier   ## ########################   ######### ##
routine_body character_data   ###SQL#####SQL######EXTERNAL###
routine_definition character_data   ########### ######################NULL#### ###SQL###   ##routine_body#SQL############ ####PostgreSQL######################   ###########
external_name character_data   ###C#########################NULL### ####routine_definition######   ##############
external_language character_data   ##############
parameter_style character_data   ##GENERAL## ###SQL#######################PostgreSQL##########
is_deterministic yes_or_no   ##########SQL#######################YES######NO### ##########   PostgreSQL##############################
sql_data_access character_data

##
######SQL#####################MODIFIES### #####PostgreSQL############
is_null_call yes_or_no ############NULL########NULL######YES######NO### ###########NULL##
sql_path character_data PostgreSQL####################
schema_level_routine yes_or_no ##YES### ######################### ###PostgreSQL##############
max_dynamic_result_sets cardinal_number PostgreSQL####################
is_user_defined_cast yes_or_no PostgreSQL####################
is_implicitly_invocable yes_or_no PostgreSQL####################
security_type character_data #################INVOKER####################DEFINER###
to_sql_specific_catalog sql_identifier PostgreSQL####################
to_sql_specific_schema sql_identifier PostgreSQL####################
to_sql_specific_name sql_identifier PostgreSQL####################
as_locator yes_or_no PostgreSQL####################
created time_stamp PostgreSQL####################
last_altered time_stamp PostgreSQL####################
new_savepoint_level yes_or_no PostgreSQL####################
is_udt_dependent yes_or_no #####NO#######YES#PostgreSQL###################
result_cast_from_data_type character_data PostgreSQL####################
result_cast_as_locator yes_or_no PostgreSQL####################
result_cast_char_max_length cardinal_number PostgreSQL####################
result_cast_char_octet_length cardinal_number PostgreSQL####################
result_cast_char_set_catalog sql_identifier PostgreSQL####################
result_cast_char_set_schema sql_identifier PostgreSQL####################
result_cast_char_set_name sql_identifier PostgreSQL####################
result_cast_collation_catalog sql_identifier

##
##
PostgreSQL###################
result_cast_collation_schema sql_identifier
PostgreSQL###################
result_cast_collation_name sql_identifier
PostgreSQL###################
result_cast_numeric_precision cardinal_number
PostgreSQL###################
result_cast_numeric_precision_radix cardinal_number
PostgreSQL###################
result_cast_numeric_scale cardinal_number
PostgreSQL###################
result_cast_datetime_precision cardinal_number
PostgreSQL###################
result_cast_interval_type character_data
PostgreSQL###################
result_cast_interval_precision cardinal_number
PostgreSQL###################
result_cast_type_udt_catalog sql_identifier
PostgreSQL###################
result_cast_type_udt_schema sql_identifier
PostgreSQL###################
result_cast_type_udt_name sql_identifier
PostgreSQL###################
result_cast_scope_catalog sql_identifier
PostgreSQL###################
result_cast_scope_schema sql_identifier
PostgreSQL###################
result_cast_scope_name sql_identifier
PostgreSQL###################
result_cast_maximum_cardinality cardinal_number
PostgreSQL###################
result_cast_dtd_identifier sql_identifier
PostgreSQL###################

35.46. schemata

schemata###

#35.44 schemata##

##
##
catalog_name sql_identifier
##################################
schema_name sql_identifier
#########
schema_owner sql_identifier

##
#############
default_character_set_catalog sql_identifier   PostgreSQL###################
default_character_set_schema sql_identifier   PostgreSQL###################
default_character_set_name sql_identifier   PostgreSQL###################
sql_path character_data   PostgreSQL###################

35.47. sequences

sequences################################## ################################ ###################

#35.45 sequences##

##
sequence_catalog sql_identifier   ##################################
sequence_schema sql_identifier   #################
sequence_name sql_identifier   ##########
data_type character_data   ############
numeric_precision cardinal_number   ###################################### #############   numeric_precision_radix#########10###10########2###2###########   ##
numeric_precision_radix cardinal_number   #####numeric_precision###numeric_scale################# ##2###10   ###
numeric_scale cardinal_number   ###################################### #####################   numeric_precision_radix#########10###10########2###2###########   ##
start_value character_data   ###########
minimum_value character_data   ###########
maximum_value character_data   ###########
increment character_data   ###########
cycle_option yes_or_no   ############YES######NO###

SQL##

35.48. sql_features

sql_features#######SQL##############PostgreSQL######################## ##### D############### ##########################

#35.46 sql_features##

# #	##
feature_id character_data	###########
feature_name character_data	############
sub_feature_id character_data	###########################
sub_feature_name character_data	##############################
is_supported yes_or_no	#########PostgreSQL##################YES########NO###
is_verified_by character_data	##NULL### PostgreSQL##
comments character_data	#########################

35.49. sql_implementation_info

sql_implementation_info#######SQL################################## ###### #ODBC##################### ### ######## ################# ODBC###################

#35.47 sql_implementation_info##

# #	##
implementation_info_id character_data	###############
implementation_info_name character_data	################
integer_value cardinal_number	########### ####character_value######NULL###
character_value character_data	########### ####integer_value######NULL###
comments character_data	#####################

35.50. sql_parts

sql_parts#######SQL######PostgreSQL######################

#35.48 sql_parts##

##
feature_id character_data ##############
feature_name character_data ###########
is_supported yes_or_no PostgreSQL#########################YES######NO###
is_verified_by character_data ##NULL### PostgreSQL####################################
comments character_data #######################

35.51. sql_sizing

sql_sizing#######PostgreSQL######################## #######ODBC############# ######## ## ########################## ODBC##################

#35.49 sql_sizing##

##
sizing_id cardinal_number #############
sizing_name character_data ################
supported_value cardinal_number ################################### ########################NULL# ##
comments character_data #####################

35.52. table_constraints

table_constraints#######################SELECT#########################

#35.50 table_constraints##

##
constraint_catalog sql_identifier ###############################
constraint_schema sql_identifier ##############
constraint_name sql_identifier #######
table_catalog sql_identifier ################################

##	##
table_schema sql_identifier	################
table_name sql_identifier	#########
constraint_type character_data	#####[74.137931]######## CHECK#FOREIGN KEY#PRIMARY KEY#UNIQUE#### #### ###########:#####null#######################
is_deferrable yes_or_no	#########YES######NO#
initially_deferred yes_or_no	##################YES######NO#
enforced yes_or_no	PostgreSQL##################(#####YES##)#
nulls_distinct yes_or_no	################NULL#########YES####NULL###########NO########## #NULL###

35.53. table_privileges

table_privileges## ####################1#######

#35.51 table_privileges##

##	##
grantor sql_identifier	##############
grantee sql_identifier	################
table_catalog sql_identifier	################################
table_schema sql_identifier	################
table_name sql_identifier	#########
privilege_type character_data	###### SELECT#INSERT#UPDATE#DELETE#TRUNCATE#REFERENCES### #TRIGGER#######
is_grantable yes_or_no	############YES######NO###
with_hierarchy yes_or_no	SQL#####WITH HIERARCHY OPTION################################## PostgreSQL######SELECT##################SELECT####YES########NO###

35.54. tables

tables################################### ################################## #######################

#35.52 tables##

##	##
table_catalog	sql_identifier ##################################
table_schema	sql_identifier #################
table_name	sql_identifier ##########
table_type	character_data ########## ####################BASE TABLE######VIEW########FOREIGN# #######LOCAL TEMPORARY###
self_referencing_column_name	sql_identifier PostgreSQL####################
reference_generation	character_data PostgreSQL####################
user_defined_type_catalog	sql_identifier ## #####NULL###
user_defined_type_schema	sql_identifier ## #####NULL###
user_defined_type_name	sql_identifier ################################## #####NULL###
is_insertable_into	yes_or_no ###########YES######NO### ##################################
is_typed	yes_or_no #################YES#######NO###
commit_action	character_data ######

35.55. transforms

###transforms################################## ####################from SQL######to SQL#######1#######

#35.53 transforms##

##	##
udt_catalog	sql_identifier ######################################
udt_schema	sql_identifier #####################
udt_name	sql_identifier ##############
specific_catalog	sql_identifier ##################################
specific_schema	sql_identifier #################
specific_name	sql_identifier

##
########### ###35.45#########
group_name sql_identifier ##SQL###################################### PostgreSQL############## ##################### ################################
transform_type character_data FROM SQL###TO SQL

35.56. triggered_update_columns

####(UPDATE OF column1, column2##)#################### ##triggered_update_columns############## ########################### ###### ###############SELECT#####################

#35.54 triggered_update_columns##

##
trigger_catalog sql_identifier ################################
trigger_schema sql_identifier ###############
trigger_name sql_identifier ########
event_object_catalog sql_identifier ###
event_object_schema sql_identifier #########################
event_object_table sql_identifier ##################
event_object_column sql_identifier ###############

35.57. triggers

triggers##################################SELECT######################### ##########

#35.55 triggers##

##
trigger_catalog sql_identifier ################################
trigger_schema sql_identifier ###############
trigger_name sql_identifier ########
event_manipulation character_data

##
##
############ #INSERT#UPDATE####DELETE####
event_object_catalog sql_identifier ###
event_object_schema sql_identifier #########################
event_object_table sql_identifier ###################
action_order cardinal_number ########event_manipulation#action_timing#action_orientation# ############ PostgreSQL##################################
action_condition character_data ####WHEN########NULL## (######################NULL##)#
action_statement character_data ############## ######EXECUTE FUNCTION *function*(...)####
action_orientation character_data ###################### #ROW####STATEMENT####
action_timing character_data ########## #BEFORE#AFTER####INSTEAD OF####
action_reference_old_table sql_identifier #old################NULL###
action_reference_new_table sql_identifier #new################NULL###
action_reference_old_row sql_identifier PostgreSQL###################
action_reference_new_row sql_identifier PostgreSQL###################
created time_stamp PostgreSQL###################

PostgreSQL###########SQL####2############################## 1####PostgreSQL### ## ###################1########### ############# #trigger_catalog#trigger_schema########################### ### 2####PostgreSQL#########################ON INSERT OR UPDATE#### #####SQL##1########### ##################################1################ ############ ####2#######triggers############SQL#####(trigger_catalog, trigger_schema, trigger_name)#####(trigger_catalog, trigger_schema, event_object_table, trigger_name, event_manipulation)###### ######## SQL#########################1##

##

PostgreSQL 9.1 ############ action_timing# action_reference_old_table# action_reference_new_table# action_reference_old_row# action_reference_new_row ##### condition_timing# condition_reference_old_table# condition_reference_new_table# condition_reference_old_row# condition_reference_new_row ############### SQL: 1999############### #SQL:2003##########

35.58. udt_privileges

udt_privileges#######################################USAGE######## ######### ################ #######################35.60########### ###########35.59######### ###

#35.56 udt_privileges##

##	##
grantor sql_identifier	###############
grantee sql_identifier	#################
udt_catalog sql_identifier	##############################
udt_schema sql_identifier	###########
udt_name sql_identifier	####
privilege_type character_data	##TYPE USAGE
is_grantable yes_or_no	############YES######NO###

35.59. usage_privileges

usage_privileges##USAGE######## #########PostgreSQL## ################### ###########1#######

PostgreSQL#######################################PUBLIC###############USAGE# ####### ############################

PostgreSQL#########USAGE####SELECT#UPDATE#################################### ########

#35.57 usage_privileges##

##	##
grantor sql_identifier	###############
grantee sql_identifier	#################
object_catalog sql_identifier	##################################
object_schema sql_identifier	#################################
object_name sql_identifier	############

##	##
object_type character_data	COLLATION###DOMAIN###FOREIGN DATA WRAPPER###FOREIGN SERVER## #SEQUENCE
privilege_type character_data	##USAGE###
is_grantable yes_or_no	############YES######NO###

35.60. user_defined_types

user_defined_types#################################### ################### ####################################

SQL##########################PostgreSQL#######################PostgreSQL###### ####### #######user_defined_type_category################## PostgreSQL##### ################################ #############35.23##########

#35.58 user_defined_types##

##	##
user_defined_type_catalog sql_identifier	##############################
user_defined_type_schema sql_identifier	#############
user_defined_type_name sql_identifier	####
user_defined_type_category character_data	#####STRUCTURED###
is_instantiable yes_or_no	PostgreSQL###################
is_final yes_or_no	PostgreSQL###################
ordering_form character_data	PostgreSQL###################
ordering_category character_data	PostgreSQL###################
ordering_routine_catalog sql_identifier	PostgreSQL###################
ordering_routine_schema sql_identifier	PostgreSQL###################
ordering_routine_name sql_identifier	PostgreSQL###################
reference_type character_data	PostgreSQL###################
data_type character_data	PostgreSQL###################
character_maximum_length cardinal_number	

##
PostgreSQL###################
character_octet_length cardinal_number PostgreSQL###################
character_set_catalog sql_identifier PostgreSQL###################
character_set_schema sql_identifier PostgreSQL###################
character_set_name sql_identifier PostgreSQL###################
collation_catalog sql_identifier PostgreSQL###################
collation_schema sql_identifier PostgreSQL###################
collation_name sql_identifier PostgreSQL###################
numeric_precision cardinal_number PostgreSQL###################
numeric_precision_radix cardinal_number PostgreSQL###################
numeric_scale cardinal_number PostgreSQL###################
datetime_precision cardinal_number PostgreSQL###################
interval_type character_data PostgreSQL###################
interval_precision cardinal_number PostgreSQL###################
source_dtd_identifier sql_identifier PostgreSQL###################
ref_dtd_identifier sql_identifier PostgreSQL###################

35.61. user_mapping_options

user_mapping_options## ############ ##

#35.59 user_mapping_options##

##
authorization_identifier sql_identifier ####################public###PUBLIC###
foreign_server_catalog sql_identifier ##
foreign_server_name sql_identifier ####################
option_name sql_identifier

##
##########
option_value character_data ######### ##########################PUBLIC####################### ###############NULL###### #####################################

35.62. user_mappings

user_mappings################################## ##########################
###

#35.60 user_mappings##

##
authorization_identifier sql_identifier ####################public###PUBLIC###
foreign_server_catalog sql_identifier ##
foreign_server_name sql_identifier #####################

35.63. view_column_usage

view_column_usage######################SELECT################# #########
#########################

> ##
>
> ################ ##################

#35.61 view_column_usage##

##
view_catalog sql_identifier ################################
view_schema sql_identifier ###############
view_name sql_identifier ########
table_catalog sql_identifier ################################ ###############
table_schema sql_identifier ##############################
table_name sql_identifier ######################
column_name sql_identifier

##
###############

35.64. view_routine_usage

view_routine_usage#######################SELECT#########################
######## ##########################

#35.62 view_routine_usage##

##
table_catalog sql_identifier ################################
table_schema sql_identifier ###############
table_name sql_identifier ########
specific_catalog sql_identifier ##################################
specific_schema sql_identifier ################
specific_name sql_identifier ########### ###35.45#########

35.65. view_table_usage

view_table_usage#######################SELECT#################### #######
##########################

> ##
>
> ############### ##################

#35.63 view_table_usage##

##
view_catalog sql_identifier ################################
view_schema sql_identifier ###############
view_name sql_identifier ########
table_catalog sql_identifier ########################## ###############
table_schema sql_identifier #########################

##
table_name sql_identifier ##################

35.66. views

views################################ ##################################### ##################

#35.64 views##

##
table_catalog sql_identifier ################################
table_schema sql_identifier ###############
table_name sql_identifier ########
view_definition character_data ############### ######################NULL####
check_option character_data ####CHECK OPTION##########CASCADED###LOCAL######NONE###
is_updatable yes_or_no ########UPDATE###DELETE######YES######NO###
is_insertable_into yes_or_no ########INSERT######YES######NO###
is_trigger_updatable yes_or_no ####INSTEAD OF UPDATE##############YES######NO###
is_trigger_deletable yes_or_no ####INSTEAD OF DELETE##############YES######NO###
is_trigger_insertable_into yes_or_no ####INSTEAD OF INSERT##############YES######NO###

V.

#####[88.161994]## ####### ##PostgreSQL################################## #########PostgreSQL######################## ######## ## ################################36######## ######################## ## ###### #########PostgreSQL## ############PostgreSQL### ########################36###### ######### ##############

##

36. SQL### 1041
36.1. ###### 1041
36.2. PostgreSQL###### 1041
36.2.1. ### 1041
36.2.2. ##### 1041
36.2.3. #### 1041
36.2.4. ###### 1042
36.2.5. ### 1042
36.3. ####### 1043
36.4. ########### 1044
36.5. ########SQL### 1044
36.5.1. SQL###### 1045
36.5.2. ########SQL## 1045
36.5.3. ########SQL## 1047
36.5.4. ##########SQL## 1049
36.5.5. ##########SQL###### 1050
36.5.6. ########SQL## 1050
36.5.7. ############SQL## 1051
36.5.8. ###########SQL## 1052
36.5.9. #####SQL## 1052
36.5.10. TABLE###SQL## 1055
36.5.11. ##SQL## 1056
36.5.12. #######SQL## 1057
36.6. ########## 1058
36.7. ######## 1058
36.8. ######## 1060
36.9. #### 1060
36.10. C#### 1060
36.10.1. ##### 1060
36.10.2. C########### 1061
36.10.3. Version 1 ###### 1063
36.10.4. ###### 1066
36.10.5. ##################### 1067
36.10.6. ##### 1068
36.10.7. ######### 1069
36.10.8. ##### 1071
36.10.9. ########## 1076
36.10.10. Shared Memory 1077
36.10.11. LWLocks 1078
36.10.12. Custom Wait Events 1079
36.10.13. Injection Points 1079
36.10.14. ####C++### 1080
36.11. ########### 1080
36.12. ######## 1081
36.12.1. ####### 1082
36.12.2. ############ 1083
36.12.3. ####### 1085
36.12.4. #### 1085
36.12.5. ######## 1086
36.13. ####### 1086
36.13.1. TOAST### 1089
36.14. ######### 1090
36.15. ############ 1090
36.15.1. COMMUTATOR 1091
36.15.2. NEGATOR 1091

36.15.3. RESTRICT 1091
36.15.4. JOIN 1092
36.15.5. HASHES 1092
36.15.6. MERGES 1093
36.16. ################### 1093
36.16.1. ################## 1093
36.16.2. ################ 1094
36.16.3. ################### 1095
36.16.4. # 1098
36.16.5. ########## 1099
36.16.6. ################## 1102
36.16.7. ####### 1103
36.16.8. ############ 1103
36.17. ######################## 1104
36.17.1. ####### 1104
36.17.2. ####### 1106
36.17.3. ######## 1107
36.17.4. ##### 1108
36.17.5. ######################## 1108
36.17.6. ################################ 1109
36.17.7. #### 1110
36.18. ###### 1110
37. ### 1115
37.1. ######## 1115
37.2. ######### 1117
37.3. C########### 1117
37.4. ######## 1120
38. ####### 1124
38.1. ############ 1124
38.2. ############## 1125
38.3. C################### 1128
38.4. ############ 1129
38.5. ################## 1130
38.6. A Database Login Event Trigger Example 1131
39. ####### 1133
39.1. ########## 1133
39.2. ########### 1134
39.2.1. SELECT###### 1134
39.2.2. #SELECT######## 1138
39.2.3. PostgreSQL########## 1139
39.2.4. ########## 1139
39.3. ########### 1140
39.4. INSERT#UPDATE#DELETE######## 1143
39.4.1. ######## 1143
39.4.2. ####### 1147
39.5. ###### 1152
39.6. ############# 1153
39.7. ####### 1154
40. ##### 1156
40.1. ############ 1156
41. PL/pgSQL — SQL##### 1158
41.1. ## 1158
41.1.1. PL/pgSQL########## 1158
41.1.2. ############## 1158
41.2. PL/pgSQL### 1159
41.3. ## 1160
41.3.1. ####### 1161
41.3.2. ALIAS 1164

41.3.3. ##### 1164
41.3.4. ## 1165
41.3.5. ##### 1165
41.3.6. PL/pgSQL##### 1165
41.4. # 1166
41.5. ##### 1167
41.5.1. ## 1167
41.5.2. SQL####### 1167
41.5.3. 1############## 1168
41.5.4. ######### 1170
41.5.5. ########## 1172
41.5.6. ######### 1173
41.6. #### 1174
41.6.1. ####### 1174
41.6.2. ########### 1176
41.6.3. ########### 1176
41.6.4. #### 1176
41.6.5. ###### 1179
41.6.6. ############## 1181
41.6.7. ##### 1183
41.6.8. ###### 1183
41.6.9. ######### 1186
41.7. #### 1187
41.7.1. ######### 1187
41.7.2. ####### 1187
41.7.3. ####### 1189
41.7.4. ############# 1191
41.8. ########## 1192
41.9. ######### 1193
41.9.1. ############ 1193
41.9.2. ###### 1194
41.10. ##### 1195
41.10.1. ########### 1195
41.10.2. ########## 1202
41.11. PL/pgSQL### 1202
41.11.1. #### 1202
41.11.2. ######## 1204
41.12. PL/pgSQL########### 1205
41.12.1. ###### 1205
41.12.2. ################### 1207
41.13. Oracle PL/SQL##### 1208
41.13.1. ### 1208
41.13.2. ######## 1213
41.13.3. ## 1214
42. PL/Tcl — Tcl##### 1217
42.1. ## 1217
42.2. PL/Tcl##### 1217
42.3. PL/Tcl######### 1219
42.4. PL/Tcl############ 1219
42.5. PL/Tcl############# 1219
42.6. PL/Tcl###### 1221
42.7. PL/Tcl############# 1223
42.8. PL/Tcl###### 1223
42.9. PL/Tcl################## 1224
42.10. ########## 1225
42.11. PL/Tcl### 1225
42.12. Tcl####### 1226
43. PL/Perl — Perl##### 1227

43.1. PL/Perl##### 1227
43.2. PL/Perl######## 1231
43.3. ###### 1231
43.3.1. PL/Perl############# 1231
43.3.2. PL/Perl########## 1235
43.4. PL/Perl########### 1236
43.5. #####PL/Perl#########PL/Perl 1236
43.6. PL/Perl### 1237
43.7. PL/Perl####### 1239
43.8. PL/Perl### 1239
43.8.1. ## 1239
43.8.2. ############ 1240
44. PL/Python — Python##### 1241
44.1. PL/Python## 1241
44.2. #### 1242
44.2.1. ######### 1242
44.2.2. Null#None 1243
44.2.3. ###### 1243
44.2.4. ### 1244
44.2.5. ####### 1246
44.3. ###### 1247
44.4. ######### 1247
44.5. ##### 1247
44.6. ########## 1248
44.6.1. ############ 1248
44.6.2. ###### 1250
44.7. ############# 1251
44.7.1. ################### 1251
44.8. ########## 1252
44.9. ######### 1252
44.10. Python 2#Python 3 1253
44.11. #### 1253
45. ################# 1255
45.1. ######### 1255
45.2. ############# 1294
45.3. ##### 1303
45.4. ########## 1313
45.5. ######### 1316
45.6. # 1316
46. ################ 1320
47. ########### 1323
47.1. ############# 1323
47.2. ################# 1326
47.2.1. ########### 1326
47.2.2. ############ 1326
47.2.3. Replication Slot Synchronization 1327
47.2.4. ####### 1328
47.2.5. ############### 1328
47.3. ########################### 1328
47.4. ###########SQL####### 1328
47.5. ######################## 1329
47.6. ################### 1329
47.6.1. ##### 1329
47.6.2. ## 1330
47.6.3. ##### 1330
47.6.4. ############# 1330
47.6.5. ###### 1335
47.7. ################ 1335

47.8. ############################## 1336
47.8.1. ## 1336
47.8.2. ## 1336
47.9. ################################ 1336
47.10. ##############2##### 1337
48. ############# 1339
49. ########## 1340
49.1. ##### 1340
49.2. ################ 1340
49.2.1. ############# 1340
49.2.2. ########## 1340
49.2.3. ########### 1341
49.2.4. ############# 1341

#36# SQL###

#############PostgreSQL#SQL#########################

- ###36.3####
- ###36.12####
- #####36.13####
- ####36.14####
- #############36.16####
- ##############36.17####

36.1.

PostgreSQL################################## ################################ ## ######### ##################### ##############################DBMS################## PostgreSQL##############################PostgreSQL######################### #### ## ######################PostgreSQL ########################PostgreSQL##################### ###################### ##############################DBMS##################################

###PostgreSQL## ######################### ########################PostgreSQL############## SQL######################### ## #########################PostgreSQL######################################

36.2. PostgreSQL######

PostgreSQL################################

36.2.1.

`integer` #####SQL################ ####C################## ################ ########### PostgreSQL## ############## ##########8##########

##(enum)################### ########################SQL################### ### #######8.7#########

36.2.2.

PostgreSQL################################### ################

##################### ################################### ################ ### ######################### #########8.15#########

################################ ############################CREATE TYPE#### ############ ############################## ######################### ######## ##8.16#########

################################## ################################ ######## ##8.17#########

36.2.3.

#####SQL#####CREATE DOMAIN######### #########8.18#########

36.2.4.

#################### ################################### ###################### ##### ################################### # 8.27##############

36.2.5.

##################################### ## ### ##### 36.1###### ########36.5.11######

#36.1 ###

##	#	##
anyelement	Simple	########################
anyarray	Simple	#########################
anynonarray	Simple	############################
anyenum	Simple	###########8.7#################
anyrange	Simple	##############8.17############## ###
anymultirange	Simple	################8.17########### ######
anycompatible	Common	############################## ##################
anycompatiblearray	Common	############################## ###################
anycompatiblenonarray	Common	############################## ######################
anycompatiblerange	Common	############################## ###################
anycompatiblemultirange	Common	############################## #####################

2################### ######################## ###

#####simple#########################

anyelement##1#################### ############ anyarray### ######## anyrange################################### anymultirange######## ##

####anyarray########anyelement################anyarray################# #anyelement#################### anynonarray########################### ######anyelement############# anyenum################################ ##anyelement#############

###anyrange########anyelement####anyarray################anyrange###### ###############anyelement#############anyarray################### anymultirange##############################anyrange############## #anyelement###anyarray######################

######2### ##
##equal(anyelement, anyelement)#########2#######################2#######
###

#######################1#################### ################################
##################subscript(anyarray, integer) returns
anyelement#################### ##
#######f(anyarray) returns anyenum#######################

####anyarray#anyelement######
########### ###anyrange######anyrange########### anyarray####anyelement#
############ ##############################

anynonarray##anyenum########################## ###anyelement############
######### ####f(anyelement, anyenum)#############f(anyenum, anyenum)####
######### #######################

#####common###################simple######################### ###############
########################## ####UNION############10.5############## ######
##anycompatible#anycompatiblenonarray########anycompatiblearray#####
###anycompatiblerange######anycompatiblemultirange######
##anycompatiblerange################## anycompatiblenonarray###########
################ ##############anycompatible###anycompatiblenonarray####
#################anycompatiblearray##########################

#############################anycompatiblerange###/##
#anycompatiblemultirange################################/###############
anyrange##
#anymultirange#####anycompatiblerange###anymultirange##############
##anycompatiblerange###anycompatiblemultirange#############

anycompatibleenum################ ##
#####################################

#simple###common########################## #############

```
CREATE FUNCTION myfunc(a anyelement, b anyelement,
                       c anycompatible, d anycompatible)
RETURNS anycompatible AS ...
```

#################2######################## ###2##############################2#
##################### ######2##############

#########36.5.6############################## ########VARIADIC anyarray##
#VARIADIC anycompatiblearray############## ##########################
#anynonarray###anycompatiblenonarray###########################

36.3.

PostgreSQL#4############

- ##########SQL##########36.5#
- ########PL/pgSQL#PL/Tcl############36.8#
- #####36.9#
- C#####36.10#

##########################
################################

################################### ################ ####################

SQL########################## SQL############################

#####################CREATE FUNCTION######################## ######### PostgreSQL#######src/tutorial########funcs.sql#funcs.c######

36.4.

######################### #############

- #######CREATE FUNCTION######CREATE PROCEDURE##########
- ####################CREATE PROCEDURE##RETURNS######## ################## ##########################
- #########DML##########################CALL##################
- ####CALL## ########################## ###########
- #######(strictness)############################ ############################## ###################

###

########################## ######################ALTER ROUTINE#DROP ROUTINE############ #######CREATE ROUTINE##################

36.5. ########SQL###

SQL######SQL############################### ############################### ##### ##################ORDER BY################################## ########### #######NULL#######

####SQL####SETOF *sometype*##################RETURNS TABLE(*columns*)###### ############################ ########################### ###########

#####[87.469880]#SQL################SQL############### ################### ## #void####################SELECT####RETURNING####INSERT#UPDATE### #DELETE########## ######SQL#################SQL############### ### ################ ###void##################SELECT####RETURNING### #INSERT#UPDATE#DELETE#MERGE##########

SQL################################## SELECT######################## ##INSERT#UPDATE#DELETE###MERGE######SQL############### #SQL### #COMMIT#SAVEPOINT##################VACUUM######################## ######### ######################SELECT####RETURNING############### ################## #######SQL##############void################### ##########emp############### ########

```
CREATE FUNCTION clean_emp() RETURNS void AS '
    DELETE FROM emp
        WHERE salary < 0;
' LANGUAGE SQL;

SELECT clean_emp();

 clean_emp
-----------
```

```
(1 row)
```

##

```
CREATE PROCEDURE clean_emp() AS '
    DELETE FROM emp
        WHERE salary < 0;
' LANGUAGE SQL;

CALL clean_emp();
```

###########void############################ ################################## ########## ###void######PostgreSQL#################SQL#####

> ##
>
> SQL########################## SQL#######################CREATE TABLE## ###### ##CREATE TABLE foo (...); INSERT INTO foo VALUES(...);####SQL# ########################INSERT##############foo############ ######### SQL#######PL/pgSQL##########

CREATE FUNCTION################################## ##################### #4.1.2.4######### ####################################'##########\########### ##################4.1.2.1#####

36.5.1. SQL######

SQL############################## ##############

##########SQL############## ############### ############*function_name.argument_name*################## ########## ############################## SQL##################################

#################$*n*################## ####$1##1######$2##2########## ##### ##############################

#########*argname.fieldname*#$1.*fieldname*######################### ###### ####################################

SQL################################## ########

```
INSERT INTO mytable VALUES ($1);
```

###################

```
INSERT INTO $1 VALUES (42);
```

> ##
>
> SQL#######################PostgreSQL 9.2######### #################$*n*### ############

36.5.2. ########SQL##

#####SQL############integer##############

```
CREATE FUNCTION one() RETURNS integer AS $$
    SELECT 1 AS result;
$$ LANGUAGE SQL;


-- ##############
CREATE FUNCTION one() RETURNS integer AS '
    SELECT 1 AS result;
' LANGUAGE SQL;

SELECT one();

 one
-----
   1
```

##################result#################### ##############################
##########result#####one###############

###########SQL####################

```
CREATE FUNCTION add_em(x integer, y integer) RETURNS integer AS $$
    SELECT x + y;
$$ LANGUAGE SQL;

SELECT add_em(1, 2) AS answer;

 answer
--------
      3
```

##############################

```
CREATE FUNCTION add_em(integer, integer) RETURNS integer AS $$
    SELECT $1 + $2;
$$ LANGUAGE SQL;

SELECT add_em(1, 2) AS answer;

 answer
--------
      3
```

################## #######################

```
CREATE FUNCTION tf1 (accountno integer, debit numeric) RETURNS numeric AS
 $$
    UPDATE bank
        SET balance = balance - debit
        WHERE accountno = tf1.accountno;
    SELECT 1;
$$ LANGUAGE SQL;
```

#######################17##100##############

```
SELECT tf1(17, 100.0);
```

#############accountno##########bank############## UPDATE######
##accountno#bank.accountno###################tf1.accountno###########
################################

#############1##################################

```
CREATE FUNCTION tf1 (accountno integer, debit numeric) RETURNS numeric AS
 $$
    UPDATE bank
        SET balance = balance - debit
        WHERE accountno = tf1.accountno;
    SELECT balance FROM bank WHERE accountno = tf1.accountno;
$$ LANGUAGE SQL;
```

#################### #####RETURNING#################

```
CREATE FUNCTION tf1 (accountno integer, debit numeric) RETURNS numeric AS
 $$
    UPDATE bank
        SET balance = balance - debit
        WHERE accountno = tf1.accountno
    RETURNING balance;
$$ LANGUAGE SQL;
```

SQL#####SELECT##RETURNING#######################PostgreSQL################
###################### ######################### #######add_em#######float8#
########### ##############

```
CREATE FUNCTION add_em(integer, integer) RETURNS float8 AS $$
    SELECT $1 + $2;
$$ LANGUAGE SQL;
```

integer###float8############## #############10####CREATE CAST#########

36.5.3. ########SQL##

####emp########
#####double_salary################
##############

```
CREATE TABLE emp (
    name        text,
    salary      numeric,
    age         integer,
    cubicle     point
);

INSERT INTO emp VALUES ('Bill', 4200, 45, '(2,1)');

CREATE FUNCTION double_salary(emp) RETURNS numeric AS $$
    SELECT $1.salary * 2 AS salary;
$$ LANGUAGE SQL;

SELECT name, double_salary(emp.*) AS dream
    FROM emp
    WHERE emp.cubicle ~= point '(2,1)';

 name | dream
------+-------
 Bill |  8400
```

$1.salary################1####################### ###*table_name*.*####
#SELECT###
#########################

```
SELECT name, double_salary(emp) AS dream
    FROM emp
    WHERE emp.cubicle ~= point '(2,1)';
```

########################### ###########################8.16.5##########

############################## ###ROW############ ############################

```
SELECT name, double_salary(ROW(name, salary*1.1, age, cubicle)) AS dream
    FROM emp;
```

#################### ######emp##############

```
CREATE FUNCTION new_emp() RETURNS emp AS $$
    SELECT text 'None' AS name,
        1000.0 AS salary,
        25 AS age,
        point '(2,2)' AS cubicle;
$$ LANGUAGE SQL;
```

##

##########2#################

- ## ################################
- ################################### ####################

```
  ERROR:  return type mismatch in function declared to return emp
  DETAIL:  Final statement returns text instead of point at column 4.
```

  ##

#######################

```
CREATE FUNCTION new_emp() RETURNS emp AS $$
    SELECT ROW('None', 1000.0, 25, '(2,2)')::emp;
$$ LANGUAGE SQL;
```

###################SELECT######## ##
##################################### #######################################
######################### #########################

#######################

```
SELECT new_emp();

         new_emp
--------------------------
 (None,1000.0,25,"(2,2)")
```

#######################

```
SELECT * FROM new_emp();

 name | salary | age | cubicle
------+--------+-----+---------
 None | 1000.0 |  25 | (2,2)
```

2##########36.5.8###########

#####################1############################ #######################

```
SELECT (new_emp()).name;

 name
------
 None
```

########################### ##########################

```
SELECT new_emp().name;
ERROR:  syntax error at or near "."
LINE 1: SELECT new_emp().name;
                        ^
```

#########################

```
SELECT name(new_emp());

 name
------
 None
```

8.16.5#########################

###

```
CREATE FUNCTION getname(emp) RETURNS text AS $$
    SELECT $1.name;
$$ LANGUAGE SQL;

SELECT getname(new_emp());
 getname
---------
 None
(1 row)
```

36.5.4. ##########SQL##

##########

```
CREATE FUNCTION add_em (IN x int, IN y int, OUT sum int)
AS 'SELECT x + y'
LANGUAGE SQL;

SELECT add_em(3,7);
 add_em
--------
     10
(1 row)
```

36.5.2####add_em############## ## ######
####

```
CREATE FUNCTION sum_n_product (x int, y int, OUT sum int, OUT product int)
AS 'SELECT x + y, x * y'
LANGUAGE SQL;
```

```
 SELECT * FROM sum_n_product(11,42);
 sum | product
-----+---------
  53 |     462
(1 row)
```

############################ ####################

```
CREATE TYPE sum_prod AS (sum int, product int);

CREATE FUNCTION sum_n_product (int, int) RETURNS sum_prod
AS 'SELECT $1 + $2, $1 * $2'
LANGUAGE SQL;
```

##################################### ### ############### ################################

SQL## PostgreSQL############## ########################## ## #### ######################

```
DROP FUNCTION sum_n_product (x int, y int, OUT sum int, OUT product int);
DROP FUNCTION sum_n_product (int, int);
```

########IN########OUT#INOUT####VARIADIC########### INOUT################## ##################################### VARIADIC############################## #####

36.5.5. ##########SQL######

CALL############################## ######

```
CREATE PROCEDURE tp1 (accountno integer, debit numeric, OUT new_balance
 numeric) AS $$
    UPDATE bank
        SET balance = balance - debit
        WHERE accountno = tp1.accountno
    RETURNING balance;
$$ LANGUAGE SQL;
```

###############OUT################### #####NULL######

```
CALL tp1(17, 100.0, NULL);
```

##

PL/pgSQL###############NULL################################## ###41.6.3###### ##

36.5.6. ########SQL##

#########################SQL######################## ######################## ### ##########VARIADIC############ ############################## #######

```
CREATE FUNCTION mleast(VARIADIC arr numeric[]) RETURNS numeric AS $$
    SELECT min($1[i]) FROM generate_subscripts($1, 1) g(i);
```

```
$$ LANGUAGE SQL;

SELECT mleast(10, -1, 5, 4.4);
 mleast
--------
     -1
(1 row)
```

###VARIADIC#################################1################

```
SELECT mleast(ARRAY[10, -1, 5, 4.4]);    -- ######
```

######################### ################### VARIADIC#########################
##################

#####variadic######################### ###variadic################################ #
##################################variadic#################10.3########## #######
##VARIADIC############

```
SELECT mleast(VARIADIC ARRAY[10, -1, 5, 4.4]);
```

######variadic######################### ############################ VARIADIC#####
####################

#####VARIADIC#########variadic#########################

```
SELECT mleast(VARIADIC ARRAY[]::numeric[]);
```

variadic###########1######################SELECT mleast()################ #####
#################mleast##############2###########

variadic## ###########4.3########
######################### ####VARIADIC############ ###############

```
SELECT mleast(VARIADIC arr => ARRAY[10, -1, 5, 4.4]);
```

##########

```
SELECT mleast(arr => 10);
SELECT mleast(arr => ARRAY[10, -1, 5, 4.4]);
```

36.5.7. #############SQL##

################################## ### ##
########
######################
##10.3##########

##########

```
CREATE FUNCTION foo(a int, b int DEFAULT 2, c int DEFAULT 3)
RETURNS int
LANGUAGE SQL
AS $$
    SELECT $1 + $2 + $3;
$$;

SELECT foo(10, 20, 30);
```

```
 foo
-----
  60
(1 row)

SELECT foo(10, 20);
 foo
-----
  33
(1 row)

SELECT foo(10);
 foo
-----
  15
(1 row)


SELECT foo();  -- ##################
ERROR:  function foo() does not exist
```

=###DEFAULT#####################

36.5.8. ##########SQL##

####SQL#########FROM######################### ###########################1### ############ ##1##########

##########

```
CREATE TABLE foo (fooid int, foosubid int, fooname text);
INSERT INTO foo VALUES (1, 1, 'Joe');
INSERT INTO foo VALUES (1, 2, 'Ed');
INSERT INTO foo VALUES (2, 1, 'Mary');

CREATE FUNCTION getfoo(int) RETURNS foo AS $$
    SELECT * FROM foo WHERE fooid = $1;
$$ LANGUAGE SQL;

SELECT *, upper(fooname) FROM getfoo(1) AS t1;

 fooid | foosubid | fooname | upper
-------+----------+---------+-------
     1 |        1 | Joe     | JOE
(1 row)
```

##

############1################## ###SETOF############ ################

36.5.9. #####SQL##

SQL###SETOF *sometype*###

###########FROM################ ### #### ####foo#######################

```
CREATE FUNCTION getfoo(int) RETURNS SETOF foo AS $$
    SELECT * FROM foo WHERE fooid = $1;
$$ LANGUAGE SQL;

SELECT * FROM getfoo(1) AS t1;
```

```
#############


 fooid | foosubid | fooname
-------+----------+---------
     1 |        1 | Joe
     1 |        2 | Ed
(2 rows)

########################################


CREATE TABLE tab (y int, z int);
INSERT INTO tab VALUES (1, 2), (3, 4), (5, 6), (7, 8);

CREATE FUNCTION sum_n_product_with_tab (x int, OUT sum int, OUT product
 int)
RETURNS SETOF record
AS $$
    SELECT $1 + tab.y, $1 * tab.y FROM tab;
$$ LANGUAGE SQL;

SELECT * FROM sum_n_product_with_tab(10);
 sum | product
-----+---------
  11 |      10
  13 |      30
  15 |      50
  17 |      70
(4 rows)

############1#####################RETURNS SETOF record################ ######
##############record######################

################################################################################
####### #########7.2.1.5#####LATERAL############## ############################
###


SELECT * FROM nodes;
   name    | parent
-----------+--------
 Top       |
 Child1    | Top
 Child2    | Top
 Child3    | Top
 SubChild1 | Child1
 SubChild2 | Child1
(6 rows)

CREATE FUNCTION listchildren(text) RETURNS SETOF text AS $$
    SELECT name FROM nodes WHERE parent = $1
$$ LANGUAGE SQL STABLE;

SELECT * FROM listchildren('Top');
 listchildren
--------------
 Child1
 Child2
 Child3
(3 rows)

SELECT name, child FROM nodes, LATERAL listchildren(name) AS child;
  name  |   child
--------+-----------
```

```
 Top    | Child1
 Top    | Child2
 Top    | Child3
 Child1 | SubChild1
 Child1 | SubChild2
(5 rows)
```

############################ ##

################################## ##
############## #######################

```
SELECT listchildren('Top');
 listchildren
--------------
 Child1
 Child2
 Child3
(3 rows)

SELECT name, listchildren(name) FROM nodes;
  name  | listchildren
--------+--------------
 Top    | Child1
 Top    | Child2
 Top    | Child3
 Child1 | SubChild1
 Child1 | SubChild2
(5 rows)
```

###SELECT#####Child2#Child3######################### ####listchildren#### ############################ LATERAL################################

##################PostgreSQL#############LATERAL FROM################### ###

```
SELECT x, generate_series(1,5) AS g FROM tab;
```

###########

```
SELECT x, g FROM tab, LATERAL generate_series(1,5) AS g;
```

#########g#####tab#LATERAL############################g################### ######### ########################### ###############FROM#################### ######FROM################################

####################2################LATERAL ROWS FROM(...) FROM# ###################### ####################################2################ ## ##NULL#########1################ ## ## ################

#########FROM############################## ################LATERAL ROWS FROM(...)################ ####

```
SELECT srf1(srf2(x), srf3(y)), srf4(srf5(z)) FROM tab;
```

##########srf2#srf3#srf5#tab#################################### #srf1#srf4##############

########CASE#COALESCE###################### ##################

```
SELECT x, CASE WHEN x > 0 THEN generate_series(1, 5) ELSE 0 END FROM tab;
```

```
####x        >        0######5##############1##############################
##generate_series(1, 5)#CASE##########LATERAL FROM###################5#
########### ##################################
```

```
##

#####[87.658802]#############RETURNING###INSERT#UPDATE####DELETE####
#####SETOF#############################################################
RETURNING#################################################################
###########                                ################RETURNING##
#INSERT#UPDATE#DELETE#MERGE##############SETOF####################
##############################  RETURNING##############################
#####################################
```

```
##

PostgreSQL   10#############2##############################################
##############################################  #######################
##########################1##################################   #######
#CASE########################################  PostgreSQL###############
####################################LATERAL############  ###############
############################################ ###


SELECT x, CASE WHEN y > 0 THEN generate_series(1, z) ELSE 5 END FROM
 tab;
```

```
############


CREATE FUNCTION case_generate_series(cond bool, start int, fin int,
 els int)
  RETURNS SETOF int AS $$
BEGIN
  IF cond THEN
    RETURN QUERY SELECT generate_series(start, fin);
  ELSE
    RETURN QUERY SELECT els;
  END IF;
END$$ LANGUAGE plpgsql;

SELECT x, case_generate_series(y > 0, 1, z, 5) FROM tab;
```

```
######PostgreSQL####################
```

36.5.10. TABLE###SQL##

```
############################## RETURNS   TABLE(columns)########### #######
#OUT###############SETOF record######################SETOF################
### #######SQL#################SETOF######################
```

```
######################


CREATE FUNCTION sum_n_product_with_tab (x int)
RETURNS TABLE(sum int, product int) AS $$
    SELECT $1 + tab.y, $1 * tab.y FROM tab;
```

```
$$ LANGUAGE SQL;
```

RETURNS TABLE#########OUT###INOUT############# ########TABLE#############
##

36.5.11. ##SQL##

SQL####36.2.5#################### ###make_array########2##################

```
CREATE FUNCTION make_array(anyelement, anyelement) RETURNS anyarray AS $$
    SELECT ARRAY[$1, $2];
$$ LANGUAGE SQL;

SELECT make_array(1, 2) AS intarray, make_array('a'::text, 'b') AS
 textarray;
 intarray | textarray
----------+-----------
 {1,2}    | {a,b}
(1 row)
```

'a'::text################text######################## ####################
######unknown################unknown###############
#######

```
ERROR:  could not determine polymorphic type because input has type
 unknown
```

######make_array##################2################ #################### ##
##################

```
SELECT make_array(1, 2.5) AS numericarray;
ERROR:  function make_array(integer, numeric) does not exist
```

############################## #########################

```
CREATE FUNCTION make_array2(anycompatible, anycompatible)
RETURNS anycompatiblearray AS $$
    SELECT ARRAY[$1, $2];
$$ LANGUAGE SQL;

SELECT make_array2(1, 2.5) AS numericarray;
 numericarray
--------------
 {1,2.5}
(1 row)
```

##################################text#################

```
SELECT make_array2('a', 'b') AS textarray;
 textarray
-----------
 {a,b}
(1 row)
```

#################################### ##########

```
CREATE FUNCTION is_greater(anyelement, anyelement) RETURNS boolean AS $$
    SELECT $1 > $2;
$$ LANGUAGE SQL;
```

```
SELECT is_greater(1, 2);
 is_greater
------------
 f
(1 row)

CREATE FUNCTION invalid_func() RETURNS anyelement AS $$
    SELECT 1;
$$ LANGUAGE SQL;
ERROR:  cannot determine result data type
DETAIL:  A result of type anyelement requires at least one input of type
 anyelement, anyarray, anynonarray, anyenum, or anyrange.
```

###################### ##########

```
CREATE FUNCTION dup (f1 anyelement, OUT f2 anyelement, OUT f3 anyarray)
AS 'select $1, array[$1,$1]' LANGUAGE SQL;

SELECT * FROM dup(22);
 f2 |   f3
----+---------
 22 | {22,22}
(1 row)
```

####variadic##################

```
CREATE FUNCTION anyleast (VARIADIC anyarray) RETURNS anyelement AS $$
    SELECT min($1[i]) FROM generate_subscripts($1, 1) g(i);
$$ LANGUAGE SQL;

SELECT anyleast(10, -1, 5, 4);
 anyleast
----------
       -1
(1 row)

SELECT anyleast('abc'::text, 'def');
 anyleast
----------
 abc
(1 row)

CREATE FUNCTION concat_values(text, VARIADIC anyarray) RETURNS text AS $$
    SELECT array_to_string($2, $1);
$$ LANGUAGE SQL;

SELECT concat_values('|', 1, 4, 2);
 concat_values
---------------
 1|4|2
(1 row)
```

36.5.12. #######SQL##

SQL#################################23.2#################################### ################### ## ######################### ########################### #######anyleast####### ##

```
SELECT anyleast('abc'::text, 'ABC');
```

########################## C######ABC######################abc###### ########
##COLLATE############### #########

```
SELECT anyleast('abc'::text, 'ABC' COLLATE "C");
```

###COLLATE######### ##
#anyleast##############en_US######

```
CREATE FUNCTION anyleast (VARIADIC anyarray) RETURNS anyelement AS $$
    SELECT min($1[i] COLLATE "en_US") FROM generate_subscripts($1, 1)
 g(i);
$$ LANGUAGE SQL;
```

##

#######################SQL###
################################

##

36.6.

#################SQL#####1######### ################### ###################
###10.3######### #############
#################################### ##
#####

###################

```
CREATE FUNCTION test(int, real) RETURNS ...
CREATE FUNCTION test(smallint, double precision) RETURNS ...
```

test(1, 1.5)#################################### ##############10#########
##################################

#####1##
attribute(*table*)#*table.attribute*################### ###################
######################### ######################*schema.func*(*table*)########
################################

####foo(numeric)#foo(VARIADIC
numeric[])########## ######################foo(10.1)####################
####################2###

C########################## ##################C#############################
#C############## ######################### ################################
################## CREATE FUNCTION SQL########AS###SQL####C##############
#########

```
CREATE FUNCTION test(int) RETURNS int
    AS 'filename', 'test_1arg'
    LANGUAGE C;
CREATE FUNCTION test(int, int) RETURNS int
    AS 'filename', 'test_2arg'
    LANGUAGE C;
```

####C###############1#########

36.7.

################# #######VOLATILE#STABLE#####IMMUTABLE### CREATE FUNCTION###################VOLATILE###### ################################ ###

- VOLATILE############################## ############################### #### ######################### VOLATILE## #####
- STABLE################# ## ## ###################################### ################################### #### ##VOLATILE######### ##########
- IMMUTABLE################# ################################### ########## ###################################### ####SELECT ... WHERE x = 2 + 2### #######SELECT ... WHERE x = 4########### ##################IMMUTABLE# #############

###

###########VOLATILE############ ################## ######################### #########VOLATILE############ ####random()#currval()#timeofday()#####

#########current_timestamp##############################STABLE#######

#######################################STABLE###IMMUTABLE############### ## ## ######################### ####### ############################IMMUTABLE############################## ################ #########################PL/pgSQL###############

SQL## #############SQL######### ################## VOLATILE################STABLE###IMMUTABLE######### ## ###MVCC##13############################## STABLE#IMMUTABLE############## #######################VOLATILE##

> ##
>
> ####C#####################################C######################

###SELECT####### #######STABLE######## PostgreSQL################################STABLE#### ############# ##

IMMUTABLE####SELECT########################## ########IMMUTABLE###### ###########SELECT############# ################################### ## ##PostgreSQL########SELECT###############

#############################IMMUTABLE######## ######################## #TimeZone############### ################STABLE########

> ##
>
> PostgreSQL############STABLE###IMMUTABLE###SELECT###SQL############ ##### ####################VOLATILE################################## ## #######STABLE####IMMUTABLE## ################

36.8.

PostgreSQL##SQL#C####################### #################PL######## ######
PostgreSQL################################## ####40################

36.9.

#######C#######PostgreSQL################## ##############C############## #####
SQL############################## ##########C######SQL#########################
######

#################################18.2#################CREATE FUNCTION#### ################## #####internal#########CREATE FUNCTION########## ## ##sqrt#####################

```
CREATE FUNCTION square_root(double precision) RETURNS double precision
    AS 'dsqrt'
    LANGUAGE internal
    STRICT;
```

###########strict####################

> ##
>
> ############################## SQL#################

36.10. C####

#########C#####C++####C################## ##################################
########################### ##########C##################### ################
########################C#######################

###1##########C##########(#version 1#)# ######################################
##PG_FUNCTION_INFO_V1()###########

36.10.1.

###
########## ##########C####CREATE FUNCTION#################################
#################C##############2################## C##################SQL####
###############

CREATE FUNCTION###

1. ############################
2. ###$libdir##################PostgreSQL########################### ##########
   ##########
3. ########################dynamic_library_path######################
4. ##
   ## ##############################

###.so#######################
######################

#######$libdir################################### ######################
################ $libdir###########pg_config --pkglibdir##############
###

PostgreSQL########ID############################ #########postgres###########
###

#######CREATE FUNCTION################################ ####################
##################

> ##
>
> PostgreSQL#C################ CREATE FUNCTION########################
> ################ ##########36.10.5#########

#####################################PostgreSQL############################
############# ######################PostgreSQL################################
##fmgr.h########include###########
###

```
PG_MODULE_MAGIC;
```

##################################
############################## ##
#################

_PG_init############################
############## ##########void############# ###############################

36.10.2. C###########

C##################PostgreSQL##
####PostgreSQL#################### #######################PostgreSQL######
########### ####PostgreSQL###

#######3###################

- #######
- ########
- ########

#####1#2#4#######################sizeof(Datum)#8####8########### ##########
####long#########4######
###8###############int######Unix#####4###### Unix######int4################
##

```
/* 4 ######### */
typedef int int4;
```

####PostgreSQL#C########int32###### int*XX*#*XX* ########C########### #####int8#
##C######################## int8###SQL###C##int64####### # 36.2##########

################################ #######PostgreSQL##############

```
/* 16 ########### */
typedef struct
{
    double  x, y;
} Point;
```

############PostgreSQL############## ##############palloc()################ ################################ ########1####################palloc######## ####### ########################

############################## ###############4########length################## ## ######SET_VARSIZE############################## #################length####### #################### length##########################length#####################

#########

> ##
>
> ######################## ## ###################### ##############36.12#######

```
####text##################
```

```
typedef struct {
    int32 length;
    char data[FLEXIBLE_ARRAY_MEMBER];
} text;
```

```
[FLEXIBLE_ARRAY_MEMBER]####################################
```

##########################length########################### ####40####text##### #######################

```
#include "postgres.h"
...

char buffer[40]; /* ######### */
...
text *destination = (text *) palloc(VARHDRSZ + 40);
SET_VARSIZE(destination, VARHDRSZ + 40);
memcpy(destination->data, buffer, 40);
...
```

```
VARHDRSZ#sizeof(int32)##############################VARHDRSZ###########
############## ##################SET_VARSIZE####################
```

36.2##PostgreSQL#####SQL###########C######## ##################include######### #########################include## ######## ###################postgres.h####include################## ############ #########################include######################

#36.2 ####SQL######C##

SQL#	C ###	####
boolean	bool	postgres.h#####################
box	BOX*	utils/geo_decls.h

SQL#	C ###	####
bytea	bytea*	postgres.h
"char"	char	#############
character	BpChar*	postgres.h
cid	CommandId	postgres.h
date	DateADT	utils/date.h
float4 (real)	float4	postgres.h
float8 (double precision)	float8	postgres.h
int2 (smallint)	int16	postgres.h
int4 (integer)	int32	postgres.h
int8 (bigint)	int64	postgres.h
interval	Interval*	datatype/timestamp.h
lseg	LSEG*	utils/geo_decls.h
name	Name	postgres.h
numeric	Numeric	utils/numeric.h
oid	Oid	postgres.h
oidvector	oidvector*	postgres.h
path	PATH*	utils/geo_decls.h
point	POINT*	utils/geo_decls.h
regproc	RegProcedure	postgres.h
text	text*	postgres.h
tid	ItemPointer	storage/itemptr.h
time	TimeADT	utils/date.h
time with time zone	TimeTzADT	utils/date.h
timestamp	Timestamp	datatype/timestamp.h
timestamp with time zone	TimestampTz	datatype/timestamp.h
varchar	VarChar*	postgres.h
xid	TransactionId	postgres.h

###

36.10.3. Version 1

Version-1## Version-1###C#################
##

```
Datum funcname(PG_FUNCTION_ARGS)
```

###########

```
PG_FUNCTION_INFO_V1(funcname);
```

######################## ################### PostgreSQL###########Version-1#######
##############internal############# #####################

Version-1#######################PG_GETARG_*xxx*()############### ########
##PG_ARGISNULL()######NULL################# ##########
#PG_RETURN_*xxx*()############ PG_GETARG_*xxx*()###########################
######## PG_RETURN_*xxx*()####################

Version-1######################

```
#include "postgres.h"
#include <string.h>
#include "fmgr.h"
#include "utils/geo_decls.h"
#include "varatt.h"

PG_MODULE_MAGIC;


/* ### */

PG_FUNCTION_INFO_V1(add_one);

Datum
add_one(PG_FUNCTION_ARGS)
{
    int32   arg = PG_GETARG_INT32(0);

    PG_RETURN_INT32(arg + 1);
}


/* ######## */

PG_FUNCTION_INFO_V1(add_one_float8);

Datum
add_one_float8(PG_FUNCTION_ARGS)
{

    /* FLOAT8################### */
    float8   arg = PG_GETARG_FLOAT8(0);

    PG_RETURN_FLOAT8(arg + 1.0);
}

PG_FUNCTION_INFO_V1(makepoint);

Datum
makepoint(PG_FUNCTION_ARGS)
{

    /* ###Point################### */
    Point     *pointx = PG_GETARG_POINT_P(0);
    Point     *pointy = PG_GETARG_POINT_P(1);
    Point     *new_point = (Point *) palloc(sizeof(Point));

    new_point->x = pointx->x;
    new_point->y = pointy->y;

    PG_RETURN_POINT_P(new_point);
}


/* ######## */
```

```
PG_FUNCTION_INFO_V1(copytext);

Datum
copytext(PG_FUNCTION_ARGS)
{
    text     *t = PG_GETARG_TEXT_PP(0);

    /*

     * VARSIZE########VARHDRSZ###VARHDRSZ_SHORT####
     * ####################
     * ########################
     */
    text     *new_t = (text *) palloc(VARSIZE_ANY_EXHDR(t) + VARHDRSZ);
    SET_VARSIZE(new_t, VARSIZE_ANY_EXHDR(t) + VARHDRSZ);

    /*

     * VARDATA######################
     * #####short#############VARDATA_ANY############
     */
    memcpy(VARDATA(new_t),          /* destination */
           VARDATA_ANY(t),          /* source */
           VARSIZE_ANY_EXHDR(t));   /* how many bytes */
    PG_RETURN_TEXT_P(new_t);
}

PG_FUNCTION_INFO_V1(concat_text);

Datum
concat_text(PG_FUNCTION_ARGS)
{
    text  *arg1 = PG_GETARG_TEXT_PP(0);
    text  *arg2 = PG_GETARG_TEXT_PP(1);
    int32 arg1_size = VARSIZE_ANY_EXHDR(arg1);
    int32 arg2_size = VARSIZE_ANY_EXHDR(arg2);
    int32 new_text_size = arg1_size + arg2_size + VARHDRSZ;
    text *new_text = (text *) palloc(new_text_size);

    SET_VARSIZE(new_text, new_text_size);
    memcpy(VARDATA(new_text), VARDATA_ANY(arg1), arg1_size);
    memcpy(VARDATA(new_text) + arg1_size, VARDATA_ANY(arg2), arg2_size);
    PG_RETURN_TEXT_P(new_text);
}
```

##########funcs.c#####################################PostgreSQL#############
###

```
CREATE FUNCTION add_one(integer) RETURNS integer
     AS 'DIRECTORY/funcs', 'add_one'
     LANGUAGE C STRICT;


-- SQL###"add_one"###########
CREATE FUNCTION add_one(double precision) RETURNS double precision
     AS 'DIRECTORY/funcs', 'add_one_float8'
     LANGUAGE C STRICT;

CREATE FUNCTION makepoint(point, point) RETURNS point
     AS 'DIRECTORY/funcs', 'makepoint'
     LANGUAGE C STRICT;
```

```
CREATE FUNCTION copytext(text) RETURNS text
     AS 'DIRECTORY/funcs', 'copytext'
     LANGUAGE C STRICT;

CREATE FUNCTION concat_text(text, text) RETURNS text
     AS 'DIRECTORY/funcs', 'concat_text'
     LANGUAGE C STRICT;
```

#####*DIRECTORY*#####################PostgreSQL################################ ################ (*DIRECTORY*###########AS##'funcs'################## ####### ########################.so########)

####strict############################NULL#################NULL############ ### ###############NULL######################## ########PG_ARGISNULL()#### NULL##################

PG_ARGISNULL(*n*)############NULL#################### ################### ########### PG_GETARG_*xxx*()######################## ###NULL########### ##PG_GETARG_*xxx*()############### #####NULL#######PG_RETURN_NULL()####### ###########################

###Version-1#############C#################################### ####NULL####### ######TOAST#######################

Version 1#####################PG_GETARG_*xxx*()#######2######## 1## #PG_GETARG_*xxx*_COPY()################################## ################## ################################## PG_GETARG_*xxx*_COPY()################## ######## 2#########3###PG_GETARG_*xxx*_SLICE()########## 1################## ### 2###3######################## ################################## ######## #############external################################## ###########ALTER TABLE *tablename* ALTER COLUMN *colname* SET STORAGE *storagetype*########### *storagetype*##plain#external#extended####main########

####Version-1################36.10.8#################37##################56###### #### ##############src/backend/utils/fmgr/README#########

36.10.4.

#############PostgreSQL C########################### C############### PostgreSQL#########################C++#FORTRAN#Pascal#######C############### ########################## ## #####C#######C#################

C####################

- pg_config --includedir-server################################PostgreSQL ##############################
- PostgreSQL#######################/######################## ################ #######/##########36.10.5#########
- ####36.10.1################################
- ##########C######malloc#free#####PostgreSQL#palloc#pfree######### palloc##
- memset################################palloc0############## ############## ## ################################## ### ## ############################## ## ##

- #####PostgreSQL#####postgres.h######### ##############PG_FUNCTION_ARGS#
  ###fmgr.h######### ############2#######include######### ##########
  ##postgres.h##############################include########### postgres.h#
  include#####elog.h#palloc.h#include#########

- ##############################PostgreSQL################################## #
  ####################################

36.10.5.

C#####PostgreSQL### ########
############

##################################C#####cc#######ld################## ##
##PostgreSQL#######contrib################## ##############PostgreSQL#######
##########################

################################## ##
######### ######################PIC############### ###############
################################## ##
##
####################

###########foo.c###########foo.so################## ######################
#foo.o####### #######1############################1#########

FreeBSD

PIC##############-fPIC### ##################-shared###

```
cc -fPIC -c foo.c
cc -shared -o foo.so foo.o
```

#####[51.351351]####FreeBSD######3.0######## #########FreeBSD######13.0########
#########gcc##############

Linux

PIC##############-fPIC### ##################-shared### ###############

```
cc -fPIC -c foo.c
cc -shared -o foo.so foo.o
```

macOS

######### ##########################

```
cc -c foo.c
cc -bundle -flat_namespace -undefined suppress -o foo.so foo.o
```

NetBSD

PIC##############-fPIC### ELF######-shared########################### ###
##ELF######ld -Bshareable#######

```
gcc -fPIC -c foo.c
```

```
gcc -shared -o foo.so foo.o
```

OpenBSD

PIC#################-fPIC### ##############ld -Bshareable#######

```
gcc -fPIC -c foo.c
ld -Bshareable -o foo.so foo.o
```

Solaris

PIC##############Sun#######-KPIC##GCC##-fPIC### ######################
##############-G##GCC#######-shared###############

```
cc -KPIC -c foo.c
cc -G -o foo.so foo.o
```

```
####
```

```
gcc -fPIC -c foo.c
gcc -G -o foo.so foo.o
```

> **###**
>
> #################GNU Libtool[1]############ ################################
> #######

##################PostgreSQL############## CREATE FUNCTION############
######################.so####.sl#
#CREATE FUNCTION##################################

##########################36.10.1#########

36.10.6.

#####C###################### ###########NULL############### ##################
#################################### #####PostgreSQL#C#########################
############

#################################

```
SELECT name, c_overpaid(emp, 1500) AS overpaid
    FROM emp
    WHERE name = 'Bill' OR name = 'Sam';
```

Version 1#############c_overpaid##############

```
#include "postgres.h"

#include "executor/executor.h"  /* GetAttributeByName()# */

PG_MODULE_MAGIC;
```

[1] https://www.gnu.org/software/libtool/

```
PG_FUNCTION_INFO_V1(c_overpaid);

Datum
c_overpaid(PG_FUNCTION_ARGS)
{
    HeapTupleHeader  t = PG_GETARG_HEAPTUPLEHEADER(0);
    int32            limit = PG_GETARG_INT32(1);
    bool isnull;
    Datum salary;

    salary = GetAttributeByName(t, "salary", &isnull);
    if (isnull)
        PG_RETURN_BOOL(false);

    /* ####salary#NULL#####PG_RETURN_NULL()############ */

    PG_RETURN_BOOL(DatumGetInt32(salary) > limit);
}
```

GetAttributeByName###############PostgreSQL######## ####3######### ######## ####HeapTupleHeader#################NULL#################### GetAttributeByName####DatumGet*XXX*()##################Datum######## ## NULL##NULL############

#################GetAttributeByNum######

########c_overpaid###SQL#######

```
CREATE FUNCTION c_overpaid(emp, integer) RETURNS boolean
    AS 'DIRECTORY/funcs', 'c_overpaid'
    LANGUAGE C STRICT;
```

#####NULL################STRICT##################

36.10.7.

C###API######## ##API##### ################include#########

```
#include "funcapi.h"
```

#########################2######### Datum################################ #######C################ ##################TupleDesc################### ### Datum########TupleDesc#BlessTupleDesc#########heap_form_tuple## ###### C###########TupleDesc #TupleDescGetAttInMetadata######### BuildTupleFromCStrings######## ## ####

###TupleDesc################### ##############################

```
TypeFuncClass get_call_result_type(FunctionCallInfo fcinfo,
                                   Oid *resultTypeId,
                                   TupleDesc *resultTupleDesc)
```

###############fcinfo############### #########version 1#################### resultTypeId#NULL##############################OID############ resultTupleDesc#####TupleDesc################# ###TYPEFUNC_COMPOSITE## ########### TYPEFUNC_COMPOSITE######resultTupleDesc#####TupleDesc#####

#TYPEFUNC_COMPOSITE#########function returning record called in context that cannot accept type record##

###

get_call_result_type##################### ########################### ############ resultTypeId######################

##

get_call_result_type##get_expr_result_type#######################
################# ############################## #
##get_func_result_type########## ######OID################# ########
####record###################### ###get_func_result_type###########
#########get_call_result_type########

########TupleDesc##################

```
TupleDesc RelationNameGetTupleDesc(const char *relname)
```

#################TupleDesc################ ###

```
TupleDesc TypeGetTupleDesc(Oid typeoid, List *colaliases)
```

#####OID#####TupleDesc################ ##############TupleDesc############
###record################

TupleDesc########Datum####################

```
TupleDesc BlessTupleDesc(TupleDesc tupdesc)
```

C#######################

```
AttInMetadata *TupleDescGetAttInMetadata(TupleDesc tupdesc)
```

##########################FuncCallContext############ ###
#tuple_desc#attinmeta######

Datum###############

```
HeapTuple heap_form_tuple(TupleDesc tupdesc, Datum *values, bool *isnull)
```

#######Datum#######HeapTuple#####

C##################

```
HeapTuple BuildTupleFromCStrings(AttInMetadata *attinmeta, char **values)
```

#######C#########HeapTuple###### *values*######1#####C######## #C#########
########################### #####NULL##########*values*###########NULL#####
##############################

#####################Datum############# #######

```
HeapTupleGetDatum(HeapTuple tuple)
```

HeapTuple####Datum######### ###############Datum############ ############## ##################

#########

36.10.8.

C################2######### ####*ValuePerCall*################################ ########################1##################NULL##### #############SRF#### ## ######*Materialize*################ ################################ ######################################

ValuePerCall### ####LIMIT################ ### ################################## ##################### ################################Materialize################ ##

##########ValuePerCall## #### Materialize###########src/backend/utils/fmgr/README######### ### PostgreSQL########contrib########ValuePerCall#Materialize###################### #########

#######ValuePerCall##############funcapi.h#include#### ######################## ##########FuncCallContext########### ###########fcinfo->flinfo->fn_extra########FuncCallContext################### ##################### ################################

```
typedef struct FuncCallContext
{
    /*

     * ##############
     *

     * SRF_FIRSTCALL_INIT()####call_cntr#0#######
     * SRF_RETURN_NEXT()################
     */
    uint64 call_cntr;

    /*

     * #### : ########
     *
     * max_calls##########################
     * ##############################
     * ############
     */
    uint64 max_calls;

    /*

     * #### : #######################
     *
     * user_fctx#######################
     * ##############################
     */
```

```
    void *user_fctx;

    /*

     * #### : #######################
     *
     * attinmeta #########################
     * #####################
     * BuildTupleFromCStrings()##########################
     */
    AttInMetadata *attinmeta;

    /*

     *  ###############################
     *
     * multi_call_memory_ctx##SRF_FIRSTCALL_INIT()##########
     * SRF_RETURN_DONE()#################
     * ###SRF########################################
     */
    MemoryContext multi_call_memory_ctx;

    /*

     * ####: ##################
     * tuple_desc##########################BuildTupleFromCStrings()
     * ####heap_form_tuple()######################
     * ##########TupleDesc####BlessTupleDesc()#################
     * ####
     */
    TupleDesc tuple_desc;

} FuncCallContext;

##########SRF################


SRF_IS_FIRSTCALL()

######################2################### #############


SRF_FIRSTCALL_INIT()

######FuncCallContext####### ###################


SRF_PERCALL_SETUP()

######FuncCallContext#############

##########################


SRF_RETURN_NEXT(funcctx, result)

###################### ######### result#Datum#####1#################### ######
#############


SRF_RETURN_DONE(funcctx)

#####SRF###########

SRF################################################## ###palloc##############
###pfree########### ################### ######################################
```

################### multi_call_memory_ctx####################SRF###########
############################# ##########################
#multi_call_memory_ctx################## funcctx->user_fctx############
########################## #multi_call_memory_ctx##########################
############################

##
####################################TOAST############   ##PG_GETARG_*xxx*#################TOAST#################### ##   ##user_fctx##################TOAST######multi_call_memory_ctx#####   ##########TOAST##################################

################

```
Datum
my_set_returning_function(PG_FUNCTION_ARGS)
{
    FuncCallContext  *funcctx;
    Datum             result;
    further declarations as needed

    if (SRF_IS_FIRSTCALL())
    {
        MemoryContext oldcontext;

        funcctx = SRF_FIRSTCALL_INIT();
        oldcontext = MemoryContextSwitchTo(funcctx-
>multi_call_memory_ctx);

        /* #######################: */
        user code
        if returning composite
            build TupleDesc, and perhaps AttInMetadata
        endif returning composite
        user code
        MemoryContextSwitchTo(oldcontext);
    }


    /* ########################: */
    user code
    funcctx = SRF_PERCALL_SETUP();
    user code


    /* #####################1###: */
    if (funcctx->call_cntr < funcctx->max_calls)
    {

        /* #############: */
        user code
        obtain result Datum
        SRF_RETURN_NEXT(funcctx, result);
    }
    else
    {

        /* ############### ########### */
        /* ############################## */
```

```
        SRF_RETURN_DONE(funcctx);
    }
}
```

#########SRF##############

```
PG_FUNCTION_INFO_V1(retcomposite);

Datum
retcomposite(PG_FUNCTION_ARGS)
{
    FuncCallContext     *funcctx;
    int                  call_cntr;
    int                  max_calls;
    TupleDesc            tupdesc;
    AttInMetadata       *attinmeta;


     /* ################ */
    if (SRF_IS_FIRSTCALL())
    {
        MemoryContext   oldcontext;


        /* ####################### */
        funcctx = SRF_FIRSTCALL_INIT();


        /* ########################### */
        oldcontext = MemoryContextSwitchTo(funcctx-
>multi_call_memory_ctx);


        /* ########### */
        funcctx->max_calls = PG_GETARG_INT32(0);


        /*  ############## */
        if (get_call_result_type(fcinfo, NULL, &tupdesc) !=
 TYPEFUNC_COMPOSITE)
            ereport(ERROR,
                    (errcode(ERRCODE_FEATURE_NOT_SUPPORTED),
                     errmsg("function returning record called in context "
                            "that cannot accept type record")));

        /*

         * ######C######################
         * ##########
         */
        attinmeta = TupleDescGetAttInMetadata(tupdesc);
        funcctx->attinmeta = attinmeta;

        MemoryContextSwitchTo(oldcontext);
    }


    /* ############ */
    funcctx = SRF_PERCALL_SETUP();

    call_cntr = funcctx->call_cntr;
    max_calls = funcctx->max_calls;
    attinmeta = funcctx->attinmeta;
```

```
    if (call_cntr < max_calls)    /* ############  */
    {
        char       **values;
        HeapTuple    tuple;
        Datum        result;

        /*

         * #############values#########
         * ###################
         * C#################
         */
        values = (char **) palloc(3 * sizeof(char *));
        values[0] = (char *) palloc(16 * sizeof(char));
        values[1] = (char *) palloc(16 * sizeof(char));
        values[2] = (char *) palloc(16 * sizeof(char));

        snprintf(values[0], 16, "%d", 1 * PG_GETARG_INT32(1));
        snprintf(values[1], 16, "%d", 2 * PG_GETARG_INT32(1));
        snprintf(values[2], 16, "%d", 3 * PG_GETARG_INT32(1));


        /* ###### */
        tuple = BuildTupleFromCStrings(attinmeta, values);


        /* ####datum### */
        result = HeapTupleGetDatum(tuple);


        /* ###################### */
        pfree(values[0]);
        pfree(values[1]);
        pfree(values[2]);
        pfree(values);

        SRF_RETURN_NEXT(funcctx, result);
    }

    else    /* ########## */
    {
        SRF_RETURN_DONE(funcctx);
    }
}
```

########SQL#############

```
CREATE TYPE __retcomposite AS (f1 integer, f2 integer, f3 integer);

CREATE OR REPLACE FUNCTION retcomposite(integer, integer)
    RETURNS SETOF __retcomposite
    AS 'filename', 'retcomposite'
    LANGUAGE C IMMUTABLE STRICT;
```

#########OUT##################

```
CREATE OR REPLACE FUNCTION retcomposite(IN integer, IN integer,
    OUT f1 integer, OUT f2 integer, OUT f3 integer)
    RETURNS SETOF record
    AS 'filename', 'retcomposite'
    LANGUAGE C IMMUTABLE STRICT;
```

###################record###############

36.10.9.

C######36.2.5################################ ################################ ## Version-1 C################################2#######fmgr.h######### #######get_fn_expr_rettype(FmgrInfo *flinfo)#get_fn_expr_argtype(FmgrInfo *flinfo, int argnum)######## #############OID###### ################InvalidOid#### flinfo######fcinfo->flinfo########### argnum######0######## ##get_fn_expr_rettype#####get_call_result_type############ ###variadic#### #######################get_fn_expr_variadic###### ########variadic########### ##################VARIADIC "any"########

#######################1################

```
PG_FUNCTION_INFO_V1(make_array);
Datum
make_array(PG_FUNCTION_ARGS)
{
    ArrayType  *result;
    Oid         element_type = get_fn_expr_argtype(fcinfo->flinfo, 0);
    Datum       element;
    bool        isnull;
    int16       typlen;
    bool        typbyval;
    char        typalign;
    int         ndims;
    int         dims[MAXDIM];
    int         lbs[MAXDIM];

    if (!OidIsValid(element_type))
        elog(ERROR, "could not determine data type of input");


    /* ########NULL####################*/
    isnull = PG_ARGISNULL(0);
    if (isnull)
        element = (Datum) 0;
    else
        element = PG_GETARG_DATUM(0);


    /* ####1 */
    ndims = 1;

    /* ###1# */
    dims[0] = 1;

    /* ###1 */
    lbs[0] = 1;


    /* ################### */
    get_typlenbyvalalign(element_type, &typlen, &typbyval, &typalign);


    /* ######## */
    result = construct_md_array(&element, &isnull, ndims, dims, lbs,
                                element_type, typlen, typbyval, typalign);

    PG_RETURN_ARRAYTYPE_P(result);
```

```
}

########SQL#make_array#########


CREATE FUNCTION make_array(anyelement) RETURNS anyarray
    AS 'DIRECTORY/funcs', 'make_array'
    LANGUAGE C IMMUTABLE;

C########################  "any"####################  #######SQL##############
#######################  ######"any"#########################################
##anyelement#########  C############VARIADIC  "any"############  ###########
##########################  ###########variadic#######################  #########
###########  PG_NARGS()#######################################################
##  #######################################################VARIADIC#########
###########  ###########get_fn_expr_variadic############VARIADIC############
#####################
```

36.10.10. Shared Memory

36.10.10.1. Requesting Shared Memory at Startup

```
#####[72.666667]######LWLocks################################
shared_preload_libraries#####################################  #########_PG_init#
##shmem_request_hook############   ##shmem_request_hook##LWLocks#######
#########   ##########   ##############################   ##################
#shared_preload_libraries#####################  ###shmem_request_hook#_PG_init#
#############  ##shmem_request_hook#######################


void RequestAddinShmemSpace(Size size)

#######################################################


void *ShmemInitStruct(const char *name, Size size, bool *foundPtr)

###########foundPtr#false########################################
foundPtr#true#################################################################

#####[74.096386]#######################################################
#AddinShmemInitLock##################  ####################################
####LWLockAddinShmemInitLock############


static mystruct *ptr = NULL;
bool        found;

LWLockAcquire(AddinShmemInitLock, LW_EXCLUSIVE);
ptr = ShmemInitStruct("my struct name", size, &found);
if (!found)
{
    ... initialize contents of shared memory ...
    ptr->locks = GetNamedLWLockTranche("my tranche name");
}
LWLockRelease(AddinShmemInitLock);

######shmem_startup_hook########################################  #########
#################shmem_startup_hook######            ####################
#AddinShmemInitLock############
```

#####[76.050420]#shmem_request_hook####PostgreSQL######contrib/
pg_stat_statements/pg_stat_statements.c##### ####
##shmem_request_hook#shmem_startup_hook####PostgreSQL######contrib/
pg_stat_statements/pg_stat_statements.c#####

36.10.10.2. Requesting Shared Memory After Startup

############shmem_request_hook##############################1###### #######
###

```
void *GetNamedDSMSegment(const char *name, size_t size,
                         void (*init_callback) (void *ptr),
                         bool *found)
```

##init_callback#############
##
##

###########################GetNamedDSMSegment###########
#AddinShmemInitLock################################ ####################
#######1###

#####[54.838710]#shmem_request_hook####PostgreSQL######contrib/
pg_stat_statements/pg_stat_statements.c##### ####
##GetNamedDSMSegment########PostgreSQL######src/test/modules/
test_dsm_registry/test_dsm_registry.c#####

36.10.11. LWLocks

36.10.11.1. Requesting LWLocks at Startup

#####[75.052411]######LWLocks###############################
shared_preload_libraries#################################### ########_PG_init#
##shmem_request_hook############ ##shmem_request_hook##LWLocks#########
##################LWLock#######
######shared_preload_libraries##################_PG_init###shmem_request_hook#
############ ##shmem_request_hook##########LWLock#######

```
void RequestNamedLWLockTranche(const char *tranche_name, int num_lwlocks)
```

############num_lwlocks LWLocks####tranche_name############### ############
################

```
LWLockPadded *GetNamedLWLockTranche(const char *tranche_name)
```

36.10.11.2. Requesting LWLocks After Startup

############shmem_request_hook#####LWLock#################### ###########
######tranche_id#######

```
int LWLockNewTrancheId(void)
```

############tranche_id##########LWLock#######

```
void LWLockInitialize(LWLock *lock, int tranche_id)
```

##########################tranche_id#1################LWLock############# ######## #####1######AddinShmemInitLock################################# ######### GetNamedDSMSegment#######init_callback######################## ##########

##########tranche_id######################tranche_name#############

```
void LWLockRegisterTranche(int tranche_id, const char *tranche_name)
```

######LWLockNewTrancheId#LWLockInitialize#LWLockRegisterTranche###### ###PostgreSQL#######contrib/pg_prewarm/autoprewarm.c######

36.10.12. Custom Wait Events

##################################Extension####################

```
uint32 WaitEventExtensionNew(const char *wait_event_name)
```

##################################### ##PostgreSQL######src/test/modules/worker_spi######

################pg_stat_activity##########

```
=# SELECT wait_event_type, wait_event FROM pg_stat_activity
     WHERE backend_type ~ 'worker_spi';
 wait_event_type |  wait_event
-----------------+---------------
 Extension       | WorkerSpiMain
(1 row)
```

36.10.13. Injection Points

###########name####################

```
INJECTION_POINT(name);
```

C############################

###

```
extern void InjectionPointAttach(const char *name,
                                 const char *library,
                                 const char *function,
                                 const void *private_data,
                                 int private_data_size);
```

######name######library#######function################## private_data#### ####################private_data_size###############

######InjectionPointCallback################

```
static void
custom_injection_callback(const char *name, const void *private_data)
{
    uint32 wait_event_info = WaitEventInjectionPointNew(name);

    pgstat_report_wait_start(wait_event_info);
    elog(NOTICE, "%s: executed custom callback", name);
    pgstat_report_wait_end();
}
```

###################NOTICE###

#####################################

```
extern bool InjectionPointDetach(const char *name);
```

############true########false######

##########################InjectionPointAttach#########################
########### ####################InjectionPointDetach#####################
########

########PostgreSQL######src/###/#####/########_points######

##########################configure#--enable-injection-points#####
##Meson#-Injection_points=true######

36.10.14. ####C++###

###############PostgreSQL#########C++################

- ##############################C#################### ##C###C++###############
  ######################extern C######## #########C++######################
  ###

- ######################### ##################palloc()#########pfree()#####
  ####### #####C++#delete()############

- ###C##################extern C################################ #########
  #######################C++############################## ##################
  #########C############### ####################-fno-exceptions####C++####
  ######## #######new()#####NULL######C++######################

- C++######################C++#########C#######POD#################### ##
  ##########POD########C++#########################longjmp()##############
  ##

#############################extern C######C++##############################
#########

36.11.

####################
############################## ##################################

```
CREATE     FUNCTION################################  ####################
#(IMMUTABLE#STABLE###VOLATILE)################################ ###########
#####################(PARALLEL   UNSAFE#PARALLEL   RESTRICTED###PARALLEL
```

SAFE)############# ## ######2#####
##

SQL###
######### ##########################C##################################

###############SQL###############

```
supportfn(internal) returns internal
```

###########SUPPORT#####################

###########API#####PostgreSQL##########src/include/nodes/supportnodes.h##
####### ################################## ##################################
#####################

#################################### ####int4mul(n, 1)#n########## ######
##SupportRequestSimplify################################ #################
####################### ## ########
####################—#####n * 1#n######### ##############################
######PostgreSQL############## ################PostgreSQL####################
####################################

boolean####################WHERE################################ ##
#SupportRequestSelectivity######################

##
#SupportRequestCost######################

###SupportRequestRows##
#####################

boolean############WHERE#################################### ############
###########################
##########################WHERE##
###################SupportRequestIndexCondition###################

36.12.

PostgreSQL############################## ##################################### #
##########################
(###)############### ## #######
################################## ##
###

###

sum############## sum#0######
######################### ######sum######(complex)##############################
########## ################

```
CREATE AGGREGATE sum (complex)
(
    sfunc = complex_add,
    stype = complex,
    initcond = '(0,0)'
);
```

##############

```
SELECT sum(a) FROM test_complex;

   sum
-----------
 (34,53.9)
```

############################## sum##################PostgreSQL###complex### ####sum################

###sum########NULL########0########### #######NULL##################SQL# #sum#################### ##########initcond############NULL###### ###### ##sfunc#NULL######################### ###sum#####max#min################## ####NULL######## #####NULL######################## PostgreSQL########NULL#### #####strict################################NULL##################

##1###strict####################NULL################################### #### ##NULL######### ##NULL#######################strict#################NULL##### ##############################

avg#################### ####2################################ ############## ############# ########################### ####avg(float8)################### ##

```
CREATE AGGREGATE avg (float8)
(
    sfunc = float8_accum,
    stype = float8[],
    finalfunc = float8_avg,
    initcond = '{0,0,0}'
);
```

> ##
>
> float8_accum## ####avg## ####################

SQL#############DISTINCT#ORDER BY## ################################

###########CREATE AGGREGATE#############

36.12.1.

(#######################3.5# 4.2.8 #########) ################################## ## ####sum########### ################################ ## ########### ##############################

#######################

##########################sum#########

```
CREATE AGGREGATE sum (complex)
(
    sfunc = complex_add,
    stype = complex,
    initcond = '(0,0)',
    msfunc = complex_add,
    minvfunc = complex_sub,
    mstype = complex,
    minitcond = '(0,0)'
);
```

m########################## #####minvfunc###m#################### ############################null############## ######null######################## ## ###################### ########################## ## ######## ########NaN(##)## ## ###################### ################################ ##########################float4#float8## #sum### ###sum(float8)######

```
CREATE AGGREGATE unsafe_sum (float8)
(
    stype = float8,
    sfunc = float8pl,
    mstype = float8,
    msfunc = float8pl,
    minvfunc = float8mi
);
```

#######

```
SELECT
  unsafe_sum(x) OVER (ORDER BY n ROWS BETWEEN CURRENT ROW AND 1 FOLLOWING)
FROM (VALUES (1, 1.0e20::float8),
             (2, 1.0::float8)) AS v (n,x);
```

######## ##########1####0###### ####################1e20#1#########1e20#### ########1e20####1####0###### ####PostgreSQL##################################

36.12.2.

36.2.5######### ##### ##(###)################ #########

```
CREATE AGGREGATE array_accum (anycompatible)
(
    sfunc = array_append,
    stype = anycompatiblearray,
    initcond = '{}'
);
```

##########array_agg##################################

###2###########################

```
SELECT attrelid::regclass, array_accum(attname)
    FROM pg_attribute
    WHERE attnum > 0 AND attrelid = 'pg_tablespace'::regclass
    GROUP BY attrelid;

   attrelid    |              array_accum
---------------+---------------------------------------
 pg_tablespace | {spcname,spcowner,spcacl,spcoptions}
(1 row)

SELECT attrelid::regclass, array_accum(atttypid::regtype)
    FROM pg_attribute
    WHERE attnum > 0 AND attrelid = 'pg_tablespace'::regclass
    GROUP BY attrelid;

   attrelid    |        array_accum
---------------+---------------------------
 pg_tablespace | {name,oid,aclitem[],text[]}
(1 row)
```

################################## ##
######################CREATE FUNCTION################################ ######
#################### ####################C######internal#################
###########SQL############## ##
######## ##null######### ###############
######################### ################array_agg######

```
CREATE FUNCTION array_agg_transfn(internal, anynonarray)
  RETURNS internal ...;
CREATE FUNCTION array_agg_finalfn(internal, anynonarray)
  RETURNS anyarray ...;

CREATE AGGREGATE array_agg (anynonarray)
(
    sfunc = array_agg_transfn,
    stype = internal,
    finalfunc = array_agg_finalfn,
    finalfunc_extra
);
```

####finalfunc_extra###
array_agg_finalfn#####anynonarray####################

##############################VARIADIC#####################################
36.5.6######### #######(###)############################ ####(###)######
##VARIADIC####################

> ##
>
> #######ORDER BY######4.2.7###
> ############## ORDER BY### ####
>
> ```
> SELECT myaggregate(a ORDER BY a, b, c) FROM ...
> ```
>
> ############################## ################################

```
SELECT myaggregate(a, b, c ORDER BY a) FROM ...

##myaggregate##############################

##########################################
```

36.12.3.

```
###################### PostgreSQL######################################### ####
############################################################## #############
################ ##################################(#######)##################
############# #############percentile_disc########


CREATE FUNCTION ordered_set_transition(internal, anyelement)
  RETURNS internal ...;
CREATE FUNCTION percentile_disc_final(internal, float8, anyelement)
  RETURNS anyelement ...;

CREATE AGGREGATE percentile_disc (float8 ORDER BY anyelement)
(
    sfunc = ordered_set_transition,
    stype = internal,
    finalfunc = percentile_disc_final,
    finalfunc_extra
);

######float8######################################## #########################
###


SELECT percentile_disc(0.5) WITHIN GROUP (ORDER BY income) FROM
 households;
 percentile_disc
-----------------
           50489

###0.5##################################

############################################################## ###############
###tuplesort#################################################################  ####
################################################## ############PL/pgSQL#####PL
########################################SQL#####################C######### (##
########### — ################################) ################################
###################### ########READ_ONLY#####################################
#####CREATE AGGREGATE#READ_WRITE###SHAREABLE#############

#######################################################  ######################
########################### ##############################finalfunc_extra####
#################null######## #########finalfunc_extra######(#)##############
#########(###)##################################

#########################################################
```

36.12.4.

```
########################## ##################################################
############################################################## ################
############################################## ##################################
```

############# ##
####

##################################### ##
#######################
############## ##

##########MAX#MIN###
############## SUM###################### ##################################

######null##strict########
########## ###initcond##null###
##

#######internal## ######
#################NULL###
##########

#######internal###
#################### ############################/##########################
#####

##########internal##############blob###############bytea####### #############
######### bytea##internal##########internal###### #######################
############## ##

##################PARALLEL SAFE############################### #############
###############

36.12.5.

C############AggCheckCallContext###################################### ####

```
if (AggCheckCallContext(fcinfo, NULL))
```

#
###int8inc()######### ##
#################################### ######CREATE AGGREGATE########

AggCheckCallContext## ###########
############36.13.1################## ##
array_append()############
#array_append()###

#############C#######AggGetAggref###########################Aggref########
################################Aggref#####################
##PostgreSQL#######orderedsetaggs.c#############

36.13.

36.2############PostgreSQL########################### #####SQL#################
######################### ####################C######################

#################src/tutorial######complex.sql#complex.c############# #
###################README#########

  ########################### ##
########################### ### ##########

```
######################## ##################################################
#############

###########complex############# #########C############################


typedef struct Complex {
    double      x;
    double      y;
} Complex;

###Datum################################

#############(x,y)##################

############################## ##################################################
########## ##########


PG_FUNCTION_INFO_V1(complex_in);

Datum
complex_in(PG_FUNCTION_ARGS)
{
    char       *str = PG_GETARG_CSTRING(0);
    double      x,
                y;
    Complex    *result;

    if (sscanf(str, " ( %lf , %lf )", &x, &y) != 2)
        ereport(ERROR,
                (errcode(ERRCODE_INVALID_TEXT_REPRESENTATION),
                 errmsg("invalid input syntax for type %s: \"%s\"",
                        "complex", str)));

    result = (Complex *) palloc(sizeof(Complex));
    result->x = x;
    result->y = y;
    PG_RETURN_POINTER(result);
}


##################


PG_FUNCTION_INFO_V1(complex_out);

Datum
complex_out(PG_FUNCTION_ARGS)
{
    Complex    *complex = (Complex *) PG_GETARG_POINTER(0);
    char       *result;

    result = psprintf("(%g,%g)", complex->x, complex->y);
    PG_RETURN_CSTRING(result);
}


##############################  ##############################################
## ############################

######################################### ####################################
###### ##################################### ################################
########## complex###float8####################
```

```
PG_FUNCTION_INFO_V1(complex_recv);

Datum
complex_recv(PG_FUNCTION_ARGS)
{
    StringInfo  buf = (StringInfo) PG_GETARG_POINTER(0);
    Complex    *result;

    result = (Complex *) palloc(sizeof(Complex));
    result->x = pq_getmsgfloat8(buf);
    result->y = pq_getmsgfloat8(buf);
    PG_RETURN_POINTER(result);
}

PG_FUNCTION_INFO_V1(complex_send);

Datum
complex_send(PG_FUNCTION_ARGS)
{
    Complex    *complex = (Complex *) PG_GETARG_POINTER(0);
    StringInfoData buf;

    pq_begintypsend(&buf);
    pq_sendfloat8(&buf, complex->x);
    pq_sendfloat8(&buf, complex->y);
    PG_RETURN_BYTEA_P(pq_endtypsend(&buf));
}
```

##########################SQL#complex############## ###############

```
CREATE TYPE complex;
```

###########################

```
CREATE FUNCTION complex_in(cstring)
    RETURNS complex
    AS 'filename'
    LANGUAGE C IMMUTABLE STRICT;

CREATE FUNCTION complex_out(complex)
    RETURNS cstring
    AS 'filename'
    LANGUAGE C IMMUTABLE STRICT;

CREATE FUNCTION complex_recv(internal)
   RETURNS complex
   AS 'filename'
   LANGUAGE C IMMUTABLE STRICT;

CREATE FUNCTION complex_send(complex)
   RETURNS bytea
   AS 'filename'
   LANGUAGE C IMMUTABLE STRICT;
```

##########################

```
CREATE TYPE complex (
   internallength = 16,
   input = complex_in,
   output = complex_out,
```

```
   receive = complex_recv,
   send = complex_send,
   alignment = double
);
```

 #############PostgreSQL###################### #########################_#######
#####

#####
######################

##4####char[4]################
################vl_len_#######   SET_VARSIZE()##########################
##VARSIZE()################## (##
####)

#######CREATE TYPE################

36.13.1. TOAST###

################################TOAST#################65.2########## #######
#########TOAST###
###########

TOAST#####################C######PG_DETOAST_DATUM##########TOAST########
########## ###############GETARG_DATATYPE_P############## ####CREATE TYPE#
##############variable#######plain######################

##PG_DETOAST_DATUM########
############ ####PG_DETOAST_DATUM_PACKED############GETARG_DATATYPE_PP#
################# ####VARSIZE_ANY_EXHDR###VARDATA_ANY####################
################ ## #####
#######PG_DETOAST_DATUM###################

##
###########vl_len_#char[4]####int32########## #############int32#### ######################### ## ### ### ###

TOAST###
############# ###########varlena#####################blob#####################
########################   ##
##PostgreSQL###
#####

###################src/include/utils/expandeddatum.h####################
######varlena#######################################varlena#####################
#######C###
######## ##PG_DETOAST_DATUM#####
#################################### ########varlena##########################
###

#################C##########2###varlena###
####### ###
##############################
##varlena########################### #

##########################varlena#######################################
################ ###########varlena#################################### #####
#######################VARATT_IS_EXPANDED_HEADER()########

TOAST#########varlena###
################ ##C##################
################## ##################C#######################################
###
#############C################################ #######################C#
##

####################src/backend/utils/adt/array_expanded.c#######

36.14.

################################ ##################################### #######
################# ## ##########
#############

PostgreSQL######################## ################ ############################
################### ##

###2########################## ##complex#####36.13###################### #######
########## ###############


```
CREATE FUNCTION complex_add(complex, complex)
    RETURNS complex
    AS 'filename', 'complex_add'
    LANGUAGE C IMMUTABLE STRICT;

CREATE OPERATOR + (
    leftarg = complex,
    rightarg = complex,
    function = complex_add,
    commutator = +
);
```

############################


```
SELECT (a + b) AS c FROM test_complex;

        c
-----------------
 (5.2,6.05)
 (133.42,144.95)
```

########################## ##############leftarg######### function######
###CREATE OPERATOR######## #####commutator##############################
commutator################################

36.15.

PostgreSQL## #########
######################
########## ## ###########
################## ##################################

PostgreSQL################################## ###################17beta3#########

```
########################################################## ###########36.11#
########
```

36.15.1. COMMUTATOR

```
COMMUTATOR#################################### #########x#y#####(x A y)#(y B
x)########A####B############## ###B#A#################### #################
#<#>############## ########+############## #########-############

################################################ ######PostgreSQL############
######################COMMUTATOR################

######################################### ####################################
##################### ####tab1.x  =  tab2.y####WHERE################ #
##tab1.x#tab2.y########tab2.y################### ############tab2.y  =
tab1.x#################################### ##############################
############### PostgreSQL#################### =############################
#####################
```

36.15.2. NEGATOR

```
NEGATOR################################## ###x#y##########################(x
A y)#NOT (x B y)#########A####B############ ###B#A################### #######
######<#>=########### ########################
```

```
################################### ######x####(A x)#NOT (B x)############

################################################  #####COMMUTATOR###########
#NEGATOR############
```

```
NOT (x = y)#####x <> y################################################## #
#########NOT####################################
```

36.15.3. RESTRICT

```
RESTRICT##################################      ##########################
RESTRICT##boolean################### ###########################WHERE#


column OP constant

########################## #######WHERE#####################################
####### ######################################COMMUTATOR#######1########

##############################################################################
### #########################

=##eqsel
<>##neqsel
<##scalarltsel
<=##scalarlesel
>##scalargtsel
>=##scalargesel

##############################################eqsel###neqsel##################
##########################################eqsel######

########################################
##scalarltsel#scalarlesel#scalargtsel#scalargesel############ ####src/
backend/utils/adt/selfuncs.c#convert_to_scalar()#####################
```

##########pg_type##
##

##1################matchingsel##
######### ########eqsel################2################################ (#
#####generic_restriction_selectivity############################)

###src/backend/utils/adt/geo_selfuncs.c################
#areasel#positionsel#contsel##### ######################################
###########

36.15.4. JOIN

JOIN################################## ############################## JOIN#
#boolean################### ######################WHERE#

```
table1.column1 OP table2.column2
```

############################ RESTRICT#######################################
##

############################## ######################################

=##eqjoinsel
<>##neqjoinsel
<##scalarltjoinsel
<=##scalarlejoinsel
>##scalargtjoinsel
>=##scalargejoinsel
#########matchingjoinsel
2############areajoinsel
2############positionjoinsel
2##############contjoinsel

36.15.5. HASHES

HASHES## HASHES#
#boolean################ ##

2############
###################
##HASHES##################### ##
##
################## ##

HASHES## ############################
###################### ## ####
##############################
#######

###################### ## #############
##
#######hash_any########### ########IEEE####################0####0###########
######################### ############0####################0###################
########

##################################
#################### ############################## #######################
######################

##

##################immutable####stable########## volatile################### ###########

##

#####################strict###############2###NULL################## NULL################# ##########IN################################ ## #######NULL############IN############ #####NULL#################### ###############

36.15.6. MERGES

MERGES## MERGES# #boolean################ ##

####################2################ ## ##smallint#integer##################

MERGES################btree################################ ################################## ## ######MERGES##

################btree### ###

##

##################immutable####stable########## volatile################### ###############

36.16.

##########################B-tree##

1########################### ######

36.16.1.

############B-##########GIN#################### ##################CREATE ACCESS METHOD####### ####62##########

##########################1############ ########################WHERE############## ################################## ##########WHERE######################################

######1#######################
################### ####B-tree################################### #############
###############B-tree####################B-tree########################### ######
####1###

###B-tree####################
`#int4_ops`######################### ######################################

36.16.2.

####################################### ######################################
############## ####B-tree################################## ######B-tree#######
############################# PostgreSQL#################PostgreSQL#########
#<#>=####################################### ################################
######################### ##
##############

36.3#######B-tree##################5#######

#36.3 B-tree#####

##	#######
###	1
##	2
###	3
##	4
###	5

######################### ####### 36.4#############1#########

#36.4

##	#######
###	1

GiST############## ###################### ########GiST########consistent##########
##################### #####2######################R-tree###############GiST####
############## 36.5###### ###4##2######################### ####4##X##########4##
Y###################

#36.5 GiST###2####R-tree######

##	#######
#####	1
#########	2
###	3
#########	4
#####	5
##	6
##	7
####	8
#########	9

##	#######
#####	10
#####	11
#########	12

SP-GiST##############GiST#################### ################################ ################### ################################### 36.6######

#36.6 SP-GiST###########

##	#######
#####	1
#####	5
##	6
#####	8
####	10
####	11

GIN################################GiST###SP-GiST############# ################ ################################### ################################### 36.7#### ##

#36.7 GIN

##	#######
##	1
##	2
#####	3
###	4

BRIN################################GiST#SP-GiST###GIN############# ######### ## ##########Minmax################### ## 36.8######

#36.8 BRIN Minmax#####

##	#######
###	1
##	2
###	3
##	4
###	5

############################ ##################WHERE############################ ################################boolean########## ############################ ################### ########36.16.7########

36.16.3.

###B-tree###########2##

#######SQL########################## ###

############

#options####################################

36.9#######B-tree###########################4#################################### ############ ##64.1.3################

#36.9 B-tree######

##	######
2###############2#########################0###0#####0###### ######	1
C####################################	2
###	3
################B-tree##################################	4
##############################	5

36.10##2####################

#36.10

##	######
###32###########	1
64bit###############64############### ####0#######32######1# #########################	2
##############################	3

36.11#######GiST########11##################6######### (#######64.2##########)

#36.11 GiST######

##	##	######
consistent	#########################	1
union	##############	2
compress	###################################	3
decompress	###########################	4
penalty	### ###	5
picksplit	## ########	6
same	2###################	7
distance	###########################	8
fetch	### ###	9
options	##############################	10

##	##	######
sortsupport	################################	11

36.12#######SP-GiST########6##################1######### (#######64.3##########)

#36.12 SP-GiST######

##	##	######
config	###################	1
choose	########################	2
picksplit	################	3
inner_consistent	################################	4
leaf_consistent	########################	5
options	################################	6

36.13#######GIN#########7##################4######### (#######64.4##########)

#36.13 GIN######

##	##	######
compare	2#########0###0#0############# ####### #####################	1
extractValue	######################	2
extractQuery	################	3
consistent	######################(2####)# (######6 #########)	4
comparePartial	##################################GIN# ## ## ##############################	5
triConsistent	######################(3####)# (######4 #########)	6
options	################################	7

36.14######BRIN#########5#####################1######### ##################### ########## (#######64.5.3##########)

#36.14 BRIN######

##	##	######
opcInfo	################################	1
add_value	#########################	2
consistent	########################	3
union	2################	4
options	################################	5

##B-tree#################### ######### ########################### B-tree## ##########GIN#SP-GiST#GiST####BRIN####################

36.16.4.

################### ############################ ######################src/tutorial/complex.c#src/tutorial/complex.sql########## ################ ######################### #########complex_abs_ops####### ################ ##########36.14######## B-tree######################

- ################1#
- ###############2#
- ################3#
- ###############4#
- ################5#

#####################################B-tree##################################1## ################ ################################## ###################

```
#define Mag(c)  ((c)->x*(c)->x + (c)->y*(c)->y)

static int
complex_abs_cmp_internal(Complex *a, Complex *b)
{
    double      amag = Mag(a),
                bmag = Mag(b);

    if (amag < bmag)
        return -1;
    if (amag > bmag)
        return 1;
    return 0;
}
```

####################

```
PG_FUNCTION_INFO_V1(complex_abs_lt);

Datum
complex_abs_lt(PG_FUNCTION_ARGS)
{
    Complex    *a = (Complex *) PG_GETARG_POINTER(0);
    Complex    *b = (Complex *) PG_GETARG_POINTER(1);

    PG_RETURN_BOOL(complex_abs_cmp_internal(a, b) < 0);
}
```

##4##################################

###################SQL#######

```
CREATE FUNCTION complex_abs_lt(complex, complex) RETURNS bool
    AS 'filename', 'complex_abs_lt'
    LANGUAGE C IMMUTABLE STRICT;

CREATE OPERATOR < (
   leftarg = complex, rightarg = complex, procedure = complex_abs_lt,
   commutator = > , negator = >= ,
   restrict = scalarltsel, join = scalarltjoinsel
);
```

##
#######

###################

- ####complex##########=#########1########## #####complex####=########
  ## ##=######### ###
  ##complex_abs_eq#######################

- PostgreSQL#################SQL################C##1#####1################ #
  ####C###abs_eq#################### ######################C###############
  #######

- abs_eq###SQL###PostgreSQL####################SQL#########################
  ## #################C####SQL###############

########B-tree################## #######C######################## ###############
########

```
CREATE FUNCTION complex_abs_cmp(complex, complex)
    RETURNS integer
    AS 'filename'
    LANGUAGE C IMMUTABLE STRICT;
```

################################## #####################

```
CREATE OPERATOR CLASS complex_abs_ops
    DEFAULT FOR TYPE complex USING btree AS
        OPERATOR        1       < ,
        OPERATOR        2       <= ,
        OPERATOR        3       = ,
        OPERATOR        4       >= ,
        OPERATOR        5       > ,
        FUNCTION        1       complex_abs_cmp(complex, complex);
```

######### ###complex##B-tree##########################

#################################

```
        OPERATOR        1       < (complex, complex) ,
```

##

#####################complex##########B-tree######################## #########
##DEFAULT###############

36.16.5.

#################1####################### ##################1##################
#################################### ##
######################## SQL##
##################################B-tree#############

#############PostgreSQL################ #####1################### ###############
###########################
###################### ##########1#######################################

#################################### ################################## #####
####################################

###############
##############################
############### ##
####### ##

###PostgreSQL##integer_ops#######B-tree######### ####bigint (int8)#integer (int4)#smallint (int2)####################int8_ops#int4_ops#int2_ops#####
############ ############2## ############
################################### ########################

```
CREATE OPERATOR FAMILY integer_ops USING btree;

CREATE OPERATOR CLASS int8_ops
DEFAULT FOR TYPE int8 USING btree FAMILY integer_ops AS
  -- standard int8 comparisons
  -- ##int8##
  OPERATOR 1 < ,
  OPERATOR 2 <= ,
  OPERATOR 3 = ,
  OPERATOR 4 >= ,
  OPERATOR 5 > ,
  FUNCTION 1 btint8cmp(int8, int8) ,
  FUNCTION 2 btint8sortsupport(internal) ,
  FUNCTION 3 in_range(int8, int8, int8, boolean, boolean) ,
  FUNCTION 4 btequalimage(oid) ;

CREATE OPERATOR CLASS int4_ops
DEFAULT FOR TYPE int4 USING btree FAMILY integer_ops AS
  -- standard int4 comparisons
  -- ##int4##
  OPERATOR 1 < ,
  OPERATOR 2 <= ,
  OPERATOR 3 = ,
  OPERATOR 4 >= ,
  OPERATOR 5 > ,
  FUNCTION 1 btint4cmp(int4, int4) ,
  FUNCTION 2 btint4sortsupport(internal) ,
  FUNCTION 3 in_range(int4, int4, int4, boolean, boolean) ,
  FUNCTION 4 btequalimage(oid) ;

CREATE OPERATOR CLASS int2_ops
DEFAULT FOR TYPE int2 USING btree FAMILY integer_ops AS
  -- standard int2 comparisons
  -- ##int2##
  OPERATOR 1 < ,
  OPERATOR 2 <= ,
  OPERATOR 3 = ,
  OPERATOR 4 >= ,
  OPERATOR 5 > ,
  FUNCTION 1 btint2cmp(int2, int2) ,
  FUNCTION 2 btint2sortsupport(internal) ,
  FUNCTION 3 in_range(int2, int2, int2, boolean, boolean) ,
  FUNCTION 4 btequalimage(oid) ;

ALTER OPERATOR FAMILY integer_ops USING btree ADD
  -- cross-type comparisons int8 vs int2
  -- ###### int8#int2
  OPERATOR 1 < (int8, int2) ,
  OPERATOR 2 <= (int8, int2) ,
  OPERATOR 3 = (int8, int2) ,
  OPERATOR 4 >= (int8, int2) ,
  OPERATOR 5 > (int8, int2) ,
  FUNCTION 1 btint82cmp(int8, int2) ,
```

```
-- cross-type comparisons int8 vs int4
-- ###### int8#int4
OPERATOR 1 < (int8, int4) ,
OPERATOR 2 <= (int8, int4) ,
OPERATOR 3 = (int8, int4) ,
OPERATOR 4 >= (int8, int4) ,
OPERATOR 5 > (int8, int4) ,
FUNCTION 1 btint84cmp(int8, int4) ,

-- cross-type comparisons int4 vs int2
-- ###### int4#int2
OPERATOR 1 < (int4, int2) ,
OPERATOR 2 <= (int4, int2) ,
OPERATOR 3 = (int4, int2) ,
OPERATOR 4 >= (int4, int2) ,
OPERATOR 5 > (int4, int2) ,
FUNCTION 1 btint42cmp(int4, int2) ,

-- cross-type comparisons int4 vs int8
-- ###### int4#int8
OPERATOR 1 < (int4, int8) ,
OPERATOR 2 <= (int4, int8) ,
OPERATOR 3 = (int4, int8) ,
OPERATOR 4 >= (int4, int8) ,
OPERATOR 5 > (int4, int8) ,
FUNCTION 1 btint48cmp(int4, int8) ,

-- cross-type comparisons int2 vs int8
-- ###### int2#int8
OPERATOR 1 < (int2, int8) ,
OPERATOR 2 <= (int2, int8) ,
OPERATOR 3 = (int2, int8) ,
OPERATOR 4 >= (int2, int8) ,
OPERATOR 5 > (int2, int8) ,
FUNCTION 1 btint28cmp(int2, int8) ,

-- cross-type comparisons int2 vs int4
-- ###### int2#int4
OPERATOR 1 < (int2, int4) ,
OPERATOR 2 <= (int2, int4) ,
OPERATOR 3 = (int2, int4) ,
OPERATOR 4 >= (int2, int4) ,
OPERATOR 5 > (int2, int4) ,
FUNCTION 1 btint24cmp(int2, int4) ,

-- cross-type in_range functions
FUNCTION 3 in_range(int4, int4, int8, boolean, boolean) ,
FUNCTION 3 in_range(int4, int4, int2, boolean, boolean) ,
FUNCTION 3 in_range(int2, int2, int8, boolean, boolean) ,
FUNCTION 3 in_range(int2, int2, int4, boolean, boolean) ;
```

##

###64.1.2##########B-tree### ################ ########2############################ ### ########### ##

######################################2################################### ### ##

####1############1########### ######### ################### ### ######

GiST#SP-GiST#GIN################################### ############################ ###############

BRIN############################### `minmax`#########################B-tree###### ##

> ##
>
> PostgreSQL8.3############################## ############################### ################################### ##################################### ##

36.16.6.

PostgreSQL## ########## ##

`######ORDER BY#DISTINCT`################SQL####### ################ ##########PostgreSQL##############B-tree############ ####################`GROUP BY#DISTINCT`###`ORDER BY`##### #####

############B-tree## ##################### ################################

################################SQL## #####

> ##
>
> PostgreSQL#####7.4#######################=#<#>################## ####### ##

`##############USING`######################B-tree################## ##########

```
SELECT * FROM mytable ORDER BY somecol USING ~<~;
```

`####USING`################################

`####################`B-tree#################### ######B-tree######################## ##################################

`####################`SQL###############4.2.8##########`RANGE` *`offset`* `PRECEDING/FOLLOWING`############## ###############

```
SELECT sum(x) OVER (ORDER BY x RANGE BETWEEN 5 PRECEDING AND 10 FOLLOWING)
  FROM mytable;
```

`########x`#################### ######################################`x####5`## ####10#################### `ORDER BY`#######B-tree############################## ##x################## ## #######

#####B-tree## #####B-tree######### ###########################*in_range*################### RANGE#################### ################in_range################### ######ORDER BY########B-tree######## ####in_range##############RANGE *offset* PRECEDING/FOLLOWING###############

########################

36.16.7.

######################GiST#SP-GiST######################### ################## ### #######WHERE *indexed_column operator constant*########################## ############ ##################################### ########################### ############## ########ORDER BY *indexed_column operator constant*######## ############################### ## ###### ####################

```
SELECT * FROM places ORDER BY location <-> point '(101,456)' LIMIT 10;
```

################10########### <->#############location###GiST################### #####

##B-tree############### ##########B-tree#####PostgreSQL############## ############### point####<->####float8####################################

```
OPERATOR 15    <-> (point, point) FOR ORDER BY float_ops
```

###float_ops##float8##################### #############<->################## ##############

36.16.8.

##################2############# ## ####

###WHERE################## ######### ##########

```
SELECT * FROM table WHERE integer_column < 4;
```

#########B-tree################################ ################################## ############ ####GiST################################## ##################### #############WHERE################## ## ### ################### ################### ##*recheck*############ ################## ## ### ### ################ ######################GiST#SP-GiST###GIN################### recheck################## #####################

########box######################### #########CREATE OPERATOR CLASS#STORAGE####################

```
CREATE OPERATOR CLASS polygon_ops
    DEFAULT FOR TYPE polygon USING gist AS
        ...
        STORAGE box;
```

######GiST#SP-GiST#GIN###BRIN#####################STORAGE########### STORAGE########GiST#compress###decompress######################### SP-GiST#####STORAGE##############compress############# SP-GiST############## ###########consistent############## GIN###STORAGE################ ######### ################# ##################################### GIN#extractValue## #extractQuery##################################### BRIN#GIN######STORAGE## ##

36.17.

PostgreSQL##############SQL############## ################################### ############## ## PostgreSQL################### ################################SQL########### ################################## #####C##########C##################### ### ############CREATE EXTENSION###########################

############SQL##PostgreSQL###### ###################### ###DROP EXTENSION################################### ################# #########pg_dump## ### #######CREATE EXTENSION############ ##################################### ######################### ## ########################

PostgreSQL## ###################### ########CREATE OR REPLACE FUNCTION#######################pg_dump######## ############## ## ################### #############36.17.3########## ## ########

#########GRANT##REVOKE################################ ###################### ##############pg_init_privs############### pg_dump########CREATE EXTENSION##GRANT##REVOKE#### #####

PostgreSQL############CREATE POLICY##SECURITY LABEL############### ####### ################ ##################################pg_dump##################

###############SQL################################### ############1.1######1.0### ## ###ALTER EXTENSION UPDATE## ###

##########SQL#########ALTER EXTENSION###### ############################ ## ################ ## ############################## ##################################### ### ########################### ### ######## ##

##

CREATE EXTENSION###################.control########################### ###### ##################SHAREDIR/extension################### ######## ##*extension--version*.sql##########foo#######1.0##foo--1.0.sql####SQL ##################### ##################SHAREDIR/extension############## ################################

################postgresql.conf######### #####*parameter_name* = *value*##### ############# ##################### ##################################

#####################

directory (string)

###SQL################### ##############################SHAREDIR####### ########## ########directory = 'extension'############

default_version (string)

##############CREATE EXTENSION############################### ######## ######VERSION#######CREATE EXTENSION###### ####################

comment (string)

################### ### ########### ###################COMMENT###################

encoding (string)

########################### ##############ASCII#################### ##### ############################

module_pathname (string)

####################MODULE_PATHNAME############ ################### ### ######################################$libdir/*shared_library_name*### ###C######CREATE FUNCTION######MODULE_PATHNAME######

requires (string)

################## ###requires = 'foo, bar'### ######################## ########################

no_relocate (string)

############ALTER EXTENSION ... SET SCHEMA########################### ############(@extschema:*name*@######)################################

superuser (boolean)

#######true###trusted## ######## false## ##### ############################true########## ############################ ########################

trusted (boolean)

########true##################################superuser#true######## ################ ###############CREATE############################## CREATE EXTENSION## ## superuser#false######### ###### ##true####

####### ######trusted###
##36.17.6##########

relocatable (boolean)

######false##########
########### ####36.17.2##########

schema (string)

################################ ####################################
schema### ####36.17.2##########

#######*extension*.control######*extension--version*.control##############
########### ## ####################
##
#directory###default_version#########################

###SQL#########################BEGIN#COMMIT##############################
#VACUUM#########SQL############### ##################################

###SQL############\echo################# ############################## #####
#######CREATE EXTENSION#36.17.7##############psql############################
########### ##
############

############@extowner@################CREATE EXTENSION#ALTER EXTENSION#
############################## ###########trusted################################
################################## ############################C##############
####################################

##PostgreSQL##########################
############ASCII################ ##############ASCII####################
################comment##############################COMMENT ON
EXTENSION############

36.17.2.

####################
##

- ### ###################
  #################ALTER EXTENSION SET SCHEMA######## ##################
  ############################## ##
  ########################### ########relocatable = true###################
  ####

- ###################################### ###################SQL###
  #search_path################################ ################relocatable
  = false##############################@extschema@######### #
  ############################(###############)################## ####CREATE
  EXTENSION#SCHEMA#######################

- ############################relocatable = false#####
  ###schema######################### ##################################
  ##CREATE EXTENSION#SCHEMA############## ###########@extschema@########
  ################################ @extschema@##############################
  #####################

##search_path########### ###CREATE
EXTENSION##############

```
SET LOCAL search_path TO @extschema@, pg_temp;
```

```
########################################## ###############search_path######
############# CREATE EXTENSION#####search_path############
```

```
###############schema########################          ############CREATE
EXTENSION#SCHEMA##############################################
#search_path############ ######schema#####################################
#########################
```

```
####################requires##############################################
#search_path############ ######################################
```

```
###############pg_temp#####search_path##########
```

```
##########################################################################
## ######################### ########################search_path##########
####
```

```
##################################################         ###################
#@extschema:name@###########name#########(#########requires#############
####)# ######################(#############)######## ########################
##########################################################################
#######@extschema:name@######search_path############################ ####
#################OID#################### ################################
##############################################no_relocate#########
```

36.17.3.

```
##########################################################  ##################
###########pg_dump########### ######################### ######################
########### ##############################
```

```
#############################################################pg_dump##########
######################################     ################################
#pg_extension_config_dump(regclass, text)############
```

```
CREATE TABLE my_config (key text, value text);
CREATE SEQUENCE my_config_seq;

SELECT pg_catalog.pg_extension_config_dump('my_config', '');
SELECT pg_catalog.pg_extension_config_dump('my_config_seq', '');
```

```
################################## serial####bigserial######################
#####
```

```
pg_extension_config_dump##########################pg_dump###########   ##
##############################################  ############################
##pg_extension_config_dump######################WHERE######### ##########
```

```
CREATE TABLE my_config (key text, value text, standard_entry boolean);

SELECT pg_catalog.pg_extension_config_dump('my_config', 'WHERE NOT
 standard_entry');
```

```
##########################standard_entry##############
```

```
##########pg_extension_config_dump################
```

##
###############

pg_extension_config_dump#################################### ###########
############### ##########################ALTER EXTENSION ... DROP TABLE#
#######################

####################pg_dump################## ###pg_dump###################
############### #######CREATE EXTENSION#(###############)#################
####### ##

serial####bigserial## ##############
################

36.17.4.

#########################SQL############################ ###################
##################################
#######
#*extension--old_version--target_version*.sql###############
##foo--1.0--1.1.sql#foo#######1.0######1.1###########################

####################ALTER EXTENSION UPDATE##########################
####### ########CREATE EXTENSION################################ ####
#search_path## ##############
##

###

ALTER EXTENSION################################## ##
#foo--1.0--1.1.sql#foo--1.1--2.0.sql###############1.0###############
###2.0############ALTER EXTENSION#############

PostgreSQL############################# ###1.0###1.1############## ##############
##########--###########-########
###############

################################ ###foo--1.1--1.0.sql######1.1############
####### ### ##############
####################
##################################
####################

#####################################

SELECT * FROM pg_extension_update_paths('*extension_name*');

##
#####NULL########## ###--########################## ##########
#regexp_split_to_array(path,'--')#######

36.17.5.

#######foo#
#####1.0#1.1#1.2##################
#foo--1.0--1.1.sql#foo--1.1--1.2.sql############# PostgreSQL 10##
##############################foo--1.1.sql#foo--1.2.sql############## #
#################################1.0########################## ###########
######### #######CREATE EXTENSION############################

#####foo--1.0.sql#foo--1.0--1.1.sql#foo--1.1--1.2.sql############ #1.2### ########1.0#########1.2#### ############# #ALTER EXTENSION UPDATE######################### ##########

##foo--1.0.control#requires = 'bar'#########foo################### ##1.0###############bar#################

36.17.6.

####################

superuser## ## ##

###trusted### ############################ ######trusted#################################### ############################

################################ 36.17.6.1 ##################################### #### 36.17.6.2

36.17.6.1.

##########SQL###PL################################### ######################### #####

CREATE FUNCTION###########SECURITY DEFINER######################### ##### ##

search_path## ###### ####search_path################ ####IN# CASE # WHEN################## ### ########OPERATOR(####.=) ANY#CASE WHEN ##########

myschema.myfunc(bigint)############ ##myschema.myfunc(42)#################myschema.myfunc(integer)######## ######

36.17.6.2.

##

CREATE FUNCTION#CREATE OPERATOR CLASS###DDL############################ ########### ####CREATE FUNCTION#DEFAULT########CREATE VIEW#########

###########SQL################# ####DDL################# ############ #search_path################### CREATE/ALTER EXTENSION################### ###### ##########search_path#'pg_catalog, pg_temp'###################### ################(######################)# #######PostgreSQL#######contrib#### ########

############################ ################################## ########## ######################### ##################search_path################## ####### ############################## ################################## #pg_catalog###################trusted###########

36.17.7.

#####SQL################ #k###v## #
##########################

pair--1.0.sql####################

```
-- complain if script is sourced in psql, rather than via CREATE EXTENSION
-- #######CREATE EXTENSION#######psql####################
\echo Use "CREATE EXTENSION pair" to load this file. \quit

CREATE TYPE pair AS ( k text, v text );

CREATE FUNCTION pair(text, text)
RETURNS pair LANGUAGE SQL AS 'SELECT ROW($1, $2)::@extschema@.pair;';

CREATE OPERATOR ~> (LEFTARG = text, RIGHTARG = text, FUNCTION = pair);

-- "SET search_path" is easy to get right, but qualified names perform
 better.
CREATE FUNCTION lower(pair)
RETURNS pair LANGUAGE SQL
AS 'SELECT ROW(lower($1.k), lower($1.v))::@extschema@.pair;'
SET search_path = pg_temp;

CREATE FUNCTION pair_concat(pair, pair)
RETURNS pair LANGUAGE SQL
AS 'SELECT ROW($1.k OPERATOR(pg_catalog.||) $2.k,
               $1.v OPERATOR(pg_catalog.||) $2.v)::@extschema@.pair;';
```

pair.control#################

```
# pair extension
comment = 'A key/value pair data type'
default_version = '1.0'
# cannot be relocatable because of use of @extschema@
relocatable = false
```

################################makefile#######################Makefile######
##

```
EXTENSION = pair
DATA = pair--1.0.sql

PG_CONFIG = pg_config
PGXS := $(shell $(PG_CONFIG) --pgxs)
include $(PGXS)
```

##makefile#36.18#####PGXS###### make install#####################pg_config####
#####################

################CREATE EXTENSION#####################################

36.18.

PostgreSQL## ####
PostgreSQL###PGXS###########
########## PGXS###C#################SQL################ PGXS#PostgreSQL#####

```
########################################################## ####################
############## #######################################

######PGXS##############makefile############        ##makefile##################
PGXSmakefile################ ###isbn_issn######################### ############
#C#######################SQL##############################SQL##############
########################################


MODULES = isbn_issn
EXTENSION = isbn_issn
DATA = isbn_issn--1.0.sql
DOCS = README.isbn_issn
HEADERS_isbn_issn = isbn_issn.h

PG_CONFIG = pg_config
PGXS := $(shell $(PG_CONFIG) --pgxs)
include $(PGXS)

############ ##################make############

#########################

MODULES

   ################################ ###########################

MODULE_big

   ########################## #OBJS##################

PROGRAM

   ############## #OBJS##################

############

EXTENSION

   ######## ########prefix/share/extension#########extension.control###
   ###########

MODULEDIR

   DATA###DOCS#################prefix/share##########     ##############
   #EXTENSION##########extension############contrib######

DATA

   prefix/share/$MODULEDIR##################

DATA_built

   prefix/share/$MODULEDIR################################

DATA_TSEARCH

   prefix/share/tsearch_data####################

DOCS

   prefix/doc/$MODULEDIR####################
```

```
HEADERS
HEADERS_built

   #############prefix/include/server/$MODULEDIR/$MODULE_big###########
   ######

   DATA_built#####HEADERS_built######clean############################
   #EXTRA_CLEAN##########################

HEADERS_$MODULE
HEADERS_built_$MODULE

   ################prefix/include/server/$MODULEDIR/$MODULE############
   #######$MODULE#MODULES#MODULE_big######################

   DATA_built#####HEADERS_built_$MODULE######clean####################
   #########EXTRA_CLEAN##########################

   ######################################MODULES###########built_##########
   ##################################### ####################HEADERS_built_
   $MODULE###########

SCRIPTS

   prefix/bin####################################

SCRIPTS_built

   prefix/bin#####################################################

REGRESS

   ########################### ######

REGRESS_OPTS

   pg_regress############

ISOLATION

   ############ #########

ISOLATION_OPTS

   pg_isolation_regress#############

TAP_TESTS

   TAP############################ ######

NO_INSTALL

   install############ ##############################

NO_INSTALLCHECK

   installcheck############ ###################pg_regress################

EXTRA_CLEAN

   make clean##############
```

PG_CPPFLAGS

CPPFLAGS##########

PG_CFLAGS

CFLAGS#######

PG_CXXFLAGS

CXXFLAGS#######

PG_LDFLAGS

LDFLAGS##########

PG_LIBS

PROGRAM############

SHLIB_LINK

MODULE_big###########

PG_CONFIG

#####PostgreSQL###########pg_config########### ####PATH#########pg_config## #########

##makefile#Makefile######################### #############make############# ######make install########## #######PATH#########pg_config### #######PostgreSQL############################## makefile###make############# #PG_CONFIG###pg_config#######################################

##make############ #####*VPATH*######## ## ###########

```
mkdir build_dir
cd build_dir
make -f /path/to/extension/source/tree/Makefile
make -f /path/to/extension/source/tree/Makefile install
```

#################VPATH################# #######1#############config/prep_buildtree######## ########make##VPATH####################

```
make VPATH=/path/to/extension/source/tree
make VPATH=/path/to/extension/source/tree install
```

############################

REGRESS##############make install######make installcheck############## ########################## ###########PostgreSQL################## REGRESS## ########################sql/############################ #########.sql######## ########## ######makefile##REGRESS########### #######expected/############## ###################.out########################make installcheck#psql###### ################################### #######diff -c###regression.diffs##### #### ################################### ###########################

ISOLATION##############make install######make installcheck# ## ###########PostgreSQL######## ########## ISOLATION#########################specs/######################

```
######  ########.spec#################  ######makefile##ISOLATION#########
## #######expected/################################.out##################
##### make  installcheck######################################### #######diff
-c###output_iso/regression.diffs######## ################################
## ############################

TAP_TESTS#TAP############# ############tmp_check/#################### ##
#31.4##########
```

```
                                   ###

#######################################################   #REGRESS#####
##results/#########ISOLATION######output_iso/results/###########
################################expected/###########
```

#37#

########################### #######PL/pgSQL (#41#)#PL/Tcl (#42#)#PL/Perl (#43#)#PL/Python (#44#)############################## ################################### ##############

###C#################### ### ######### SQL#######################

37.1.

##################################

####################INSERT#UPDATE###DELETE##################SQL########### ############ ####UPDATE###############UPDATE##SET######################### ## #######TRUNCATE############# ######################################

###########INSERT#UPDATE###DELETE###################### ####INSTEAD OF### ################################### ################################### ################################ #########SQL#####INSERT#UPDATE###DELETE## ###################### ################INSTEAD OF################# INSTEAD OF## ################ ###################

################################## ##############trigger################### # ############################TriggerData##############

###############CREATE TRIGGER################### ########################## ##

PostgreSQL########################## ### #### ## ############# ########################## ##2#################################### TRUNCATE# ##################################

#######BEFORE####AFTER###### #INSTEAD OF######### #####BEFORE############################AFTER######### ######### ##################################### #####BEFORE################## ########AFTER#####################AFTER############# ##################### ################# INSTEAD OF########################## #####################

#################AFTER## ###### ## ########

######INSERT###ON######DO#######-####INSERT######################## #INSERT##############EXCLUDED## ################

############-################ON######DO################# ################ ######### ##################### ON######DO######INSERT#############-## ##INSERT##################################-###AFTER############### #-AFTERINSERT######### ###

############################

######################UPDATE#####################################DELETE#####
##########INSERT############# ############BEFORE UPDATE####BEFORE DELETE#
################## ###########BEFORE INSERT#################### ##
AFTER ROW######
##AFTER DELETE#AFTER INSERT#############AFTER UPDATE#############
###UPDATE#DELETE#INSERT########## ############################DELETE###
#INSERT###########UPDATE################UPDATE############

MERGE################ ################UPDATE#DELETE###INSERT#####MERGE###
############(##########)############(##########)###########

MERGE#############BEFORE###AFTER############################MERGE######
###################### ##########UPDATE###################### #############
########################## ##
##########################

#######################NULL############ ##################################
######HeapTuple#################### ############################

- NULL######################## ##
  ###########

- #####INSERT###UPDATE###################################### ##############
  ##########################

#################BEFORE##################INSERT###UPDATE#####NEW##DELETE#
###OLD##########################

#####INSTEAD OF#####################################NULL#############
#INSERT#UPDATE#####NEW##DELETE#####OLD############## #NULL###############
################################ ########################### INSERT#UPDATE##
##########NEW############# ###INSERT RETURNING###UPDATE RETURNING######
##

##################################NULL##########

################### #########BEFORE######AFTER############ #########
##AFTER############## BEFORE################OLD##################NEW#
################################## C#####################################
##BEFORE######NEW################### BEFORE#########################

##################1## BEFORE####INSTEAD
OF## #####BEFORE####INSTEAD OF###
#NULL######(######)#######################

###################WHEN################## ###########WHEN##########(####)###
############# (######################WHEN############) BEFORE###############
#################WHEN################### ######WHEN#########################
############### ###AFTER######WHEN####################(#####)###############
################## #######AFTER####WHEN#####################################
############# ##
######## INSTEAD OF####WHEN#############

#######BEFORE## ####BEFORE###
##timestamp######################2############################## #####AFTER##
#############
##AFTER################BEFORE################## #####BEFORE########
####### ####BEFORE####AFTER##
##BEFORE##########

######SQL################################### ################### ######## ############ ############################## ####INSERT#################################INSERT############ ##############################

###SQL######### ########### ################################### ############################## ############################## ######################

 ############################## ## ######### ####2## ########## ################################## ## ###############INSERT################ ###UPDATE############################## #######

#INSERT#UPDATE###CREATE TRIGGER################## ###############INSERT## #UPDATE#######NEW###UPDATE###DELETE#######OLD#######

#####AFTER STATEMENT##########(*transition tables*)################ AFTER ROW## ## ##SQL##################

37.2.

#######SQL### #### ##SQL## ###########

- ######################### #################BEFORE########### ########AFTER# ##############
- #########BEFORE#####SQL## ######
- ########BEFORE########SQL## # ################################## ########SQL############################
- ########INSTEAD OF######################INSTEAD OF######################
- #####AFTER###

##VOLATILE################## STABLE#####IMMUTABLE##

###############45.5###### 37.4############

37.3. C############

################################### #####C##################### ############### ################### ########C############################## ################### ###############

#######version 1##########################

#######################################TriggerData#######context############ C## ##

CALLED_AS_TRIGGER(fcinfo)

#############

((fcinfo)->context != NULL && IsA((fcinfo)->context, TriggerData))

###########fcinfo->context#TriggerData *###########TriggerData######## ####### ######TriggerData###########################

struct TriggerData#commands/trigger.h###########

```
typedef struct TriggerData
{
    NodeTag          type;
    TriggerEvent     tg_event;
    Relation         tg_relation;
    HeapTuple        tg_trigtuple;
    HeapTuple        tg_newtuple;
    Trigger         *tg_trigger;
    TupleTableSlot  *tg_trigslot;
    TupleTableSlot  *tg_newslot;
    Tuplestorestate *tg_oldtable;
    Tuplestorestate *tg_newtable;
    const Bitmapset *tg_updatedcols;
} TriggerData;
```

##################

type

##T_TriggerData###

tg_event

#################### tg_event#######################

TRIGGER_FIRED_BEFORE(tg_event)

########(before)##############

TRIGGER_FIRED_AFTER(tg_event)

########(after)##############

TRIGGER_FIRED_INSTEAD(tg_event)

####INSTEAD OF##############

TRIGGER_FIRED_FOR_ROW(tg_event)

##########################

TRIGGER_FIRED_FOR_STATEMENT(tg_event)

##########################

TRIGGER_FIRED_BY_INSERT(tg_event)

####INSERT##################

TRIGGER_FIRED_BY_UPDATE(tg_event)

####UPDATE##################

TRIGGER_FIRED_BY_DELETE(tg_event)

####DELETE##################

TRIGGER_FIRED_BY_TRUNCATE(tg_event)

####TRUNCATE##################

tg_relation

############################## #############utils/rel.h######### ###### ##tg_relation->rd_att##############tg_relation->rd_rel->relname## ###########char*####NameData### #############char*##### #SPI_getrelname(tg_relation)##########

tg_trigtuple

################## ########################## ##INSERT/DELETE########### ###################INSERT## ######## ##########################

tg_newtuple

####UPDATE############################ INSERT####DELETE#####NULL### UPDATE## ########## ########################

tg_trigger

######utils/reltrigger.h######Trigger###########

```
typedef struct Trigger
{
    Oid         tgoid;
    char       *tgname;
    Oid         tgfoid;
    int16       tgtype;
    char        tgenabled;
    bool        tgisinternal;
    bool        tgisclone;
    Oid         tgconstrrelid;
    Oid         tgconstrindid;
    Oid         tgconstraint;
    bool        tgdeferrable;
    bool        tginitdeferred;
    int16       tgnargs;
    int16       tgnattr;
    int16      *tgattr;
    char      **tgargs;
    char       *tgqual;
    char       *tgoldtable;
    char       *tgnewtable;
} Trigger;
```

####tgname########tgnargs#tgargs######tgargs#CREATE TRIGGER######### ########### #################

tg_trigslot

tg_trigtuple######################NULL#######

tg_newslot

tg_newtuple######################NULL#######

tg_oldtable

tg_relation###########0#######Tuplestorestate#############OLD TABLE# ###############NULL#######

tg_newtable

tg_relation###########0#######Tuplestorestate#############NEW TABLE# ###############NULL#######

tg_updatedcols

UPDATE#################################### ############################## #############

########attnum(1###)############################
##bms_is_member(attnum - FirstLowInvalidHeapAttributeNumber, trigdata->tg_updatedcols))#######

UPDATE#################NULL######

SPI##############################SPI_register_trigger_data#########

######HeapTuple#######NULL#####SQL#NULL####### ######*isNull*############# ############# ######################tg_trigtuple#tg_newtuple#############

37.4.

C############################## ####################################

trigf####ttest####################x#NULL########################## ####### ###################NOT NULL#############

###################

```
CREATE TABLE ttest (
    x integer
);
```

#################

```
#include "postgres.h"
#include "fmgr.h"

#include "executor/spi.h"       /* ###SPI############# */
#include "commands/trigger.h"   /* ############ */
#include "utils/rel.h"          /* ############### */

PG_MODULE_MAGIC;

PG_FUNCTION_INFO_V1(trigf);

Datum
trigf(PG_FUNCTION_ARGS)
{
    TriggerData *trigdata = (TriggerData *) fcinfo->context;
    TupleDesc   tupdesc;
    HeapTuple   rettuple;
    char       *when;
    bool        checknull = false;
    bool        isnull;
```

```
    int         ret, i;


    /* ################### */
    if (!CALLED_AS_TRIGGER(fcinfo))
        elog(ERROR, "trigf: not called by trigger manager");


    /* ############# */
    if (TRIGGER_FIRED_BY_UPDATE(trigdata->tg_event))
        rettuple = trigdata->tg_newtuple;
    else
        rettuple = trigdata->tg_trigtuple;


    /* NULL###### */
    if (!TRIGGER_FIRED_BY_DELETE(trigdata->tg_event)
        && TRIGGER_FIRED_BEFORE(trigdata->tg_event))
        checknull = true;

    if (TRIGGER_FIRED_BEFORE(trigdata->tg_event))
        when = "before";
    else
        when = "after ";

    tupdesc = trigdata->tg_relation->rd_att;


    /* SPI######## */
    if ((ret = SPI_connect()) < 0)
        elog(ERROR, "trigf (fired %s): SPI_connect returned %d", when,
 ret);


    /* ########### */
    ret = SPI_exec("SELECT count(*) FROM ttest", 0);

    if (ret < 0)
        elog(ERROR, "trigf (fired %s): SPI_exec returned %d", when, ret);


    /* count(*)#int8###############*/
    i = DatumGetInt64(SPI_getbinval(SPI_tuptable->vals[0],
                                    SPI_tuptable->tupdesc,
                                    1,
                                    &isnull));

    elog (INFO, "trigf (fired %s): there are %d rows in ttest", when, i);

    SPI_finish();

    if (checknull)
    {
        SPI_getbinval(rettuple, tupdesc, 1, &isnull);
        if (isnull)
            rettuple = NULL;
    }

    return PointerGetDatum(rettuple);
}
```

#############36.10.5################################

```
CREATE FUNCTION trigf() RETURNS trigger
    AS 'filename'
    LANGUAGE C;

CREATE TRIGGER tbefore BEFORE INSERT OR UPDATE OR DELETE ON ttest
    FOR EACH ROW EXECUTE FUNCTION trigf();

CREATE TRIGGER tafter AFTER INSERT OR UPDATE OR DELETE ON ttest
    FOR EACH ROW EXECUTE FUNCTION trigf();

######################


=> INSERT INTO ttest VALUES (NULL);
INFO:  trigf (fired before): there are 0 rows in ttest
INSERT 0 0


-- #############AFTER############

=> SELECT * FROM ttest;
 x
---
(0 rows)

=> INSERT INTO ttest VALUES (1);
INFO:  trigf (fired before): there are 0 rows in ttest
INFO:  trigf (fired after ): there are 1 rows in ttest
                                         ^^^^^^^^

                             #################
INSERT 167793 1
vac=> SELECT * FROM ttest;
 x
---
 1
(1 row)

=> INSERT INTO ttest SELECT x * 2 FROM ttest;
INFO:  trigf (fired before): there are 1 rows in ttest
INFO:  trigf (fired after ): there are 2 rows in ttest
                                         ^^^^^^

                             #################
INSERT 167794 1
=> SELECT * FROM ttest;
 x
---
 1
 2
(2 rows)

=> UPDATE ttest SET x = NULL WHERE x = 2;
INFO:  trigf (fired before): there are 2 rows in ttest
UPDATE 0
=> UPDATE ttest SET x = 4 WHERE x = 2;
INFO:  trigf (fired before): there are 2 rows in ttest
INFO:  trigf (fired after ): there are 2 rows in ttest
UPDATE 1
vac=> SELECT * FROM ttest;
 x
---
 1
 4
(2 rows)
```

```
=> DELETE FROM ttest;
INFO:  trigf (fired before): there are 2 rows in ttest
INFO:  trigf (fired before): there are 1 rows in ttest
INFO:  trigf (fired after ): there are 0 rows in ttest
INFO:  trigf (fired after ): there are 0 rows in ttest
                                         ^^^^^^

                              #################
DELETE 2
=> SELECT * FROM ttest;
 x
---
(0 rows)
```

src/test/regress/regress.c#spi##############

#38#

#37#####################PostgreSQL############## #############DML############# ################################DDL###############

######################SQL#######################C##############

38.1.

#####[84.196891]## ############# #####ddl_command_start#ddl_command_end#table_rewrite#sql_drop### ### ######################### ## ##### ####################DDL_####_#####DDL_####_end#####_rewrite### #SQL_drop##### ################################

######################event_triggers#################### ###### ### ################################## ##postgres################## ######################### ######################## ### ######################## ############# ################psql############################

ddl_command_start######CREATE#ALTER#DROP#SECURITY LABEL#COMMENT#GRANT#REVOKE################ ############################## ############## ################### — ################ — ########DDL######### ##################################### ddl_command_start####SELECT INTO### ####################CREATE TABLE AS############

ddl_command_end######ddl_command_start########################## ####DDL#### ##########ddl_command_end#########(9.30#########)####### #pg_event_trigger_ddl_commands()######## #############(################# #)##

sql_drop############################ddl_command_end################ ###### ##################sql_drop#########(9.30#########)# #pg_event_trigger_dropped_objects()################ ################### ##

####ALTER TABLE#ALTER TYPE#######################table_rewrite######## ## CLUSTER#VACUUM##############################table_rewrite##############

####DDL################## #ddl_command_end########### #####ddl_command_start#################### ###################### ######ddl_command_end############################## ######DDL############

##########################38.2#########

############CREATE EVENT TRIGGER########### ################### #event_trigger############### ## ############################

##

######WHEN#################ddl_command_start############################## ##### ################################DDL###############

38.2.

38.1################################

#38.1

######	ddl_command_start	ddl_command_end	sql_drop	table_rewrite	##
ALTER AGGREGATE	X	X	-	-	
ALTER COLLATION	X	X	-	-	
ALTER CONVERSION	X	X	-	-	
ALTER DOMAIN	X	X	-	-	
ALTER DEFAULT PRIVILEGES	X	X	-	-	
ALTER EXTENSION	X	X	-	-	
ALTER FOREIGN DATA WRAPPER	X	X	-	-	
ALTER FOREIGN TABLE	X	X	X	-	
ALTER FUNCTION	X	X	-	-	
ALTER LANGUAGE	X	X	-	-	
ALTER LARGE OBJECT	X	X	-	-	
ALTER MATERIALIZED VIEW	X	X	-	X	
ALTER OPERATOR	X	X	-	-	
ALTER OPERATOR CLASS	X	X	-	-	
ALTER OPERATOR FAMILY	X	X	-	-	
ALTER POLICY	X	X	-	-	
ALTER PROCEDURE	X	X	-	-	
ALTER PUBLICATION	X	X	-	-	
ALTER ROUTINE	X	X	-	-	
ALTER SCHEMA	X	X	-	-	
ALTER SEQUENCE	X	X	-	-	
ALTER SERVER	X	X	-	-	
ALTER STATISTICS	X	X	-	-	
ALTER SUBSCRIPTION	X	X	-	-	
ALTER TABLE	X	X	X	X	
ALTER TEXT SEARCH CONFIGURATION	X	X	-	-	
ALTER TEXT SEARCH DICTIONARY	X	X	-	-	
ALTER TEXT SEARCH PARSER	X	X	-	-	

# # # # # #	ddl_command_start	ddl_command_end	sql_drop	table_rewrite	# #
ALTER TEXT SEARCH TEMPLATE	X	X	-	-	
ALTER TRIGGER	X	X	-	-	
ALTER TYPE	X	X	-	X	
ALTER USER MAPPING	X	X	-	-	
ALTER VIEW	X	X	-	-	
COMMENT	X	X	-	-	# # # # # # # # # # # # # #
CREATE ACCESS METHOD	X	X	-	-	
CREATE AGGREGATE	X	X	-	-	
CREATE CAST	X	X	-	-	
CREATE COLLATION	X	X	-	-	
CREATE CONVERSION	X	X	-	-	
CREATE DOMAIN	X	X	-	-	
CREATE EXTENSION	X	X	-	-	
CREATE FOREIGN DATA WRAPPER	X	X	-	-	
CREATE FOREIGN TABLE	X	X	-	-	
CREATE FUNCTION	X	X	-	-	
CREATE INDEX	X	X	-	-	
CREATE LANGUAGE	X	X	-	-	
CREATE MATERIALIZED VIEW	X	X	-	-	
CREATE OPERATOR	X	X	-	-	
CREATE OPERATOR CLASS	X	X	-	-	
CREATE OPERATOR FAMILY	X	X	-	-	
CREATE POLICY	X	X	-	-	
CREATE PROCEDURE	X	X	-	-	
CREATE PUBLICATION	X	X	-	-	
CREATE RULE	X	X	-	-	
CREATE SCHEMA	X	X	-	-	
CREATE SEQUENCE	X	X	-	-	
CREATE SERVER	X	X	-	-	
CREATE STATISTICS	X	X	-	-	
CREATE SUBSCRIPTION	X	X	-	-	
CREATE TABLE	X	X	-	-	
CREATE TABLE AS	X	X	-	-	

# # # # # #	ddl_command_start	ddl_command_end	sql_drop	table_rewrite	##
CREATE TEXT SEARCH CONFIGURATION	X	X	-	-	
CREATE TEXT SEARCH DICTIONARY	X	X	-	-	
CREATE TEXT SEARCH PARSER	X	X	-	-	
CREATE TEXT SEARCH TEMPLATE	X	X	-	-	
CREATE TRIGGER	X	X	-	-	
CREATE TYPE	X	X	-	-	
CREATE USER MAPPING	X	X	-	-	
CREATE VIEW	X	X	-	-	
DROP ACCESS METHOD	X	X	X	-	
DROP AGGREGATE	X	X	X	-	
DROP CAST	X	X	X	-	
DROP COLLATION	X	X	X	-	
DROP CONVERSION	X	X	X	-	
DROP DOMAIN	X	X	X	-	
DROP EXTENSION	X	X	X	-	
DROP FOREIGN DATA WRAPPER	X	X	X	-	
DROP FOREIGN TABLE	X	X	X	-	
DROP FUNCTION	X	X	X	-	
DROP INDEX	X	X	X	-	
DROP LANGUAGE	X	X	X	-	
DROP MATERIALIZED VIEW	X	X	X	-	
DROP OPERATOR	X	X	X	-	
DROP OPERATOR CLASS	X	X	X	-	
DROP OPERATOR FAMILY	X	X	X	-	
DROP OWNED	X	X	X	-	
DROP POLICY	X	X	X	-	
DROP PROCEDURE	X	X	X	-	
DROP PUBLICATION	X	X	X	-	
DROP ROUTINE	X	X	X	-	
DROP RULE	X	X	X	-	
DROP SCHEMA	X	X	X	-	
DROP SEQUENCE	X	X	X	-	
DROP SERVER	X	X	X	-	

######	ddl_command_start	ddl_command_end	sql_drop	table_rewrite	##
DROP STATISTICS	X	X	X	-	
DROP SUBSCRIPTION	X	X	X	-	
DROP TABLE	X	X	X	-	
DROP TEXT SEARCH CONFIGURATION	X	X	X	-	
DROP TEXT SEARCH DICTIONARY	X	X	X	-	
DROP TEXT SEARCH PARSER	X	X	X	-	
DROP TEXT SEARCH TEMPLATE	X	X	X	-	
DROP TRIGGER	X	X	X	-	
DROP TYPE	X	X	X	-	
DROP USER MAPPING	X	X	X	-	
DROP VIEW	X	X	X	-	
GRANT	X	X	-	-	########## ######
IMPORT FOREIGN SCHEMA	X	X	-	-	
REFRESH MATERIALIZED VIEW	X	X	-	-	
REINDEX	X	X	-	-	
REVOKE	X	X	-	-	########## ######
SECURITY LABEL	X	X	-	-	########## ######
SELECT INTO	X	X	-	-	

38.3. C##################

################################## #####C##################### ############## ################## ########C############################## ################## ##################

#######version 1##########################

##EventTriggerData#######context###### ###### C###

```
CALLED_AS_EVENT_TRIGGER(fcinfo)
```

```
############
```

```
((fcinfo)->context != NULL && IsA((fcinfo)->context, EventTriggerData))
```

```
###########fcinfo->context#EventTriggerData                *###########
#EventTriggerData############### ######TriggerData######################
#####
```

struct EventTriggerData#commands/event_trigger.h###########

```
typedef struct EventTriggerData
{
    NodeTag     type;

    const char *event;      /* ##### */
    Node       *parsetree;  /* ##### */
    CommandTag  tag;        /* ###### */
} EventTriggerData;
```

##################

type

##T_EventTriggerData###

event

#####[80.830671]###############
##"ddl_command_start"#"ddl_command_end"#"sql_drop"#"table_rewrite"#
########### ############38.1######### ####################"####"#"DDL_##
##_####"#"DDL_####_end"#"SQL_drop"#"####_rewrite"########### ######
##########38.1#########

parsetree

################### ###PostgreSQL################# #########################

tag

####################################"CREATE FUNCTION"###

##########NULL#####SQL#NULL#############*isNull*######################

38.4.

C###

noddl#################### ###############ddl_command_start########### ###
##(38.1##################)####DDL#############

##################

```
#include "postgres.h"

#include "commands/event_trigger.h"
#include "fmgr.h"

PG_MODULE_MAGIC;

PG_FUNCTION_INFO_V1(noddl);

Datum
noddl(PG_FUNCTION_ARGS)
{
    EventTriggerData *trigdata;


    if (!CALLED_AS_EVENT_TRIGGER(fcinfo))  /* ##### */
        elog(ERROR, "not fired by event trigger manager");
```

```
    trigdata = (EventTriggerData *) fcinfo->context;

    ereport(ERROR,
            (errcode(ERRCODE_INSUFFICIENT_PRIVILEGE),
             errmsg("command \"%s\" denied",
                    GetCommandTagName(trigdata->tag))));

    PG_RETURN_NULL();
}
```

#############36.10.5################################

```
CREATE FUNCTION noddl() RETURNS event_trigger
    AS 'noddl' LANGUAGE C;

CREATE EVENT TRIGGER noddl ON ddl_command_start
    EXECUTE FUNCTION noddl();
```

#######################

```
=# \dy
                      List of event triggers
 Name  |       Event       | Owner | Enabled | Function | Tags
-------+-------------------+-------+---------+----------+------
 noddl | ddl_command_start | dim   | enabled | noddl    |
(1 row)

=# CREATE TABLE foo(id serial);
ERROR:  command "CREATE TABLE" denied
```

#######DDL## ###############
###########################

```
BEGIN;
ALTER EVENT TRIGGER noddl DISABLE;
CREATE TABLE foo (id serial);
ALTER EVENT TRIGGER noddl ENABLE;
COMMIT;
```

(##############DDL################################)

38.5.

table_rewrite###

#####################

```
CREATE OR REPLACE FUNCTION no_rewrite()
 RETURNS event_trigger
 LANGUAGE plpgsql AS
$$
---

--- ###################:
---   public.foo###############
---   #########100###########
---   ##1#####6##################
---
DECLARE
```

```
  table_oid oid := pg_event_trigger_table_rewrite_oid();
  current_hour integer := extract('hour' from current_time);
  pages integer;
  max_pages integer := 100;
BEGIN
  IF pg_event_trigger_table_rewrite_oid() = 'public.foo'::regclass
  THEN
        RAISE EXCEPTION 'you''re not allowed to rewrite the table %',
                        table_oid::regclass;
  END IF;

  SELECT INTO pages relpages FROM pg_class WHERE oid = table_oid;
  IF pages > max_pages
  THEN
        RAISE EXCEPTION 'rewrites only allowed for table with less than %
 pages',
                        max_pages;
  END IF;

  IF current_hour NOT BETWEEN 1 AND 6
  THEN
        RAISE EXCEPTION 'rewrites only allowed between 1am and 6am';
  END IF;
END;
$$;

CREATE EVENT TRIGGER no_rewrite_allowed
                  ON table_rewrite
   EXECUTE FUNCTION no_rewrite();
```

38.6. A Database Login Event Trigger Example

```
##############################################################################
######### ################################################################ ####
################ ########make

############################


-- create test tables and roles
CREATE TABLE user_login_log (
  "user" text,
  "session_start" timestamp with time zone
);
CREATE ROLE day_worker;
CREATE ROLE night_worker;

-- the example trigger function
CREATE OR REPLACE FUNCTION init_session()
  RETURNS event_trigger SECURITY DEFINER
  LANGUAGE plpgsql AS
$$
DECLARE
  hour integer = EXTRACT('hour' FROM current_time at time zone 'utc');
  rec boolean;
BEGIN
-- 1. Forbid logging in between 2AM and 4AM.
IF hour BETWEEN 2 AND 4 THEN
  RAISE EXCEPTION 'Login forbidden';
END IF;
```

```
-- The checks below cannot be performed on standby servers so
-- ensure the database is not in recovery before we perform any
-- operations.
SELECT pg_is_in_recovery() INTO rec;
IF rec THEN
  RETURN;
END IF;

-- 2. Assign some roles. At daytime, grant the day_worker role, else the
-- night_worker role.
IF hour BETWEEN 8 AND 20 THEN
  EXECUTE 'REVOKE night_worker FROM ' || quote_ident(session_user);
  EXECUTE 'GRANT day_worker TO ' || quote_ident(session_user);
ELSE
  EXECUTE 'REVOKE day_worker FROM ' || quote_ident(session_user);
  EXECUTE 'GRANT night_worker TO ' || quote_ident(session_user);
END IF;

-- 3. Initialize user session data
CREATE TEMP TABLE session_storage (x float, y integer);
ALTER TABLE session_storage OWNER TO session_user;

-- 4. Log the connection time
INSERT INTO public.user_login_log VALUES (session_user,
 current_timestamp);

END;
$$;

-- trigger definition
CREATE EVENT TRIGGER init_session
  ON login
  EXECUTE FUNCTION init_session();
ALTER EVENT TRIGGER init_session ENABLE ALWAYS;
```

#39#

####PostgreSQL################## ##
#########

PostgreSQL#######
############

###################
##################################
###################### ##################[ston90b]###[ong90]##########

39.1.

###

############################## ##
################################0################### ##########################
####################SQL##############

################### ####SQL####################SQL#########
debug_print_parse#debug_print_rewritten#####debug_print_plan#########
################################ #########pg_rewrite##########################
####################################

################################ ########################SQL###################
############

############SQL## #############
########

######

#########SELECT#INSERT#UPDATE#DELETE######################

######

############################## SELECT######FROM########################

##############
##############################SQL################################ #########
######################### ##################

########

##

SELECT#################### #SELECT INTO#########INSERT ... SELECT###
#CREATE TABLE#########################

INSERT#UPDATE#DELETE#######################################

#####

########################## SELECT########################## ###
#SELECT#FROM################ #*################### ######################
#####################

DELETE################################ ######################CTID#########
######################### #CTID########################### #########39.2.4##
####################################

INSERT################################ ###VALUES##INSERT ... SELECT# ##SELECT###### ## ## ##NULL########

UPDATE################################ ##############SET column = expression############### ## DELETE##############################CTID#################

##############################1## ####################

##

################################ ####################
#INSERT#UPDATE#DELETE###SELECT#################### ###SQL####WHERE## ######

#####

###########FROM######### SELECT ... FROM a, b, c#################### ##FROM######## ########################## ###JOIN######################### ################# #########JOIN######### ###JOIN###########ON####USING# ################################### ######WHERE########################## ########### ###########SELECT#FROM##WHERE##############

###

ORDER BY################################ ################################ ################################

39.2.

PostgreSQL########################### #########ON SELECT DO INSTEAD######### ####(############)# ###########_RETURN######## ###############

```
CREATE VIEW myview AS SELECT * FROM mytab;
```

```
################
```

```
CREATE TABLE myview (same column list as mytab);
CREATE RULE "_RETURN" AS ON SELECT TO myview DO INSTEAD
    SELECT * FROM mytab;
```

```
#########ON SELECT##################################
```

```
##########DO        INSTEAD################################INSERT#UPDATE###
#DELETE################ ###########39.2.4##########
```

39.2.1. SELECT######

```
########INSERT#UPDATE#DELETE########ON        SELECT######################
## ##################################################################### ####
#SELECT##############
```

```
#######ON SELECT#####1################INSTEAD######SELECT################ #
######################################ON SELECT####################
```

```
##########################2################################     ###2#########1#
##INSERT#UPDATE#DELETE#######################################################
```

```
############### ################################################################
################1################

###2################################


CREATE TABLE shoe_data (
    shoename   text,          -- ###
    sh_avail   integer,       -- ##
    slcolor    text,          -- ########
    slminlen   real,          -- ########
    slmaxlen   real,          -- ########
    slunit     text           -- #####
);


CREATE TABLE shoelace_data (
    sl_name    text,          -- ###
    sl_avail   integer,       -- ##
    sl_color   text,          -- ####
    sl_len     real,          -- #####
    sl_unit    text           -- #####
);


CREATE TABLE unit (
    un_name    text,          -- ###
    un_fact    real           -- cm#########
);

################################

################

CREATE VIEW shoe AS
    SELECT sh.shoename,
           sh.sh_avail,
           sh.slcolor,
           sh.slminlen,
           sh.slminlen * un.un_fact AS slminlen_cm,
           sh.slmaxlen,
           sh.slmaxlen * un.un_fact AS slmaxlen_cm,
           sh.slunit
      FROM shoe_data sh, unit un
     WHERE sh.slunit = un.un_name;

CREATE VIEW shoelace AS
    SELECT s.sl_name,
           s.sl_avail,
           s.sl_color,
           s.sl_len,
           s.sl_unit,
           s.sl_len * u.un_fact AS sl_len_cm
      FROM shoelace_data s, unit u
     WHERE s.sl_unit = u.un_name;

CREATE VIEW shoe_ready AS
    SELECT rsh.shoename,
           rsh.sh_avail,
           rsl.sl_name,
           rsl.sl_avail,
           least(rsh.sh_avail, rsl.sl_avail) AS total_avail
      FROM shoe rsh, shoelace rsl
```

```
     WHERE rsl.sl_color = rsh.slcolor
       AND rsl.sl_len_cm >= rsh.slminlen_cm
       AND rsl.sl_len_cm <= rsh.slmaxlen_cm;
```

shoelace#################CREATE VIEW#####shoelace################### #shoelace###pg_rewrite##### ######## #SELECT###################SELECT###################INSTEAD### ############ ################ ###################### ######################SELECT###### ##1############

##
pg_rewrite###NEW#OLD####2#############SELECT############

#####unit#shoe_data#shoelace_data#######################

```
INSERT INTO unit VALUES ('cm', 1.0);
INSERT INTO unit VALUES ('m', 100.0);
INSERT INTO unit VALUES ('inch', 2.54);

INSERT INTO shoe_data VALUES ('sh1', 2, 'black', 70.0, 90.0, 'cm');
INSERT INTO shoe_data VALUES ('sh2', 0, 'black', 30.0, 40.0, 'inch');
INSERT INTO shoe_data VALUES ('sh3', 4, 'brown', 50.0, 65.0, 'cm');
INSERT INTO shoe_data VALUES ('sh4', 3, 'brown', 40.0, 50.0, 'inch');

INSERT INTO shoelace_data VALUES ('sl1', 5, 'black', 80.0, 'cm');
INSERT INTO shoelace_data VALUES ('sl2', 6, 'black', 100.0, 'cm');
INSERT INTO shoelace_data VALUES ('sl3', 0, 'black', 35.0 , 'inch');
INSERT INTO shoelace_data VALUES ('sl4', 8, 'black', 40.0 , 'inch');
INSERT INTO shoelace_data VALUES ('sl5', 4, 'brown', 1.0 , 'm');
INSERT INTO shoelace_data VALUES ('sl6', 0, 'brown', 0.9 , 'm');
INSERT INTO shoelace_data VALUES ('sl7', 7, 'brown', 60 , 'cm');
INSERT INTO shoelace_data VALUES ('sl8', 1, 'brown', 40 , 'inch');

SELECT * FROM shoelace;

 sl_name   | sl_avail | sl_color | sl_len | sl_unit | sl_len_cm
-----------+----------+----------+--------+---------+-----------
 sl1       |        5 | black    |     80 | cm      |        80
 sl2       |        6 | black    |    100 | cm      |       100
 sl7       |        7 | brown    |     60 | cm      |        60
 sl3       |        0 | black    |     35 | inch    |      88.9
 sl4       |        8 | black    |     40 | inch    |     101.6
 sl8       |        1 | brown    |     40 | inch    |     101.6
 sl5       |        4 | brown    |      1 | m       |       100
 sl6       |        0 | brown    |    0.9 | m       |        90
(8 rows)
```

###############SELECT######################## SELECT * FROM shoelace#### ##########################

```
SELECT shoelace.sl_name, shoelace.sl_avail,
       shoelace.sl_color, shoelace.sl_len,
       shoelace.sl_unit, shoelace.sl_len_cm
  FROM shoelace shoelace;
```

################### ## shoelace## ######################################_RETURN##########

```
SELECT s.sl_name, s.sl_avail,
       s.sl_color, s.sl_len, s.sl_unit,
       s.sl_len * u.un_fact AS sl_len_cm
  FROM shoelace old, shoelace new,
       shoelace_data s, unit u
 WHERE s.sl_unit = u.un_name;
```

```
##############################################################################
##### #######################################
```

```
SELECT shoelace.sl_name, shoelace.sl_avail,
       shoelace.sl_color, shoelace.sl_len,
       shoelace.sl_unit, shoelace.sl_len_cm
  FROM (SELECT s.sl_name,
               s.sl_avail,
               s.sl_color,
               s.sl_len,
               s.sl_unit,
               s.sl_len * u.un_fact AS sl_len_cm
          FROM shoelace_data s, unit u
         WHERE s.sl_unit = u.un_name) shoelace;
```

```
###1########## ##############2#######shoelace old#shoelace new########## ##
################################################ ############################
######################### ####################################################
#########################
```

```
################# ##################################################### #####
############################################## ####old#new######## ##########
############ #####shoelace_data#unit############### ########################
############
```

```
##############################################2##########################
```

```
SELECT * FROM shoe_ready WHERE total_avail >= 2;

 shoename | sh_avail | sl_name | sl_avail | total_avail
----------+----------+---------+----------+-------------
 sh1      |        2 | sl1     |        5 |           2
 sh3      |        4 | sl7     |        7 |           4
(2 rows)
```

```
########################
```

```
SELECT shoe_ready.shoename, shoe_ready.sh_avail,
       shoe_ready.sl_name, shoe_ready.sl_avail,
       shoe_ready.total_avail
  FROM shoe_ready shoe_ready
 WHERE shoe_ready.total_avail >= 2;
```

```
############shoe_ready##################################
```

```
SELECT shoe_ready.shoename, shoe_ready.sh_avail,
       shoe_ready.sl_name, shoe_ready.sl_avail,
       shoe_ready.total_avail
  FROM (SELECT rsh.shoename,
               rsh.sh_avail,
               rsl.sl_name,
               rsl.sl_avail,
               least(rsh.sh_avail, rsl.sl_avail) AS total_avail
```

```
          FROM shoe rsh, shoelace rsl
         WHERE rsl.sl_color = rsh.slcolor
           AND rsl.sl_len_cm >= rsh.slminlen_cm
           AND rsl.sl_len_cm <= rsh.slmaxlen_cm) shoe_ready
 WHERE shoe_ready.total_avail >= 2;
```

######shoe#shoelace##########################3####################

```
SELECT shoe_ready.shoename, shoe_ready.sh_avail,
       shoe_ready.sl_name, shoe_ready.sl_avail,
       shoe_ready.total_avail
  FROM (SELECT rsh.shoename,
               rsh.sh_avail,
               rsl.sl_name,
               rsl.sl_avail,
               least(rsh.sh_avail, rsl.sl_avail) AS total_avail
          FROM (SELECT sh.shoename,
                       sh.sh_avail,
                       sh.slcolor,
                       sh.slminlen,
                       sh.slminlen * un.un_fact AS slminlen_cm,
                       sh.slmaxlen,
                       sh.slmaxlen * un.un_fact AS slmaxlen_cm,
                       sh.slunit
                  FROM shoe_data sh, unit un
                 WHERE sh.slunit = un.un_name) rsh,
               (SELECT s.sl_name,
                       s.sl_avail,
                       s.sl_color,
                       s.sl_len,
                       s.sl_unit,
                       s.sl_len * u.un_fact AS sl_len_cm
                  FROM shoelace_data s, unit u
                 WHERE s.sl_unit = u.un_name) rsl
         WHERE rsl.sl_color = rsh.slcolor
           AND rsl.sl_len_cm >= rsh.slminlen_cm
           AND rsl.sl_len_cm <= rsh.slmaxlen_cm) shoe_ready
 WHERE shoe_ready.total_avail > 2;
```

###
######### ##

39.2.2. #SELECT########

#########################2################ ######################## ###########
###
##

SELECT################################## #########################SELECT#####
###################################### ############## #####a#b########t1#t2#
######2######################

```
SELECT t2.b FROM t1, t2 WHERE t1.a = t2.a;

UPDATE t1 SET b = t2.b FROM t2 WHERE t1.a = t2.a;
```

############

- ############t1#t2###########
- ###########t2############b######1##########

- ################a###########
- ######t1#t2##############

################################ #######2########### UPDATE##t1############## ####################################

```
UPDATE t1 SET a = t1.a, b = t2.b FROM t2 WHERE t1.a = t2.a;
```

###################

```
SELECT t1.a, t2.b FROM t1, t2 WHERE t1.a = t2.a;
```

###################### ######UPDATE############### ########################### ################ ########################### 1##SELECT#######1##UPDATE###### ############################ ########UPDATE####################t1############## ######## ######################################

###########UPDATE##DELETE########################### ##########ID#CTID#### ################################### ##############CTID########t1############## #### CTID############################

```
SELECT t1.a, t2.b, t1.ctid FROM t1, t2 WHERE t1.a = t2.a;
```

###PostgreSQL############### ##################ROLLBACK######## UPDATE### #CTID#########################CTID###############cmax#xmax################## #######ID######## ################################vacuum##################### ###

#############

39.2.3. PostgreSQL###########

#2#####1###########SELECT# ########4###########################unit#######2#########

#

#####[87.603306]#######################INSTEAD OF########################### ################### #CREATE TRIGGER########## INSERT###################### ###################### UPDATE#DELETE#################################### ################### ## ########## ##CREATE TRIGGER## ### ####################### INSERT## # ## ########## #####################-##

#####[90.697674]#################################### ######################### #############CTID######################### ##########################CTID### ################################## ######UPDATE#DELETE###################### ################################## ####################INSTEAD OF######### ### ############################

#####[65.579710]################INSERT#UPDATE#DELETE############# #INSTEAD########### ########################1########################### ## #39.4######## #######1##########INSERT#######################INSTEAD#### ############ ########################1######################## ###39.4######## ################# #####################################

#######INSTEAD OF### #INSERT#UPDATE#DELETE###

#####[82.828283]##########INSERT#UPDATE####DELETE########################## ## #############INSERT##### ########################### ## #####

####INSTEAD####INSTEAD OF## ##

39.3.

PostgreSQL###

```
CREATE MATERIALIZED VIEW mymatview AS SELECT * FROM mytab;
```

#

```
CREATE TABLE mymatview AS SELECT * FROM mytab;
```

###
##

```
REFRESH MATERIALIZED VIEW mymatview;
```

#################PostgreSQL#################################### ############## ############################## ## ##

########################

```
CREATE TABLE invoice (
    invoice_no    integer        PRIMARY KEY,
```

```
    seller_no     integer,       -- ####ID
    invoice_date  date,          -- ###
    invoice_amt   numeric(13,2)  -- ###
);
```

```
##############################################################################
##


CREATE MATERIALIZED VIEW sales_summary AS
  SELECT
      seller_no,
      invoice_date,
      sum(invoice_amt)::numeric(13,2) as sales_amt
    FROM invoice
    WHERE invoice_date < CURRENT_DATE
    GROUP BY
      seller_no,
      invoice_date;

CREATE UNIQUE INDEX sales_summary_seller
  ON sales_summary (seller_no, invoice_date);
```

```
################################################### ###SQL####################
###############


REFRESH MATERIALIZED VIEW sales_summary;
```

```
################################################################           ####
#file_fdw#######################################################################
################## ################################file_fdw###################
######## ##################################
```

```
######:


CREATE EXTENSION file_fdw;
CREATE SERVER local_file FOREIGN DATA WRAPPER file_fdw;
CREATE FOREIGN TABLE words (word text NOT NULL)
  SERVER local_file
  OPTIONS (filename '/usr/share/dict/words');
CREATE MATERIALIZED VIEW wrd AS SELECT * FROM words;
CREATE UNIQUE INDEX wrd_word ON wrd (word);
CREATE EXTENSION pg_trgm;
CREATE INDEX wrd_trgm ON wrd USING gist (word gist_trgm_ops);
VACUUM ANALYZE wrd;
```

```
file_fdw#########################


SELECT count(*) FROM words WHERE word = 'caterpiler';

 count
-------
     0
(1 row)
```

```
EXPLAIN ANALYZE###########:


 Aggregate  (cost=21763.99..21764.00 rows=1 width=0) (actual
 time=188.180..188.181 rows=1 loops=1)
   ->  Foreign Scan on words  (cost=0.00..21761.41 rows=1032 width=0)
 (actual time=188.177..188.177 rows=0 loops=1)
```

```
         Filter: (word = 'caterpiler'::text)
         Rows Removed by Filter: 479829
         Foreign File: /usr/share/dict/words
         Foreign File Size: 4953699
 Planning time: 0.118 ms
 Execution time: 188.273 ms
```

```
####################################
```

```
 Aggregate  (cost=4.44..4.45 rows=1 width=0) (actual time=0.042..0.042
 rows=1 loops=1)
   ->  Index Only Scan using wrd_word on wrd  (cost=0.42..4.44 rows=1
 width=0) (actual time=0.039..0.039 rows=0 loops=1)
         Index Cond: (word = 'caterpiler'::text)
         Heap Fetches: 0
 Planning time: 0.164 ms
 Execution time: 0.117 ms
```

```
#########word########################################file_fdw#pg_trgm########
###############caterpillar##
```

```
SELECT word FROM words ORDER BY word <-> 'caterpiler' LIMIT 10;

     word
---------------
 cater
 caterpillar
 Caterpillar
 caterpillars
 caterpillar's
 Caterpillar's
 caterer
 caterer's
 caters
 catered
(10 rows)
```

```
 Limit  (cost=11583.61..11583.64 rows=10 width=32) (actual
 time=1431.591..1431.594 rows=10 loops=1)
   ->  Sort  (cost=11583.61..11804.76 rows=88459 width=32) (actual
 time=1431.589..1431.591 rows=10 loops=1)
         Sort Key: ((word <-> 'caterpiler'::text))
         Sort Method: top-N heapsort  Memory: 25kB
         ->  Foreign Scan on words  (cost=0.00..9672.05 rows=88459
 width=32) (actual time=0.057..1286.455 rows=479829 loops=1)
               Foreign File: /usr/share/dict/words
               Foreign File Size: 4953699
 Planning time: 0.128 ms
 Execution time: 1431.679 ms
```

```
##################:
```

```
 Limit  (cost=0.29..1.06 rows=10 width=10) (actual time=187.222..188.257
 rows=10 loops=1)
   ->  Index Scan using wrd_trgm on wrd  (cost=0.29..37020.87 rows=479829
 width=10) (actual time=187.219..188.252 rows=10 loops=1)
         Order By: (word <-> 'caterpiler'::text)
 Planning time: 0.196 ms
 Execution time: 198.640 ms
```

```
##############################################################
```

39.4. INSERT#UPDATE#DELETE########

INSERT#UPDATE#DELETE################################ ##########CREATE RULE##################

- ################
- #############
- INSTEAD####ALSO#################
- #######NEW#OLD###########
- #################

######################### #########0##################################

> ##
>
> ######INSERT/UPDATE/DELETE##################################### ###### ############################## #######volatile############################ ###volatile##
>
> ############################### ##WITH##############UPDATE######SET### ############SELECT###### ## #####################

39.4.1.

```
CREATE [ OR REPLACE ] RULE name AS ON event
    TO table [ WHERE condition ]
    DO [ ALSO | INSTEAD ] { NOTHING | command | ( command ; command
 ... ) }
```

################ ##########INSERT#UPDATE#DELETE###############

##############################CREATE RULE################################ ####### ################################## ################## 0#NOTHING#### ###1################## ###########1################### ######################### ######INSTEAD#ALSO#####################

################# ## ############## ########################NEW#########OLD#################################

1############################3#########

ALSO###INSTEAD########

    ####################################

######ALSO

    ##

######INSTEAD

    ##

##########ALSO################################## #####INSTEAD##############
###############1################1#####2##################

ON INSERT###############INSTEAD#################################### ####
############################## ####ON UPDATE#ON DELETE#################### ################### ### ############
#####################################

###
##############
############### (##########################)

pg_rewrite##################################### ####NEW#OLD################
NEW################################
######### ########################### #########NEW#OLD########UPDATE####
##NULL############INSERT##### OLD##

##############################
####################

39.4.1.1.

shoelace_data######sl_avail################ ###########shoelace_data##
######UPDATE##################

```
CREATE TABLE shoelace_log (
    sl_name    text,          -- #######
    sl_avail   integer,       -- ######
    log_who    text,          -- ######
    log_when   timestamp      -- ######
);

CREATE RULE log_shoelace AS ON UPDATE TO shoelace_data
    WHERE NEW.sl_avail <> OLD.sl_avail
    DO INSERT INTO shoelace_log VALUES (
                                        NEW.sl_name,
                                        NEW.sl_avail,
                                        current_user,
                                        current_timestamp
                                    );
```

###############

```
UPDATE shoelace_data SET sl_avail = 6 WHERE sl_name = 'sl7';
```

###############

```
SELECT * FROM shoelace_log;

 sl_name | sl_avail | log_who | log_when
---------+----------+---------+----------------------------------
 sl7     |        6 | Al      | Tue Oct 20 16:14:45 1998 MET DST
(1 row)
```

############## ################### ########################

```
UPDATE shoelace_data SET sl_avail = 6
  FROM shoelace_data shoelace_data
 WHERE shoelace_data.sl_name = 'sl7';
```

```
###log_shoelace###ON UPDATE########


NEW.sl_avail <> OLD.sl_avail

###############


INSERT INTO shoelace_log VALUES (
       new.sl_name, new.sl_avail,
       current_user, current_timestamp )
  FROM shoelace_data new, shoelace_data old;

####INSERT  ...  VALUES  ...  FROM############################ ###FROM###
#new#old################################ #####INSERT######################
###########

###########ALSO##########################################2##################
### #1########################################### #############


INSERT INTO shoelace_log VALUES (
       new.sl_name, new.sl_avail,
       current_user, current_timestamp )
  FROM shoelace_data new, shoelace_data old,
       shoelace_data shoelace_data;

#2######################### #############sl_avail#############


INSERT INTO shoelace_log VALUES (
       new.sl_name, new.sl_avail,
       current_user, current_timestamp )
  FROM shoelace_data new, shoelace_data old,
       shoelace_data shoelace_data
 WHERE new.sl_avail <> old.sl_avail;

#INSERT ... VALUES#WHERE################### ########################### ####
####INSERT ... SELECT########################

#3#############################################################


INSERT INTO shoelace_log VALUES (
       new.sl_name, new.sl_avail,
       current_user, current_timestamp )
  FROM shoelace_data new, shoelace_data old,
       shoelace_data shoelace_data
 WHERE new.sl_avail <> old.sl_avail
   AND shoelace_data.sl_name = 'sl7';

#4#####################################################NEW##########


INSERT INTO shoelace_log VALUES (
       shoelace_data.sl_name, 6,
       current_user, current_timestamp )
  FROM shoelace_data new, shoelace_data old,
       shoelace_data shoelace_data
 WHERE 6 <> old.sl_avail
   AND shoelace_data.sl_name = 'sl7';

#5##########OLD####################


INSERT INTO shoelace_log VALUES (
```

```
       shoelace_data.sl_name, 6,
       current_user, current_timestamp )
  FROM shoelace_data new, shoelace_data old,
       shoelace_data shoelace_data
 WHERE 6 <> shoelace_data.sl_avail
   AND shoelace_data.sl_name = 'sl7';
```

###############ALSO#################### ##########################2##########
#######

```
INSERT INTO shoelace_log VALUES (
       shoelace_data.sl_name, 6,
       current_user, current_timestamp )
  FROM shoelace_data
 WHERE 6 <> shoelace_data.sl_avail
   AND shoelace_data.sl_name = 'sl7';

UPDATE shoelace_data SET sl_avail = 6
 WHERE sl_name = 'sl7';
```

##2##########################

###

```
UPDATE shoelace_data SET sl_color = 'green'
 WHERE sl_name = 'sl7';
```

######################sl_avail#########
##NEW.sl_avail#shoelace_data.sl_avail######### ########################
##########

```
INSERT INTO shoelace_log VALUES (
       shoelace_data.sl_name, shoelace_data.sl_avail,
       current_user, current_timestamp )
  FROM shoelace_data
 WHERE shoelace_data.sl_avail <> shoelace_data.sl_avail
   AND shoelace_data.sl_name = 'sl7';
```

###################

########################### ###########################

```
UPDATE shoelace_data SET sl_avail = 0
 WHERE sl_color = 'black';
```

#########4########sl1#sl2#sl3###sl4## ###sl3###sl_avail = 0####### #######
###

```
INSERT INTO shoelace_log
SELECT shoelace_data.sl_name, 0,
       current_user, current_timestamp
  FROM shoelace_data
 WHERE 0 <> shoelace_data.sl_avail
   AND shoelace_data.sl_color = 'black';
```

############3################ ############## [###sl3##WHERE 0 != shoelace_data.sl_avail########0!=0###########4#-1#3##############]#

#################################### ##UPDATE#################0#######0 <> shoelace_data.sl_avail############INSERT###############

39.4.2.

##########INSERT#UPDATE#DELETE##
################ ####################

```
CREATE RULE shoe_ins_protect AS ON INSERT TO shoe
    DO INSTEAD NOTHING;
CREATE RULE shoe_upd_protect AS ON UPDATE TO shoe
    DO INSTEAD NOTHING;
CREATE RULE shoe_del_protect AS ON DELETE TO shoe
    DO INSTEAD NOTHING;
```

###shoe### #################INSTEAD#####
#################### ##
######

shoelace#######
##################

```
CREATE RULE shoelace_ins AS ON INSERT TO shoelace
    DO INSTEAD
    INSERT INTO shoelace_data VALUES (
           NEW.sl_name,
           NEW.sl_avail,
           NEW.sl_color,
           NEW.sl_len,
           NEW.sl_unit
    );

CREATE RULE shoelace_upd AS ON UPDATE TO shoelace
    DO INSTEAD
    UPDATE shoelace_data
       SET sl_name = NEW.sl_name,
           sl_avail = NEW.sl_avail,
           sl_color = NEW.sl_color,
           sl_len = NEW.sl_len,
           sl_unit = NEW.sl_unit
     WHERE sl_name = OLD.sl_name;

CREATE RULE shoelace_del AS ON DELETE TO shoelace
    DO INSTEAD
    DELETE FROM shoelace_data
     WHERE sl_name = OLD.sl_name;
```

#######RETURNING#########################RETURNING#################### ####
#########################shoelace######################## ##################

```
CREATE RULE shoelace_ins AS ON INSERT TO shoelace
    DO INSTEAD
    INSERT INTO shoelace_data VALUES (
           NEW.sl_name,
           NEW.sl_avail,
           NEW.sl_color,
           NEW.sl_len,
           NEW.sl_unit
    )
    RETURNING
           shoelace_data.*,
           (SELECT shoelace_data.sl_len * u.un_fact
            FROM unit u WHERE shoelace_data.sl_unit = u.un_name);
```

##1##############INSERT######INSERT RETURNING##########################
INSERT##RETURNING###########

#################################### ###########shoelace################ ###
################################2################ #################

```
CREATE TABLE shoelace_arrive (
    arr_name    text,
    arr_quant   integer
);

CREATE TABLE shoelace_ok (
    ok_name     text,
    ok_quant    integer
);

CREATE RULE shoelace_ok_ins AS ON INSERT TO shoelace_ok
    DO INSTEAD
    UPDATE shoelace
       SET sl_avail = sl_avail + NEW.ok_quant
     WHERE sl_name = NEW.ok_name;
```

###########shoelace_arrive################

```
SELECT * FROM shoelace_arrive;

 arr_name | arr_quant
----------+-----------
 sl3      |        10
 sl6      |        20
 sl8      |        20
(3 rows)
```

##################

```
SELECT * FROM shoelace;

 sl_name  | sl_avail | sl_color | sl_len | sl_unit | sl_len_cm
----------+----------+----------+--------+---------+-----------
 sl1      |        5 | black    |     80 | cm      |        80
 sl2      |        6 | black    |    100 | cm      |       100
 sl7      |        6 | brown    |     60 | cm      |        60
 sl3      |        0 | black    |     35 | inch    |      88.9
 sl4      |        8 | black    |     40 | inch    |     101.6
 sl8      |        1 | brown    |     40 | inch    |     101.6
 sl5      |        4 | brown    |      1 | m       |       100
 sl6      |        0 | brown    |    0.9 | m       |        90
(8 rows)
```

###############

```
INSERT INTO shoelace_ok SELECT * FROM shoelace_arrive;
```

############

```
SELECT * FROM shoelace ORDER BY sl_name;

 sl_name  | sl_avail | sl_color | sl_len | sl_unit | sl_len_cm
----------+----------+----------+--------+---------+-----------
 sl1      |        5 | black    |     80 | cm      |        80
```

```
 sl2      |        6 | black    |    100 | cm      |       100
 sl7      |        6 | brown    |     60 | cm      |        60
 sl4      |        8 | black    |     40 | inch    |     101.6
 sl3      |       10 | black    |     35 | inch    |      88.9
 sl8      |       21 | brown    |     40 | inch    |     101.6
 sl5      |        4 | brown    |      1 | m       |       100
 sl6      |       20 | brown    |    0.9 | m       |        90
(8 rows)

SELECT * FROM shoelace_log;

 sl_name | sl_avail | log_who| log_when
---------+----------+--------+----------------------------------
 sl7     |        6 | Al     | Tue Oct 20 19:14:45 1998 MET DST
 sl3     |       10 | Al     | Tue Oct 20 19:25:16 1998 MET DST
 sl6     |       20 | Al     | Tue Oct 20 19:25:16 1998 MET DST
 sl8     |       21 | Al     | Tue Oct 20 19:25:16 1998 MET DST
(4 rows)
```

1##INSERT ... SELECT###################### ############################### ### #################

```
INSERT INTO shoelace_ok
SELECT shoelace_arrive.arr_name, shoelace_arrive.arr_quant
  FROM shoelace_arrive shoelace_arrive, shoelace_ok shoelace_ok;
```

###shoelace_ok_ins######################

```
UPDATE shoelace
   SET sl_avail = shoelace.sl_avail + shoelace_arrive.arr_quant
  FROM shoelace_arrive shoelace_arrive, shoelace_ok shoelace_ok,
       shoelace_ok old, shoelace_ok new,
       shoelace shoelace
 WHERE shoelace.sl_name = shoelace_arrive.arr_name;
```

####shoelace_ok#####INSERT####### ##############################2#######
#shoelace_upd############

```
UPDATE shoelace_data
   SET sl_name = shoelace.sl_name,
       sl_avail = shoelace.sl_avail + shoelace_arrive.arr_quant,
       sl_color = shoelace.sl_color,
       sl_len = shoelace.sl_len,
       sl_unit = shoelace.sl_unit
  FROM shoelace_arrive shoelace_arrive, shoelace_ok shoelace_ok,
       shoelace_ok old, shoelace_ok new,
       shoelace shoelace, shoelace old,
       shoelace new, shoelace_data shoelace_data
 WHERE shoelace.sl_name = shoelace_arrive.arr_name
   AND shoelace_data.sl_name = shoelace.sl_name;
```

#####INSTEAD######################### ########shoelace#################### #### ################################_RETURN###################

```
UPDATE shoelace_data
   SET sl_name = s.sl_name,
       sl_avail = s.sl_avail + shoelace_arrive.arr_quant,
       sl_color = s.sl_color,
       sl_len = s.sl_len,
       sl_unit = s.sl_unit
  FROM shoelace_arrive shoelace_arrive, shoelace_ok shoelace_ok,
```

```
       shoelace_ok old, shoelace_ok new,
       shoelace shoelace, shoelace old,
       shoelace new, shoelace_data shoelace_data,
       shoelace old, shoelace new,
       shoelace_data s, unit u
 WHERE s.sl_name = shoelace_arrive.arr_name
   AND shoelace_data.sl_name = s.sl_name;
```

####log_shoelace################################

```
INSERT INTO shoelace_log
SELECT s.sl_name,
       s.sl_avail + shoelace_arrive.arr_quant,
       current_user,
       current_timestamp
  FROM shoelace_arrive shoelace_arrive, shoelace_ok shoelace_ok,
       shoelace_ok old, shoelace_ok new,
       shoelace shoelace, shoelace old,
       shoelace new, shoelace_data shoelace_data,
       shoelace old, shoelace new,
       shoelace_data s, unit u,
       shoelace_data old, shoelace_data new
       shoelace_log shoelace_log
 WHERE s.sl_name = shoelace_arrive.arr_name
   AND shoelace_data.sl_name = s.sl_name
   AND (s.sl_avail + shoelace_arrive.arr_quant) <> s.sl_avail;
```

#######################################

#######SQL#######2##################

```
INSERT INTO shoelace_log
SELECT s.sl_name,
       s.sl_avail + shoelace_arrive.arr_quant,
       current_user,
       current_timestamp
  FROM shoelace_arrive shoelace_arrive, shoelace_data shoelace_data,
       shoelace_data s
 WHERE s.sl_name = shoelace_arrive.arr_name
   AND shoelace_data.sl_name = s.sl_name
   AND s.sl_avail + shoelace_arrive.arr_quant <> s.sl_avail;

UPDATE shoelace_data
   SET sl_avail = shoelace_data.sl_avail + shoelace_arrive.arr_quant
  FROM shoelace_arrive shoelace_arrive,
       shoelace_data shoelace_data,
       shoelace_data s
 WHERE s.sl_name = shoelace_arrive.sl_name
   AND shoelace_data.sl_name = s.sl_name;
```

####1##############################3##################4######5################
######2##############

################## ######2###########1#########shoelace_data##############
####### #############INSERT############################

```
Nested Loop
  ->  Merge Join
        ->  Seq Scan
              ->  Sort
                    ->  Seq Scan on s
        ->  Seq Scan
```

```
              ->  Sort
                    ->  Seq Scan on shoelace_arrive
  ->  Seq Scan on shoelace_data
```

##

```
Merge Join
  ->  Seq Scan
        ->  Sort
              ->  Seq Scan on s
  ->  Seq Scan
        ->  Sort
              ->  Seq Scan on shoelace_arrive
```

#######################shoelace_data############################## ###UPDATE# #################### ###########################

###PostgreSQL#################### ################################

```
INSERT INTO shoelace VALUES ('sl9', 0, 'pink', 35.0, 'inch', 0.0);
INSERT INTO shoelace VALUES ('sl10', 1000, 'magenta', 40.0, 'inch', 0.0);
```

###########shoelace############################ ##############

```
CREATE VIEW shoelace_mismatch AS
    SELECT * FROM shoelace WHERE NOT EXISTS
        (SELECT shoename FROM shoe WHERE slcolor = sl_color);
```

################

```
SELECT * FROM shoelace_mismatch;

 sl_name | sl_avail | sl_color | sl_len | sl_unit | sl_len_cm
---------+----------+----------+--------+---------+-----------
 sl9     |        0 | pink     |     35 | inch    |      88.9
 sl10    |     1000 | magenta  |     40 | inch    |     101.6
```

###PostgreSQL################## ## #############1###########

```
CREATE VIEW shoelace_can_delete AS
    SELECT * FROM shoelace_mismatch WHERE sl_avail = 0;
```

############

```
DELETE FROM shoelace WHERE EXISTS
    (SELECT * FROM shoelace_can_delete
             WHERE sl_name = shoelace.sl_name);
```

###########

```
SELECT * FROM shoelace;

 sl_name | sl_avail | sl_color | sl_len | sl_unit | sl_len_cm
---------+----------+----------+--------+---------+-----------
 sl1     |        5 | black    |     80 | cm      |        80
 sl2     |        6 | black    |    100 | cm      |       100
 sl7     |        6 | brown    |     60 | cm      |        60
 sl4     |        8 | black    |     40 | inch    |     101.6
```

```
 sl3     |       10 | black    |     35 | inch    |      88.9
 sl8     |       21 | brown    |     40 | inch    |     101.6
 sl10    |     1000 | magenta  |     40 | inch    |     101.6
 sl5     |        4 | brown    |      1 | m       |       100
 sl6     |       20 | brown    |    0.9 | m       |        90
(9 rows)
```

##4#####/##########################1###
#######DELETE##

################################ #######################

39.5.

PostgreSQL###/############# ##
#####################################

################### ##
PostgreSQL################################## #############CREATE VIEW########
######SELECT## #
#####################################/##########################

############# ### ##########
############

```
CREATE TABLE phone_data (person text, phone text, private boolean);
CREATE VIEW phone_number AS
    SELECT person, CASE WHEN NOT private THEN phone END AS phone
    FROM phone_data;
GRANT SELECT ON phone_number TO assistant;
```

########################phone_data############## ####GRANT#####
#phone_number######SELECT##### #######phone_number###SELECT##
#phone_data###SELECT######### ######phone_number###################
##phone_data###
phone_number###

################# ################################ ####################PUBLIC
##################### #################phone_number############# ########
#phone_data################## ################################## ########
#phone_number## #####################
##

##################
#phone_number#################1#1##################### ####################
################# GRANT##################### ############################
#REVOKE#######

####################################security_barrier#########################
######################### ####################

```
CREATE VIEW phone_number AS
    SELECT person, phone FROM phone_data WHERE phone NOT LIKE '412%';
```

#######phone_number###SELECT#phone_data###SELECT######phone#412#######
################################ ########################NOT LIKE###########
############################ ###########

```
CREATE FUNCTION tricky(text, text) RETURNS bool AS $$
BEGIN
    RAISE NOTICE '% => %', $1, $2;
    RETURN true;
END;
$$ LANGUAGE plpgsql COST 0.0000000000000000000001;

SELECT * FROM phone_number WHERE tricky(person, phone);
```

#########NOT LIKE#####tricky################phone_data############# #NOTICE######### ### ################### #####################

################### ########################shoelace######## #SELECT#INSERT#UPDATE#DELETE############# ####shoelace_log####SELECT## ### ## ######################## ######################## #####shoelace_log################INSERT########### ################################### ########################

#############################security_barrier########### ################### ################################### ##############################

```
CREATE VIEW phone_number WITH (security_barrier) AS
    SELECT person, phone FROM phone_data WHERE phone NOT LIKE '412%';
```

security_barrier## ## ##

######################### #######LEAKPROOF################################### ### ## ### ## #LEAKPROOF############### ## leak-proof ######## ## ########

####security_barrier### ######################## ##EXPLAIN##### ## ################################### ### ###### #"######"##

39.6.

PostgreSQL#########################INSERT 149592 1################## ####### ######################### ######################################

#######################

- #########INSTEAD## ###### ###INSTEAD## ################### #######################
- #########INSTEAD################################## ###################### ##INSTEAD##########################INSERT#UPDATE###DELETE############# ########### ##OID### ###0#######

###INSTEAD################################### ######################################

39.7.

################PostgreSQL############### ############1##################
##############################NOTHING##################################
################ ##
##############

################################## ##################INSTEAD OF############
###

##################
########## #####1#################1###
########1######################### ##
#######

################################### ####2##############

```
CREATE TABLE computer (
    hostname        text,    -- ########
    manufacturer    text     -- ########
);
```

```
CREATE TABLE software (
    software        text,    -- ########
    hostname        text     -- ########
);
```

2###################hostname############# ###/#################software####
################## ####################

```
DELETE FROM software WHERE hostname = $1;
```

computer#######1##
#hostname########## ###################

```
CREATE RULE computer_del AS ON DELETE TO computer
    DO DELETE FROM software WHERE hostname = OLD.hostname;
```

##################

```
DELETE FROM computer WHERE hostname = 'mypc.local.net';
```

##########computer######################## ##################################
########## ########################

```
DELETE FROM software WHERE computer.hostname = 'mypc.local.net'
                       AND software.hostname = computer.hostname;
```

#####################################

```
Nestloop
  ->  Index Scan using comp_hostidx on computer
  ->  Index Scan using soft_hostidx on software
```

##############################

```
#######hostname#old####2,000####computer########### #####2##############1#####
#########


DELETE FROM computer WHERE hostname >= 'old'
                       AND hostname <  'ole'

##########################


DELETE FROM software WHERE computer.hostname >= 'old' AND
 computer.hostname < 'ole'
                       AND software.hostname = computer.hostname;

###################


Hash Join
  ->  Seq Scan on software
  ->  Hash
    ->  Index Scan using comp_hostidx on computer

##1#################


DELETE FROM computer WHERE hostname ~ '^old';

#######################################


Nestloop
  ->  Index Scan using comp_hostidx on computer
  ->  Index Scan using soft_hostidx on software

###########AND##################################################computer#
#hostname########software#####################################  ###########
2,000##################1########computer##1############software##2,000####
############## ####################2################## #######################
#####software############## ########################################SPI######
##2,000##################

###############


DELETE FROM computer WHERE manufacturer = 'bim';

#####computer###################  ##########################################
########################


DELETE FROM software WHERE computer.manufacturer = 'bim'
                       AND software.hostname = computer.hostname;

################2########################### computer######################


Nestloop
  ->  Index Scan using comp_manufidx on computer
  ->  Index Scan using soft_hostidx on software

#######################################################

#####################################################################
```

#40#

PostgreSQL###SQL#C########################## #################PL####### ### ## ########### ###################### ####################################PostgreSQL######### ##################### ########C#######C################################## ##

##PostgreSQL########PL/pgSQL##41###PL/Tcl##42###PL/Perl##43###PL/Python##44##### 4############ ########################## ##H############## ##################### ######################56########

40.1.

####template1################# #####template1#####CREATE DATABASE## ###

###CREATE EXTENSION *language_name*#### ####### ##

####################

########5######################### ############################ ###########SQL ##CREATE EXTENSION########

1. ## #############C ############################## 36.10.5######### ############################ ######################### ############################

2. ##########################

   ```
   CREATE FUNCTION handler_function_name()
       RETURNS language_handler
       AS 'path-to-shared-object'
       LANGUAGE C;
   ```

   language_handler######################SQL#########SQL################ #############

3. (Optional) ##################################(DO####)######################### ###### ##

   ```
   CREATE FUNCTION inline_function_name(internal)
       RETURNS void
       AS 'path-to-shared-object'
       LANGUAGE C;
   ```

4. (Optional) ## ##### ###########CREATE FUNCTION######## ##################################### ####

   ```
   CREATE FUNCTION validator_function_name(oid)
       RETURNS void
       AS 'path-to-shared-object'
       LANGUAGE C STRICT;
   ```

5. #####PL#######################

```
CREATE [TRUSTED] LANGUAGE language_name
    HANDLER handler_function_name
    [INLINE inline_function_name]
    [VALIDATOR validator_function_name] ;
```

TRUSTED###
TRUSTED## PL########
################TRUSTED###
PL/pgSQL#PL/Tcl#PL/Perl###TRUSTED######### ##################PL/TclU#PL/PerlU#PL/PythonU#######TRUSTED###########

40.1################PL/Perl##################

#40.1 PL/Perl#############

##################PL/Perl#################################

```
CREATE FUNCTION plperl_call_handler() RETURNS language_handler AS
    '$libdir/plperl' LANGUAGE C;
```

PL/Perl##

```
CREATE FUNCTION plperl_inline_handler(internal) RETURNS void AS
    '$libdir/plperl' LANGUAGE C STRICT;

CREATE FUNCTION plperl_validator(oid) RETURNS void AS
    '$libdir/plperl' LANGUAGE C STRICT;
```

#########

```
CREATE TRUSTED LANGUAGE plperl
    HANDLER plperl_call_handler
    INLINE plperl_inline_handler
    VALIDATOR plperl_validator;
```

################plperl##################################

######PostgreSQL###########PL/pgSQL####################################### ##
##PL/pgSQL######################### Tcl###############PL/Tcl#PL/TclU###########
####Perl##############
PL/Perl#PL/PerlU######Python##############PL/PythonU############################
################

#41# PL/pgSQL — SQL#####

41.1.

PL/pgSQL##PostgreSQL########################## PL/pgSQL#######################
####

- ###########################
- SQL##############
- ############
- ##############################
- #####################
- #######

PL/pgSQL#################################### ################################
##

PostgreSQL 9.0####PL/pgSQL################## ####################################
###############

41.1.1. PL/pgSQL##########

SQL#PostgreSQL## ############## ##
######SQL###############################

#################################### ######################## ##############
########################
##

PL/pgSQL## ##############SQL###
##########################/#######################

- ########################
- ##
- #####################

###

###PL/pgSQL##SQL##########################

41.1.2.

PL/pgSQL###
####### ## ####7.2.1.4###########PL/
pgSQL###`record`##########################`record`#################`record`####
################################

PL/pgSQL###`VARIADIC`################################## ###36.5.6########SQL##
#############

###PL/pgSQL####36.2.5#################################### ##################
######################### ##41.3.1######

PL/pgSQL####1## ################
######RETURN NEXT##########################RETURN QUERY##################

#################PL/pgSQL####void################ ########################
####

PL/pgSQL## ##########################
###################### RETURNS TABLE###RETURNS SETOF#############

######41.3.1###41.6.1######

41.2. PL/pgSQL###

PL/pgSQL########CREATE FUNCTION###################### ######################
####

```
CREATE FUNCTION somefunc(integer, text) RETURNS integer
AS 'function body text'
LANGUAGE plpgsql;
```

#####CREATE FUNCTION################ ##############################(4.1.2.4##
#)################# ### ####
##################################

PL/pgSQL############ ######################### #################

```
[ <<label>> ]
[ DECLARE
    declarations ]
BEGIN
    statements
END [ label ];
```

########################### ########################END####################
######END#######

###
BEGIN######################### ###################

label########EXIT## END######
##################################

######################## ########################SQL#######################

PL/pgSQL##########SQL################ #######(--)######################### /*#
##################*/################ #################

############################## ## ######
######
####

```
CREATE FUNCTION somefunc() RETURNS integer AS $$
<< outerblock >>
DECLARE
```

```
    quantity integer := 30;
BEGIN

    RAISE NOTICE 'Quantity here is %', quantity;  -- Quantity here is 30##
#
    quantity := 50;
    --

    -- ########
    --
    DECLARE
        quantity integer := 80;
    BEGIN

        RAISE NOTICE 'Quantity here is %', quantity;  -- Quantity here is
 80###
        RAISE NOTICE 'Outer quantity here is %', outerblock.quantity;  --
 Quantity here is 50###
    END;


    RAISE NOTICE 'Quantity here is %', quantity;  -- Quantity here is 50##
#

    RETURN quantity;
END;
$$ LANGUAGE plpgsql;
```

##

PL/pgSQL########################### ################################ #FOUND##########41.5.5########## ###################### ################ #################

PL/pgSQL############BEGIN/END###############SQL######################## PL/pgSQL#BEGIN/END################################### #########PL/pgSQL###### ##########41.8######### ###EXCEPTION### ############### #######41.6.8#########

41.3.

########FOR######################### ######################### #######################FOR#########################

PL/pgSQL####integer#varchar#char#######SQL##############

##############

```
user_id integer;
quantity numeric(5);
url varchar;
myrow tablename%ROWTYPE;
myfield tablename.columnname%TYPE;
arow RECORD;
```

###################

name [CONSTANT] *type* [COLLATE *collation_name*] [NOT NULL] [{ DEFAULT | := | = } *expression*];

DEFAULT#################################### DEFAULT##############SQL#NULL# ######## CONSTANT## COLLATE# ##########################41.3.6########## NOT NULL########NULL############# #### NOT NULL##############NULL#################### ###=##PL/SQL########:=# ###########

######
#now()#timestamp###

##

```
quantity integer DEFAULT 32;
url varchar := 'http://mysite.com';
transaction_time CONSTANT timestamp with time zone := now();
```

################################## #########

```
DECLARE
  x integer := 1;
  y integer := x + 1;
```

41.3.1.

################$1#$2############# #############$*n*########################### ##### ####################################

##########2###########CREATE FUNCTION#################### ##########

```
CREATE FUNCTION sales_tax(subtotal real) RETURNS real AS $$
BEGIN
    RETURN subtotal * 0.06;
END;
$$ LANGUAGE plpgsql;
```

############################

name ALIAS FOR $*n*;

################

```
CREATE FUNCTION sales_tax(real) RETURNS real AS $$
DECLARE
    subtotal ALIAS FOR $1;
BEGIN
    RETURN subtotal * 0.06;
END;
$$ LANGUAGE plpgsql;
```

##

################ ######subtotal#sales_tax.subtotal############### ### #######################subtotal################

##########

```
CREATE FUNCTION instr(varchar, integer) RETURNS integer AS $$
DECLARE
    v_string ALIAS FOR $1;
    index ALIAS FOR $2;
BEGIN

    -- v_string #######################
END;
$$ LANGUAGE plpgsql;
```

```
CREATE FUNCTION concat_selected_fields(in_t sometablename) RETURNS text AS
 $$
BEGIN
    RETURN in_t.f1 || in_t.f3 || in_t.f5 || in_t.f7;
END;
$$ LANGUAGE plpgsql;
```

PL/pgSQL##$*n*######################## ##############NULL########################## ################## ############# ############

```
CREATE FUNCTION sales_tax(subtotal real, OUT tax real) AS $$
BEGIN
    tax := subtotal * 0.06;
END;
$$ LANGUAGE plpgsql;
```

RETURNS real################ ###################

OUT##

```
SELECT sales_tax(100.00);
```

########################## ###########

```
CREATE FUNCTION sum_n_product(x int, y int, OUT sum int, OUT prod int) AS
 $$
BEGIN
    sum := x + y;
    prod := x * y;
END;
$$ LANGUAGE plpgsql;

SELECT * FROM sum_n_product(2, 4);
 sum | prod
-----+------
   6 |    8
```

36.5.4## RETURNS##########RETURNS record############

#######################

```
CREATE PROCEDURE sum_n_product(x int, y int, OUT sum int, OUT prod int) AS
 $$
BEGIN
```

```
    sum := x + y;
    prod := x * y;
END;
$$ LANGUAGE plpgsql;
```

##################################### ###SQL#############################NULL#
######

```
CALL sum_n_product(2, 4, NULL, NULL);
 sum | prod
-----+------
   6 |    8
```

#######PL/pgSQL### ##
#41.6.3#########

PL/pgSQL###############RETURNS TABLE############ #########

```
CREATE FUNCTION extended_sales(p_itemno int)
RETURNS TABLE(quantity int, total numeric) AS $$
BEGIN
    RETURN QUERY SELECT s.quantity, s.quantity * s.price FROM sales AS s
                 WHERE s.itemno = p_itemno;
END;
$$ LANGUAGE plpgsql;
```

###############OUT###############RETURNS SETOF #######################

PL/pgSQL###########36.2.5################$0############ #######################
############ #########41.3.3######################### $0#NULL###################
##$0############### ##########+###
###################

```
CREATE FUNCTION add_three_values(v1 anyelement, v2 anyelement, v3
 anyelement)
RETURNS anyelement AS $$
DECLARE
    result ALIAS FOR $0;
BEGIN
    result := v1 + v2 + v3;
    RETURN result;
END;
$$ LANGUAGE plpgsql;
```

1### #######$0############## ##########
########### ##########

```
CREATE FUNCTION add_three_values(v1 anyelement, v2 anyelement, v3
 anyelement,
                                 OUT sum anyelement)
AS $$
BEGIN
    sum := v1 + v2 + v3;
END;
$$ LANGUAGE plpgsql;
```

######anycompatible################################## ####################
##########

```
CREATE FUNCTION add_three_values(v1 anycompatible, v2 anycompatible, v3
 anycompatible)
RETURNS anycompatible AS $$
BEGIN
    RETURN v1 + v2 + v3;
END;
$$ LANGUAGE plpgsql;
```

####

```
SELECT add_three_values(1, 2, 4.7);
```

############################## anyelement#########3#########################

41.3.2. ALIAS

```
newname ALIAS FOR oldname;
```

ALIAS################## ################################ ###################
#NEW#OLD##############################

##########

```
DECLARE
  prior ALIAS FOR old;
  updated ALIAS FOR new;
```

ALIAS#############2#################################### ####################
###############

41.3.3.

```
name table.column%TYPE
name variable%TYPE
```

#####[78.450363]#%TYPE#################### ################################## #
###users#####user_id############# users.user_id##########################
######%TYPE#################PL/pgSQL#############
#####users#####user_id############## users.user_id##################
########

```
user_id users.user_id%TYPE;
```

######%TYPE##

```
user_ids users.user_id%TYPE[];
user_ids users.user_id%TYPE ARRAY[4];  -- equivalent to the above
```

PostgreSQL##############
######################8.15.1##########

%TYPE################################## ####################################
###user_id#######integer##real################################

############################%TYPE############ ##################%TYPE#######
###################

41.3.4.

name table_name%ROWTYPE;
name composite_type_name;

########################## ################################SELECT#FOR#####
################## ###################rowvar.field#########################
##

table_name%ROWTYPE## ##########
################ ##################################PostgreSQL###%ROWTYPE####
################ ####%ROWTYPE###############

######%TYPE#####%ROWTYPE##

################################## ###########$*n*##################
##$1.user_id############

################## table1##table2## ##########################

```
CREATE FUNCTION merge_fields(t_row table1) RETURNS text AS $$
DECLARE
    t2_row table2%ROWTYPE;
BEGIN
    SELECT * INTO t2_row FROM table2 WHERE ... ;
    RETURN t_row.f1 || t2_row.f3 || t_row.f5 || t_row.f7;
END;
$$ LANGUAGE plpgsql;

SELECT merge_fields(t.*) FROM table1 t WHERE ... ;
```

41.3.5.

name RECORD;

##################################### ###SELECT#FOR######################## #
######################## ##
##########

RECORD#################################### PL/pgSQL###record###############
###################
####################################record##########################
##########

41.3.6. PL/pgSQL#####

PL/pgSQL################## 1#########23.2#################################
######### ##
#############

```
CREATE FUNCTION less_than(a text, b text) RETURNS boolean AS $$
BEGIN
    RETURN a < b;
END;
$$ LANGUAGE plpgsql;
```

```
SELECT less_than(text_field_1, text_field_2) FROM table1;
SELECT less_than(text_field_1, text_field_2 COLLATE "C") FROM table1;
```

##########less_than##text_field_1#text_field_2##################### ####
########C###########

############################

```
CREATE FUNCTION less_than(a text, b text) RETURNS boolean AS $$
DECLARE
    local_a text := a;
    local_b text := b;
BEGIN
    RETURN local_a < local_b;
END;
$$ LANGUAGE plpgsql;
```

##
###

###################COLLATE################################ #######

```
DECLARE
    local_a text COLLATE "en_US";
```

##

##COLLATE################## #######

```
CREATE FUNCTION less_than_c(a text, b text) RETURNS boolean AS $$
BEGIN
    RETURN a < b COLLATE "C";
END;
$$ LANGUAGE plpgsql;
```

###SQL###

41.4.

PL/pgSQL################SQL################## #######PL/pgSQL#

```
IF expression THEN ...
```

########PL/pgSQL#

```
SELECT expression
```

##SQL################## 41.11.1###############SELECT###########PL/pgSQL#######
###################### ######SELECT##
#####################PREPARE########## ####2######x#y######

```
IF x < y THEN ...
```

################

```
PREPARE statement_name(integer, integer) AS SELECT $1 < $2;
```

###################### ###IF#######PL/pgSQL################################ #####EXECUTE##### #######PL/pgSQL################################ ##### ##41.11.2#########

expression#SELECT################SELECT################################# #UNION#INTERSECT#EXCEPT############ ################################

```
IF count(*) > 0 FROM my_table THEN ...
```

IF#THEN####SELECT count(*) > 0 FROM my_table################ SELECT#1# ###2################### (################NULL###########)

41.5.

##############PL/pgSQL###################### ######################SQL##### #######41.5.2####################################

41.5.1.

##PL/pgSQL#####################

variable { := | = } *expression*;

#####################################SELECT SQL############### ##1########### #### (##############################)# ##########(##########)################### ########################### ###=##PL/SQL########:=############

#################################10.4############## ########################## ##PL/pgSQL## #### ##

##

```
tax := subtotal * 0.06;
my_record.user_id := 20;
my_array[j] := 20;
my_array[1:3] := array[1,2,3];
complex_array[n].realpart = 12.3;
```

41.5.2. SQL#######

##########SQL#####PL/pgSQL###################### ############################ ####

```
CREATE TABLE mytable (id int primary key, data text);
INSERT INTO mytable VALUES (1,'one'), (2,'two');
```

#####[89.441931]###############(###SELECT#RETURNING###INSERT/UPDATE/DELETE# #)#######2###### ########1##############################41.5.3############## #######INTO########################## ###############41.6.6##########FOR#### ######################## ####################SELECT#INSERT/UPDATE/DELETE/ MERGE#RETURNING#####2######### #######1#################################### ######41.5.3########INTO############### ##############41.6.6########FOR###### ###################

##########SQL###################### ######################################
############################ #########2#########

PL/pgSQL##########SQL####################SQL#####
##SELECT#INSERT#UPDATE#DELETE#MERGE#EXPLAIN#CREATE TABLE ... AS SELECT########1####################### ########################PL/pgSQL##
########################
##41.11.1#########

######SQL################41.11.2#########PL/pgSQL###########################

#########SQL####(#################)#################### #####PL/pgSQL#######
################### PL/pgSQL##################################41.5.4##########
######################EXECUTE##########

##EXECUTE############

####SELECT############################## ################################
####### ######PL/pgSQL###PERFORM##########

```
PERFORM query;
```

###*query*############### SQL#SELECT######*query*###############
#SELECT#PERFORM########## WITH##########PERFORM ########################
############1############## #######PL/pgSQL################################ ####
#######FOUND#########1####################################41.5.5##########

> ##
>
> ```
> ##SELECT##########################################PERFORM########
> SELECT########SQL################## ###INTO####################
> ```

#########

```
PERFORM create_mv('cs_session_page_requests_mv', my_query);
```

41.5.3. 1###############

#########1####SQL## ########SQL######
#######INTO################## #########

```
SELECT select_expressions INTO [STRICT] target FROM ...;
INSERT ... RETURNING expressions INTO [STRICT] target;
UPDATE ... RETURNING expressions INTO [STRICT] target;
DELETE ... RETURNING expressions INTO [STRICT] target;
MERGE ... RETURNING expressions INTO [STRICT] target;
```

#####[88.604353]#####*target*#########################/######################## #
#####PL/pgSQL###########(#####INTO#######)#################### #########
##SELECT#RETURNING####INSERT/UPDATE/DELETE####EXPLAIN################## ########### INTO#####SQL#####PL/pgSQL############### ########*target*######
##################/################## #######PL/pgSQL################INTO####
########################## ###SELECT#INSERT/UPDATE/DELETE/MERGE#RETURNING#
##########################EXPLAIN######## INTO#####SQL#####PL/pgSQL########
#######

> ###
>
> ###PostgreSQL#SELECT INTO####INTO#################### ####INTO### #SELECT############################ PL/pgSQL#####SELECT################ ###CREATE TABLE ... AS SELECT############

######################################

INTO##SQL########################## ######SELECT#####*select_expressions*# ################################ #########PL/pgSQL############################ ######

INTO#####STRICT#########*target*############################NULL###### ##### ####ORDER BY######################### 2################### #########FOUND## #41.5.5####################################

```
SELECT * INTO myrec FROM emp WHERE empname = myname;
IF NOT FOUND THEN
    RAISE EXCEPTION 'employee % not found', myname;
END IF;
```

STRICT#####################1############# ############NO_DATA_FOUND#2###### ###TOO_MANY_ROWS############## ######################### #########

```
BEGIN
    SELECT * INTO STRICT myrec FROM emp WHERE empname = myname;
    EXCEPTION
        WHEN NO_DATA_FOUND THEN
            RAISE EXCEPTION 'employee % not found', myname;
        WHEN TOO_MANY_ROWS THEN
            RAISE EXCEPTION 'employee % not unique', myname;
END;
```

STRICT###############FOUND#############

#####[86.714976]#PL/pgSQL#STRICT###########RETURNING### #INSERT/UPDATE/DELETE#2################## #######1########ORDER BY######## ########## ######RETURNING#####INSERT/UPDATE/DELETE/MERGE#####STRICT###### ########PL/pgSQL##################### #######################ORDER BY######## ########

print_strict_params################STRICT######################## #DETAIL########################## plpgsql.print_strict_params########## ######print_strict_params######### ######################## ############## ###################### #######

```
CREATE FUNCTION get_userid(username text) RETURNS int
AS $$
#print_strict_params on
DECLARE
userid int;
BEGIN
    SELECT users.userid INTO STRICT userid
        FROM users WHERE users.username = get_userid.username;
    RETURN userid;
END;
```

```
$$ LANGUAGE plpgsql;
```

```
##############################
```

```
ERROR:  query returned no rows
DETAIL:  parameters: $1 = 'nosuchuser'
CONTEXT:  PL/pgSQL function get_userid(text) line 6 at SQL statement
```

> **##**
>
> STRICT######Oracle#PL/SQL#SELECT INTO##############

41.5.4.

PL/pgSQL## PL/pgSQL###################41.11.2######################### ############### #EXECUTE##########

```
EXECUTE command-string [ INTO [STRICT] target ] [ USING expression
 [, ... ] ];
```

####command-string#############text############## ######*target*############ ###############/## ######USING## ######## #########

PL/pgSQL######################### ##################################### ###### ######################

###EXECUTE############################## ########################### ######## ##

INTO#######SQL###################### ## ## ##################INTO######## ## 1#########NULL#INTO######### INTO#######################

STRICT#######################1################

#########################$1#$2################## #######USING############## #### ##################################### ## ###########################SQL######################### #########

```
EXECUTE 'SELECT count(*) FROM mytable WHERE inserted_by = $1 AND inserted
 <= $2'
   INTO c
   USING checked_user, checked_date;
```

#################### ### ## ##

```
EXECUTE 'SELECT count(*) FROM '
    || quote_ident(tabname)
    || ' WHERE inserted_by = $1 AND inserted <= $2'
   INTO c
   USING checked_user, checked_date;
```

##########format()#%I################################

```
EXECUTE format('SELECT count(*) FROM %I '
   'WHERE inserted_by = $1 AND inserted <= $2', tabname)
   INTO c
   USING checked_user, checked_date;
```

(##################################SQL###########)

################SQL#####SELECT#INSERT#UPDATE#DELETE#MERGE########1###### ######################### ##

##################USING########EXECUTE########PL/pgSQL#####PL/pgSQL###### ################ ########EXECUTE## ### ###PL/pgSQL############################## ################################### ###########EXECUTE###########

SELECT INTO#EXECUTE############# #######SELECT########EXECUTE##### #INTO#########

> ##
>
> PL/pgSQL EXECUTE##PostgreSQL############EXECUTE SQL########### ### #EXECUTE##PL/pgSQL##############################

#41.1

##########################41.12.1#####################################

format()#### ###

```
EXECUTE format('UPDATE tbl SET %I = $1 '
   'WHERE key = $2', colname) USING newvalue, keyvalue;
```

####################

```
EXECUTE 'UPDATE tbl SET '
        || quote_ident(colname)
        || ' = '
        || quote_literal(newvalue)
        || ' WHERE key = '
        || quote_literal(keyvalue);
```

#####quote_ident#quote_literal###9.4################ ################### ################quote_ident################ ################################ ##quote_literal################ ## ##############################

quote_literal#STRICT############NULL#############NULL##### #### ##newvalue###keyvalue#NULL##############NULL####EXECUTE########### quote_nullable########################## ######NULL################NULL## #######quote_literal###### #########

```
EXECUTE 'UPDATE tbl SET '
```

```
        || quote_ident(colname)
        || ' = '
        || quote_nullable(newvalue)
        || ' WHERE key = '
        || quote_nullable(keyvalue);
```

NULL###################quote_literal#####quote_nullable#############

###############NULL################################## ########WHERE#########
######

```
'WHERE key = ' || quote_nullable(keyvalue)
```

###keyvalue#NULL########### ###########=#NULL#################NULL######
NULL##

```
'WHERE key IS NOT DISTINCT FROM ' || quote_nullable(keyvalue)
```

#######IS NOT DISTINCT FROM#=###
NULL#IS DISTINCT###########9.2###########

########################

```
EXECUTE 'UPDATE tbl SET '
        || quote_ident(colname)
        || ' = $$'
        || newvalue
        || '$$ WHERE key = '
        || quote_literal(keyvalue);
```

#####newvalue########$$######################## ############################
######################################
##quote_literal#quote_nullable####quote_ident##################

###SQL##format###9.4.1##########################

```
EXECUTE format('UPDATE tbl SET %I = %L '
   'WHERE key = %L', colname, newvalue, keyvalue);
```

%I#quote_ident#####%L#quote_nullable###### format###USING###########

```
EXECUTE format('UPDATE tbl SET %I = $1 WHERE key = $2', colname)
   USING newvalue, keyvalue;
```

##################%L##

########EXECUTE####### 41.10###### ###############CREATE FUNCTION###########
######

41.5.5.

######################### ##############GET DIAGNOSTICS#########

```
GET [ CURRENT ] DIAGNOSTICS variable { = | := } item [ , ... ];
```

#################################### CURRENT############41.6.8.1#GET STACKED
DIAGNOSTICS########### #*item*#######*variable*#################################

#################### ################# 41.1######### #####:=####SQL#######=##### ######## #########

```
GET DIAGNOSTICS integer_var = ROW_COUNT;
```

#41.1 ############

##	#	##
ROW_COUNT	bigint	###SQL##############
PG_CONTEXT	text	##################### #41.6.9####
PG_ROUTINE_OID	oid	######OID

###########2#######FOUND###boolean################ PL/pgSQL############## ##FOUND############## ############################

- SELECT INTO##########################FOUND#######
- PERFORM###1####################################FOUND#######
- UPDATE#INSERT#DELETE####MERGE########1############################ #FOUND#######
- FETCH#############################FOUND#######
- MOVE############################FOUND#######
- FOR####FOREACH###1############################FOUND####### FOUND######### ############## ##############FOUND########## ################################ ########
- RETURN QUERY#RETURN QUERY EXECUTE##############################FOUND## #####

##PL/pgSQL##FOUND########## ###EXECUTE#GET DIAGNOSTICS##########FOUND### ##############

FOUND######PL/pgSQL############## FOUND############################

41.5.6.

########################## ####IF/THEN/ELSE##################### ######## #NULL########

```
NULL;
```

######2###########

```
BEGIN
    y := x / 0;
EXCEPTION
    WHEN division_by_zero THEN

        NULL;  -- #######
END;
```

```
BEGIN
```

```
    y := x / 0;
EXCEPTION

    WHEN division_by_zero THEN  -- #######
END;
```

#####################

> **##**
>
> Oracle#PL/SQL#######################NULL###### ###PL/pgSQL########### #####

41.6.

#########PL/pgSQL################ PL/pgSQL##########PostgreSQL############### #########

41.6.1.

#####################2###### RETURN###RETURN NEXT###

41.6.1.1. RETURN

```
RETURN expression;
```

####RETURN#######*expression*############# ###########PL/pgSQL#########

############################

#######################RETURN######### #######################

void##############################RETURN#########RETURN###############

######################### ####RETURN#################################### ######   #######################void###############   ################### ##RETURN############

#######

```
-- #########
RETURN 1 + 2;
RETURN scalar_var;


-- ########
RETURN composite_type_var;

RETURN (1, 2, 'three'::text);  -- ####################
```

41.6.1.2. RETURN NEXT###RETURN QUERY

```
RETURN NEXT expression;
RETURN QUERY query;
RETURN QUERY EXECUTE command-string [ USING expression [, ... ] ];
```

PL/pgSQL###SETOF *sometype*######################### ###############RETURN NEXT######RETURN QUERY########### ###########RETURN#################### ##### RETURN NEXT########################## ######################## RETURN QUERY########################### RETURN NEXT#RETURN QUERY############### ######## ####################

#####RETURN NEXT###RETURN QUERY########## ###################### ######## PL/pgSQL############# RETURN NEXT###RETURN QUERY######################## ## ###############RETURN############## (####################)#

RETURN QUERY##RETURN QUERY EXECUTE################################## ##### ###EXECUTE#############USING############################

#######################RETURN NEXT########### ############################## ################### ######################################SETOF record###### #########*sometype*###########SETOF *sometype*############################

RETURN NEXT#################

```
CREATE TABLE foo (fooid INT, foosubid INT, fooname TEXT);
INSERT INTO foo VALUES (1, 2, 'three');
INSERT INTO foo VALUES (4, 5, 'six');

CREATE OR REPLACE FUNCTION get_all_foo() RETURNS SETOF foo AS
$BODY$
DECLARE
    r foo%rowtype;
BEGIN
    FOR r IN
        SELECT * FROM foo WHERE fooid > 0
    LOOP

        -- ############
        RETURN NEXT r; -- SELECT##########
    END LOOP;
    RETURN;
END;
$BODY$
LANGUAGE plpgsql;

SELECT * FROM get_all_foo();
```

RETURN QUERY#################

```
CREATE FUNCTION get_available_flightid(date) RETURNS SETOF integer AS
$BODY$
BEGIN
    RETURN QUERY SELECT flightid
                   FROM flight
                  WHERE flightdate >= $1
                    AND flightdate < ($1 + 1);


-- ##########################################
    IF NOT FOUND THEN
        RAISE EXCEPTION 'No flight at %.', $1;
    END IF;
```

```
    RETURN;
 END;
$BODY$
LANGUAGE plpgsql;


-- ##################################
SELECT * FROM get_available_flightid(CURRENT_DATE);
```

> ##
>
> #######RETURN NEXT###RETURN QUERY################################ ####
> ##PL/pgSQL#################################### ##########################
> ### ############################ ###PL/pgSQL###########################
> ################## #######################work_mem################ #####
> ##

41.6.2.

################ #############RETURN########### #############RETURN###########
######RETURN###########

##

41.6.3.

PL/pgSQL###########DO######CALL################### CALL####SQL###########
############### ########OUT####INOUT######CALL############################
##CALL################### ##########

```
CREATE PROCEDURE triple(INOUT x int)
LANGUAGE plpgsql
AS $$
BEGIN
    x := x * 3;
END;
$$;

DO $$
DECLARE myvar int := 5;
BEGIN
  CALL triple(myvar);
  RAISE NOTICE 'myvar = %', myvar;  -- prints 15
END;
$$;
```

#################################### ################

41.6.4.

IF#CASE########################## PL/pgSQL############IF#########

- IF ... THEN ... END IF
- IF ... THEN ... ELSE ... END IF

• IF ... THEN ... ELSIF ... THEN ... ELSE ... END IF

###########CASE########

• CASE ... WHEN ... THEN ... ELSE ... END CASE

• CASE WHEN ... THEN ... ELSE ... END CASE

41.6.4.1. IF-THEN

```
IF boolean-expression THEN
    statements
END IF;
```

IF-THEN######IF##### THEN#END IF################## ################

##

```
IF v_user_id <> 0 THEN
    UPDATE users SET email = v_email WHERE user_id = v_user_id;
END IF;
```

41.6.4.2. IF-THEN-ELSE

```
IF boolean-expression THEN
    statements
ELSE
    statements
END IF;
```

IF-THEN-ELSE##IF-THEN## ########NULL#
#######################

##

```
IF parentid IS NULL OR parentid = ''
THEN
    RETURN fullname;
ELSE
    RETURN hp_true_filename(parentid) || '/' || fullname;
END IF;
```

```
IF v_count > 0 THEN
    INSERT INTO users_count (count) VALUES (v_count);
    RETURN 't';
ELSE
    RETURN 'f';
END IF;
```

41.6.4.3. IF-THEN-ELSIF

```
IF boolean-expression THEN
    statements
[ ELSIF boolean-expression THEN
    statements
[ ELSIF boolean-expression THEN
```

```
    statements
    ...
]
]
[ ELSE
    statements ]
END IF;
```

######################### IF-THEN-ELSIF############################## IF## ########################### ################END IF############### ######IF### ############## ###IF#########ELSE####################

```
IF number = 0 THEN
    result := 'zero';
ELSIF number > 0 THEN
    result := 'positive';
ELSIF number < 0 THEN
    result := 'negative';
ELSE

    -- #################NULL######
    result := 'NULL';
END IF;
```

ELSIF######ELSEIF#############

######################IF-THEN-ELSE############

```
IF demo_row.sex = 'm' THEN
    pretty_sex := 'man';
ELSE
    IF demo_row.sex = 'f' THEN
        pretty_sex := 'woman';
    END IF;
END IF;
```

#############IF#####END IF######### ##############ELSIF############

41.6.4.4. ###CASE

```
CASE search-expression
    WHEN expression [, expression [ ... ]] THEN
      statements
  [ WHEN expression [, expression [ ... ]] THEN
      statements
    ... ]
  [ ELSE
      statements ]
END CASE;
```

CASE############################## *search-expression*##############WHEN## ######*expression*####### ################*statements*######END CASE####### ###### ####WHEN########## ###############ELSE *statements*########ELSE##### #CASE_NOT_FOUND##########

#########

```
CASE x
    WHEN 1, 2 THEN
        msg := 'one or two';
    ELSE
        msg := 'other value than one or two';
END CASE;
```

41.6.4.5. ####CASE

```
CASE
    WHEN boolean-expression THEN
      statements
  [ WHEN boolean-expression THEN
      statements
    ... ]
  [ ELSE
      statements ]
END CASE;
```

CASE################################ #####WHEN##*boolean-expression*#true# ################### #######*statements*#########END CASE########### ## ####WHEN########## ###############ELSE *statements*########ELSE######## #CASE_NOT_FOUND##########

#########

```
CASE
    WHEN x BETWEEN 0 AND 10 THEN
        msg := 'value is between zero and ten';
    WHEN x BETWEEN 11 AND 20 THEN
        msg := 'value is between eleven and twenty';
END CASE;
```

#####CASE###########ELSE##########################IF-THEN-ELSIF#######

41.6.5.

LOOP#EXIT#CONTINUE#WHILE#FOR#FOREACH#######PL/pgSQL#################### ##

41.6.5.1. LOOP

```
[ <<label>> ]
LOOP
    statements
END LOOP [ label ];
```

LOOP##EXIT####RETURN################################## #####*label*######## #####EXIT###CONTINUE##############################

41.6.5.2. EXIT

```
EXIT [ label ] [ WHEN boolean-expression ];
```

label######################END LOOP############## *label*################ ############################## ##END## ##########

WHEN########*boolean-expression*################## #######EXIT############

EXIT#################### ####################

BEGIN############EXIT################## ################################## ######EXIT#BEGIN#################### ##########EXIT#BEGIN################ PostgreSQL#8.4################

##

```
LOOP

    -- ######
    IF count > 0 THEN

        EXIT;  -- ########
    END IF;
END LOOP;

LOOP

-- ######
    EXIT WHEN count > 0;  -- #######
END LOOP;

<<ablock>>
BEGIN

    -- ######
    IF stocks > 100000 THEN

        EXIT ablock;  -- #####BEGIN#########
    END IF;

    -- stocks > 100000 #############
END;
```

41.6.5.3. CONTINUE

CONTINUE [*label*] [WHEN *boolean-expression*];

label########################## ## ##########(#######)###### *label*##########################

WHEN########*boolean-expression*######################### #######CONTINUE# ###########

CONTINUE################# ################

#

```
LOOP

    -- ######
    EXIT WHEN count > 100;
    CONTINUE WHEN count < 50;

    -- 50##100###########
END LOOP;
```

41.6.5.4. WHILE

```
[ <<label>> ]
WHILE boolean-expression LOOP
    statements
END LOOP [ label ];
```

WHILE##*boolean-expression*#################### ######################

##########

```
WHILE amount_owed > 0 AND gift_certificate_balance > 0 LOOP

    -- ###############
END LOOP;

WHILE NOT done LOOP

    -- ###############
END LOOP;
```

41.6.5.5. ##FOR###

```
[ <<label>> ]
FOR name IN [ REVERSE ] expression .. expression [ BY expression ] LOOP
    statements
END LOOP [ label ];
```

#####FOR#################### *name*###integer######################## ###### ###################### ###############2###################### BY########### ####1####BY########################### REVERSE############################ ############

##FOR#############

```
FOR i IN 1..10 LOOP

    -- i ###### 1#2#3#4#5#6#7#8#9#10 ########
END LOOP;

FOR i IN REVERSE 10..1 LOOP

    -- i ###### 10#9#8#7#6#5#4#3#2#1 ########
END LOOP;

FOR i IN REVERSE 10..1 BY 2 LOOP

    -- i ###### 10#8#6#4#2 ########
END LOOP;
```

###########REVERSE############################ ###########

label#FOR#############*label*#######################

41.6.6.

#####FOR#################################### ###########

```
[ <<label>> ]
```

```
FOR target IN query LOOP
    statements
END LOOP [ label ];
```

target################################ *target*#####*query*################## ############# #########

```
CREATE FUNCTION refresh_mviews() RETURNS integer AS $$
DECLARE
    mviews RECORD;
BEGIN
    RAISE NOTICE 'Refreshing all materialized views...';

    FOR mviews IN
       SELECT n.nspname AS mv_schema,
              c.relname AS mv_name,
              pg_catalog.pg_get_userbyid(c.relowner) AS owner
         FROM pg_catalog.pg_class c
    LEFT JOIN pg_catalog.pg_namespace n ON (n.oid = c.relnamespace)
        WHERE c.relkind = 'm'
     ORDER BY 1
    LOOP


        -- ###"mviews"################1###########

        RAISE NOTICE 'Refreshing materialized view %.% (owner: %)...',
                     quote_ident(mviews.mv_schema),
                     quote_ident(mviews.mv_name),
                     quote_ident(mviews.owner);
        EXECUTE format('REFRESH MATERIALIZED VIEW %I.%I',
 mviews.mv_schema, mviews.mv_name);
    END LOOP;

    RAISE NOTICE 'Done refreshing materialized views.';
    RETURN 1;
END;
$$ LANGUAGE plpgsql;
```

######EXIT##

#####[87.133183]######FOR##*query*##############SQL############## ###SELECT## ##RETURNING####INSERT#UPDATE###DELETE####### EXPLAIN################## ## ###########FOR#######*query*##############SQL###### #########SELECT## ##INSERT#UPDATE#DELETE#MERGE#RETURNING############# EXPLAIN############ ########

PL/pgSQL###################### #########41.11.1###41.11.2###################### ##

FOR-IN-EXECUTE#########1#######

```
[ <<label>> ]
FOR target IN EXECUTE text_expression [ USING expression [, ... ] ] LOOP
    statements
END LOOP [ label ];
```

####FOR###################### ####### ########EXECUTE###
EXECUTE#############USING################

###41.7.4#######

41.6.7.

FOREACH####FOR############## ####SQL ################################ #####FOREACH### ###### #FOREACH#######

```
[ <<label>> ]
FOREACH target [ SLICE number ] IN ARRAY expression LOOP
    statements
END LOOP [ label ];
```

SLICE######SLICE 0############*expression*########################## *target*################################### ##################

```
CREATE FUNCTION sum(int[]) RETURNS int8 AS $$
DECLARE
  s int8 := 0;
  x int;
BEGIN
  FOREACH x IN ARRAY $1
  LOOP
    s := s + x;
  END LOOP;
  RETURN s;
END;
$$ LANGUAGE plpgsql;
```

############################ ##*target*## ## ##

##SLICE######FOREACH################################ SLICE############### ############ *target*############### ######################### ########SLICE### ######### ###1######################

```
CREATE FUNCTION scan_rows(int[]) RETURNS void AS $$
DECLARE
  x int[];
BEGIN
  FOREACH x SLICE 1 IN ARRAY $1
  LOOP
    RAISE NOTICE 'row = %', x;
  END LOOP;
END;
$$ LANGUAGE plpgsql;

SELECT scan_rows(ARRAY[[1,2,3],[4,5,6],[7,8,9],[10,11,12]]);

NOTICE:  row = {1,2,3}
NOTICE:  row = {4,5,6}
NOTICE:  row = {7,8,9}
NOTICE:  row = {10,11,12}
```

41.6.8.

########PL/pgSQL####################################### BEGIN###### #EXCEPTION########################### ########BEGIN################

```
[ <<label>> ]
```

```
[ DECLARE
    declarations ]
BEGIN
    statements
EXCEPTION
    WHEN condition [ OR condition ... ] THEN
        handler_statements
    [ WHEN condition [ OR condition ... ] THEN
          handler_statements
      ... ]
END;
```

##########################statements#####END############ ####statements#### ##############statements########EXCEPTION########### ################### #####condition###### ##############handler_statements#####END############ ###########EXCEPTION######################### EXCEPTION################### #####################

###condition######A#################### ######################### OTHERS#### ######QUERY_CANCELED#ASSERT_FAILURE############### #QUERY_CANCELED#ASSERT_FAILURE########################### ############### ## ##########SQLSTATE########### ###########

```
WHEN division_by_zero THEN ...
WHEN SQLSTATE '22012' THEN ...
```

########handler_statements###########EXCEPTION######################### # #####EXCEPTION##############

EXCEPTION############PL/pgSQL########################### ################### ################### ##############

```
INSERT INTO mytab(firstname, lastname) VALUES('Tom', 'Jones');
BEGIN
    UPDATE mytab SET firstname = 'Joe' WHERE lastname = 'Jones';
    x := x + 1;
    y := x / 0;
EXCEPTION
    WHEN division_by_zero THEN
        RAISE NOTICE 'caught division_by_zero';
        RETURN x;
END;
```

#####y########division_by_zero######EXCEPTION############# RETURN###### #####1######x########UPDATE################### ###########INSERT######### ####### ####################Tom Jones####Joe Jones########

> ###
>
> EXCEPTION## ############ #EXCEPTION############

#41.2 UPDATE/INSERT###

###UPDATE###INSERT######################## ########################ON CONFLICT DO UPDATE####INSERT####################PL/pgSQL################

```
CREATE TABLE db (a INT PRIMARY KEY, b TEXT);

CREATE FUNCTION merge_db(key INT, data TEXT) RETURNS VOID AS
$$
BEGIN
    LOOP

        -- ##########
        UPDATE db SET b = data WHERE a = key;
        IF found THEN
            RETURN;
        END IF;

        -- #####################
        -- #####################
        -- ###############
        BEGIN
            INSERT INTO db(a,b) VALUES (key, data);
            RETURN;
        EXCEPTION WHEN unique_violation THEN

            -- ################
        END;
    END LOOP;
END;
$$
LANGUAGE plpgsql;

SELECT merge_db(1, 'david');
SELECT merge_db(1, 'dennis');
```

##########unique_violation######INSERT#####################INSERT####### ####### #######2## # ##

41.6.8.1.

################################## PL/pgSQL##################2###### ##### #GET STACKED DIAGNOSTICS######

################SQLSTATE##################### A.1######################### ### ##SQLERRM###################### ############################

############GET STACKED DIAGNOSTICS################################ ###### ########

```
GET STACKED DIAGNOSTICS variable { = | := } item [ , ... ];
```

#*item*#######*variable*## ###### ########## 41.2##########

#41.2

##	#	##
RETURNED_SQLSTATE	text	###SQLSTATE######
COLUMN_NAME	text	########
CONSTRAINT_NAME	text	#########
PG_DATATYPE_NAME	text	###########
MESSAGE_TEXT	text	################

##	#	##
TABLE_NAME	text	##########
SCHEMA_NAME	text	##########
PG_EXCEPTION_DETAIL	text	######################
PG_EXCEPTION_HINT	text	#########################
PG_EXCEPTION_CONTEXT	text	######################### #41.6.9####

```
##########################

##########


DECLARE
  text_var1 text;
  text_var2 text;
  text_var3 text;
BEGIN

  -- ##########
  ...
EXCEPTION WHEN OTHERS THEN
  GET STACKED DIAGNOSTICS text_var1 = MESSAGE_TEXT,
                          text_var2 = PG_EXCEPTION_DETAIL,
                          text_var3 = PG_EXCEPTION_HINT;
END;
```

41.6.9.

###41.5.5#######GET DIAGNOSTICS################################GET STACKED DIAGNOSTICS######################### ###PG_CONTEXT#################### ###### PG_CONTEXT################################## #################GET DIAGNOSTICS########## ################################ #######

```
CREATE OR REPLACE FUNCTION outer_func() RETURNS integer AS $$
BEGIN
  RETURN inner_func();
END;
$$ LANGUAGE plpgsql;

CREATE OR REPLACE FUNCTION inner_func() RETURNS integer AS $$
DECLARE
  stack text;
BEGIN
  GET DIAGNOSTICS stack = PG_CONTEXT;
  RAISE NOTICE E'--- Call Stack ---\n%', stack;
  RETURN 1;
END;
$$ LANGUAGE plpgsql;

SELECT outer_func();

NOTICE:  --- Call Stack ---
PL/pgSQL function inner_func() line 5 at GET DIAGNOSTICS
PL/pgSQL function outer_func() line 3 at RETURN
CONTEXT:  PL/pgSQL function outer_func() line 3 at RETURN
 outer_func
 ------------
           1
```

```
(1 row)
```

GET STACKED DIAGNOSTICS ... PG_EXCEPTION_CONTEXT#####################
##########################

41.7.

########
1############################## #####PL/pgSQL#################### FOR########
########################## ##
######## ############################

41.7.1.

PL/pgSQL############################## ############refcursor###### ########
##1#######refcursor############### ####################### #############

name [[NO] SCROLL] CURSOR [(*arguments*)] FOR *query*;

#Oracle########FOR#IS############## ##SCROLL##################### #
#NO SCROLL####################### ################################## #
#*arguments*#####*name datatype*##
##################################

#########

```
DECLARE
    curs1 refcursor;
    curs2 CURSOR FOR SELECT * FROM tenk1;
    curs3 CURSOR (key integer) FOR SELECT * FROM tenk1 WHERE unique1 =
 key;
```

###3######refcursor##############################2#######################
############ ################################ #key##########################
curs1##

##########FOR UPDATE/SHARE########SCROLL########### ##################NO
SCROLL########## SCROLL###
#########

41.7.2.

################################## ####DECLARE CURSOR SQL############# PL/
pgSQL##3###OPEN#########2#########################1####################

> ##
>
> #############41.7.4#######FOR######################## FOR#############
> #############

###############
########################### ########PL/pgSQL####################### ###########
NULL################################### #####41.7.3.5#################

41.7.2.1. OPEN FOR *query*

```
OPEN unbound_cursorvar [ [ NO ] SCROLL ] FOR query;
```

################################ ##
###refcursor################### #######SELECT#########EXPLAIN#########
############### ##########PL/pgSQL#SQL############# PL/pgSQL##############
####################### PL/pgSQL##########################OPEN#############
############## SCROLL###NO SCROLL#########################

#########

```
OPEN curs1 FOR SELECT * FROM foo WHERE key = mykey;
```

41.7.2.2. OPEN FOR EXECUTE

```
OPEN unbound_cursorvar [ [ NO ] SCROLL ] FOR EXECUTE query_string
                                    [ USING expression [, ... ] ];
```

################################ ##
###refcursor################### ######EXECUTE######################### #
##(41.11.2##)# ########################
EXECUTE####format()#USING############################ SCROLL###NO
SCROLL#########################

#########

```
OPEN curs1 FOR EXECUTE format('SELECT * FROM %I WHERE col1 = $1',tabname)
 USING keyvalue;
```

#################format()######### col1######USING#######################
####

41.7.2.3.

```
OPEN bound_cursorvar [ ( [ argument_name := ] argument_value [, ...] ) ];
```

################################OPEN###### ################### ###############
################### #####################

################################### #####EXECUTE############ SCROLL###NO
SCROLL#OPEN################### #######################

################################ ####################### ##################
#:=##################### 4.3##################################

###########################

```
OPEN curs2;
OPEN curs3(42);
OPEN curs3(key := 42);
```

##2###### OPEN#######################
#PL/pgSQL################# ################################### #######OPEN#
############# ######curs3####################

```
DECLARE
    key integer;
    curs4 CURSOR FOR SELECT * FROM tenk1 WHERE unique1 = key;
BEGIN
    key := 42;
    OPEN curs4;
```

41.7.3.

###################################

################################### ####refcursor########################### ##### ######refcursor#################################### ################## #refcursor##########################

############################### ######refcursor########################### ##########

41.7.3.1. FETCH

```
FETCH [ direction { FROM | IN } ] cursor INTO target;
```

FETCH#SELECT INTO#################################### ##################### ################ ###############NULL####### SELECT INTO#########FOUND#### ######################### ##

direction#######################SQL FETCH################### ############ NEXT, PRIOR, FIRST, LAST, ABSOLUTE *count*, RELATIVE *count*, FORWARD### BACKWARD. *direction*######NEXT######## *count*#######*count*###############(SQL FETCH## ###########################) SCROLL#########################*direction*######## #############

cursor#####################refcursor#############

##

```
FETCH curs1 INTO rowvar;
FETCH curs2 INTO foo, bar, baz;
FETCH LAST FROM curs3 INTO x, y;
FETCH RELATIVE -2 FROM curs4 INTO x;
```

41.7.3.2. MOVE

```
MOVE [ direction { FROM | IN } ] cursor;
```

MOVE############################## MOVE################################## ##FETCH############## *direction*###SQL FETCH############################## ############ ################### #####*direction*###########*count*#######PL/pgSQL####### ######*direction*#######################*count*################ ##### SELECT INTO#########FOUND########################## ################ ############################

#:

```
MOVE curs1;
```

```
MOVE LAST FROM curs3;
MOVE RELATIVE -2 FROM curs4;
MOVE FORWARD 2 FROM curs4;
```

41.7.3.3. UPDATE/DELETE WHERE CURRENT OF

```
UPDATE table SET ... WHERE CURRENT OF cursor;
DELETE FROM table WHERE CURRENT OF cursor;
```

################################
####################FOR UPDATE############ #########DECLARE##############

##########

```
UPDATE foo SET dataval = myval WHERE CURRENT OF curs1;
```

41.7.3.4. CLOSE

```
CLOSE cursor;
```

CLOSE######################### ##
#################

##

```
CLOSE curs1;
```

41.7.3.5.

PL/pgSQL######################## ##
##
############## ################################### ##################### ####
########################

#################################### ##################refcursor#########
refcursor#######OPEN##################### ####refcursor#####NULL#####
###############OPEN##########################refcursor########

> ##
>
> PostgreSQL 16##################NULL################################
> ############################## ##
> ##########

#################################

```
CREATE TABLE test (col text);
INSERT INTO test VALUES ('123');

CREATE FUNCTION reffunc(refcursor) RETURNS refcursor AS '
BEGIN
    OPEN $1 FOR SELECT col FROM test;
    RETURN $1;
END;
```

```
' LANGUAGE plpgsql;

BEGIN;
SELECT reffunc('funccursor');
FETCH ALL IN funccursor;
COMMIT;

##############################


CREATE FUNCTION reffunc2() RETURNS refcursor AS '
DECLARE
    ref refcursor;
BEGIN
    OPEN ref FOR SELECT col FROM test;
    RETURN ref;
END;
' LANGUAGE plpgsql;


-- ##################################
BEGIN;
SELECT reffunc2();

      reffunc2
--------------------
 <unnamed cursor 1>
(1 row)

FETCH ALL IN "<unnamed cursor 1>";
COMMIT;

##############################


CREATE FUNCTION myfunc(refcursor, refcursor) RETURNS SETOF refcursor AS $$
BEGIN
    OPEN $1 FOR SELECT * FROM table_1;
    RETURN NEXT $1;
    OPEN $2 FOR SELECT * FROM table_2;
    RETURN NEXT $2;
END;
$$ LANGUAGE plpgsql;


-- ##################################
BEGIN;

SELECT * FROM myfunc('a', 'b');

FETCH ALL FROM a;
FETCH ALL FROM b;
COMMIT;
```

41.7.4.

```
#######################FOR########## ##############


[ <<label>> ]
FOR recordvar IN bound_cursorvar [ ( [ argument_name := ] argument_value
 [, ...] ) ] LOOP
    statements
END LOOP [ label ];
```

FOR######## ######################## ## ##### ##OPEN#######################41.7.2.3##########

recordvar####record######################## ############################ ## ##

41.8.

CALL################################DO########COMMIT###ROLLBACK############ ################ ##START TRANSACTION####### #PL/pgSQL##BEGIN#END####################

##########

```
CREATE PROCEDURE transaction_test1()
LANGUAGE plpgsql
AS $$
BEGIN
    FOR i IN 0..9 LOOP
        INSERT INTO test1 (a) VALUES (i);
        IF i % 2 = 0 THEN
            COMMIT;
        ELSE
            ROLLBACK;
        END IF;
    END LOOP;
END;
$$;

CALL transaction_test1();
```

####COMMIT AND CHAIN#ROLLBACK AND CHAIN###########

##CALL###DO########CALL###DO########## ##### ############CALL proc1() → CALL proc2() → CALL proc3()############ ######################### ############CALL proc1() → SELECT func2() → CALL proc3()########SELECT################################

######PL/pgSQL#########################/######TO#######/################)# ## ##41.6.8##### ######################### ########### ######################################

######################### ################

```
CREATE PROCEDURE transaction_test2()
LANGUAGE plpgsql
AS $$
DECLARE
    r RECORD;
BEGIN
    FOR r IN SELECT * FROM test2 ORDER BY x LOOP
        INSERT INTO test1 (a) VALUES (r.x);
        COMMIT;
    END LOOP;
END;
$$;
```

```
CALL transaction_test2();
```

#####[68.597561]################################# ############################ #####COMMIT###ROLLBACK#################### ###########COMMIT#ROLLBACK### ############################## ## ### ##################################### ### ############################ ## ###### ## ############################### ##

```
##############################UPDATE ... RETURNING#######################
```

41.9.

41.9.1.

RAISE##############################

```
RAISE [ level ] 'format' [, expression [, ... ]] [ USING option
 = expression [, ... ] ];
RAISE [ level ] condition_name [ USING option = expression [, ... ] ];
RAISE [ level ] SQLSTATE 'sqlstate' [ USING option = expression
 [, ... ] ];
RAISE [ level ] USING option = expression [, ... ];
RAISE ;
```

level################### ########DEBUG#LOG#INFO#NOTICE#WARNING## #EXCEPTION##EXCEPTION######## EXCEPTION############################ # ############################## ## #log_min_messages###client_min_messages############## ########19##########

#####*level*###*format*######### ################################ ############## ############### ######### %########################## %%#######%############ ## #################%######################################

```
#######v_job_id#######%########
```

```
RAISE NOTICE 'Calling cs_create_job(%)', v_job_id;
```

USING#####*option* = *expression*################################ #*expression*################### #####*option*##########

MESSAGE

  ################### ########USING##########RAISE###########

DETAIL

  ################

HINT

  ###############

ERRCODE

  ##A#######################SQLSTATE#####################SQLSTATE###### ##

COLUMN
CONSTRAINT
DATATYPE
TABLE
SCHEMA

##################

##

```
RAISE EXCEPTION 'Nonexistent ID --> %', user_id
      USING HINT = 'Please check your user ID';
```

#########SQLSTATE##################

```
RAISE 'Duplicate user ID: %', user_id USING ERRCODE = 'unique_violation';
RAISE 'Duplicate user ID: %', user_id USING ERRCODE = '23505';
```

##################SQLSTATE##########RAISE######## #######

```
RAISE division_by_zero;
RAISE SQLSTATE '22012';
```

#########USING#################################### #########################

```
RAISE unique_violation USING MESSAGE = 'Duplicate user ID: ' || user_id;
```

#########RAISE USING ### RAISE *level* USING###########USING###########

###RAISE################ #####BEGIN#####EXCEPTION########### ############### #######

> ##
>
> PostgreSQL9.1####################RAISE################################### ### #####################EXCEPTION###RAISE#######EXCEPTION######### ###################### ################# PL/SQL########

RAISE EXCEPTION#########SQLSTATE###############raise_exception(P0001)# ###### ##############################SQLSTATE##################

> ##
>
> SQLSTATE## 00000######## ASCII######################## ################################ ######### ##

41.9.2.

ASSERT###PL/pgSQL#########################

```
ASSERT condition [ , message ];
```

condition##########################ASSERT################# #####NULL#
##ASSERT_FAILURE######## ####*condition*################################
###

#####*message*################NULL######*condition*##################
##assertion failed########## *message*#######################

##############plpgsql.check_asserts###########################on### off##
####ASSERT########

ASSERT### #########RAISE###
######

41.10.

PL/pgSQL################################### #######CREATE FUNCTION########
(#########)trigger#(##############)event_trigger#########################
########################TG_*something*#######################

41.10.1.

########trigger########################### #########CREATE TRIGGER######
##############TG_ARGV########

PL/pgSQL## ###########

NEW record

#########INSERT/UPDATE############### ################DELETE####NULL##
####

OLD record

#########UPDATE/DELETE################### ################INSERT####
NULL######

TG_NAME name

###########

TG_WHEN text

###########BEFORE#AFTER####INSTEAD OF#

TG_LEVEL text

###########ROW###STATEMENT#

TG_OP text

#########: INSERT#UPDATE#DELETE#### TRUNCATE#

TG_RELID oid (references pg_class.oid)

###################ID#

TG_RELNAME name

############# ################################## ####TG_TABLE_NAME#######
##

TG_TABLE_NAME name

   ############

TG_TABLE_SCHEMA name

   #################

TG_NARGS integer

   CREATE TRIGGER###################

TG_ARGV text[]

   CREATE TRIGGER###### ########0####### #########0###tg_nargs#####NULL#
   ##########

######NULL################################/##############

BEFORE###############NULL##################################### ##########
#########INSERT/UPDATE/DELETE################ #NULL####################
##NEW############################
###NEW(########)############ ##############NEW################NEW#########
####/################## DELETE####BEFORE################################
#######NULL############ DELETE#####NEW#NULL############################
DELETE############OLD#######

INSTEAD OF##
########################## #######INSERT/UPDATE/DELETE####################
#######NULL########## ##################################NULL##############
INSERT###UPDATE###############INSERT RETURNING###UPDATE RETURNING######
################NEW####### ##############ON CONFLICT DO UPDATE####INSERT#
######EXCLUDED################### DELETE########OLD#######

#####AFTER########BEFORE###AFTER################ NULL######### ##########
##################################

41.3#PL/pgSQL#############

#41.3 PL/pgSQL#####

#################
####################

```
CREATE TABLE emp (
    empname           text,
    salary            integer,
    last_date         timestamp,
    last_user         text
);

CREATE FUNCTION emp_stamp() RETURNS trigger AS $emp_stamp$
    BEGIN

        -- empname#salary###############
        IF NEW.empname IS NULL THEN
            RAISE EXCEPTION 'empname cannot be null';
        END IF;
        IF NEW.salary IS NULL THEN
            RAISE EXCEPTION '% cannot have null salary', NEW.empname;
        END IF;
```

```
        -- ###############
        IF NEW.salary < 0 THEN
            RAISE EXCEPTION '% cannot have a negative salary',
 NEW.empname;
        END IF;


        -- ############
        NEW.last_date := current_timestamp;
        NEW.last_user := current_user;
        RETURN NEW;
    END;
$emp_stamp$ LANGUAGE plpgsql;

CREATE TRIGGER emp_stamp BEFORE INSERT OR UPDATE ON emp
    FOR EACH ROW EXECUTE FUNCTION emp_stamp();
```

41.4#PL/pgSQL###################

#41.4 PL/pgSQL############

##########emp#########################emp_audit######################### ######## ###############emp_audit###########

```
CREATE TABLE emp (
    empname           text NOT NULL,
    salary            integer
);

CREATE TABLE emp_audit(
    operation         char(1)   NOT NULL,
    stamp             timestamp NOT NULL,
    userid            text      NOT NULL,
    empname           text      NOT NULL,
    salary            integer
);

CREATE OR REPLACE FUNCTION process_emp_audit() RETURNS TRIGGER AS
 $emp_audit$
    BEGIN
        --

        -- emp#############emp_audit###
        -- ###################TG_OP###
        --
        IF (TG_OP = 'DELETE') THEN
            INSERT INTO emp_audit SELECT 'D', now(), current_user, OLD.*;
        ELSIF (TG_OP = 'UPDATE') THEN
            INSERT INTO emp_audit SELECT 'U', now(), current_user, NEW.*;
        ELSIF (TG_OP = 'INSERT') THEN
            INSERT INTO emp_audit SELECT 'I', now(), current_user, NEW.*;
        END IF;

        RETURN NULL; -- AFTER#################
    END;
$emp_audit$ LANGUAGE plpgsql;

CREATE TRIGGER emp_audit
AFTER INSERT OR UPDATE OR DELETE ON emp
    FOR EACH ROW EXECUTE FUNCTION process_emp_audit();
```

##
##PL/pgSQL#################

#41.5 ######PL/pgSQL########

##############################emp_audit#######################################
############################## ###################

```
CREATE TABLE emp (
    empname           text PRIMARY KEY,
    salary            integer
);

CREATE TABLE emp_audit(
    operation         char(1)   NOT NULL,
    userid            text      NOT NULL,
    empname           text      NOT NULL,
    salary            integer,
    stamp             timestamp NOT NULL
);

CREATE VIEW emp_view AS
    SELECT e.empname,
           e.salary,
           max(ea.stamp) AS last_updated
      FROM emp e
      LEFT JOIN emp_audit ea ON ea.empname = e.empname
     GROUP BY 1, 2;

CREATE OR REPLACE FUNCTION update_emp_view() RETURNS TRIGGER AS $$
    BEGIN
        --

        -- ######## emp ####
        -- emp_audit ######
        -- emp ########
        --
        IF (TG_OP = 'DELETE') THEN
            DELETE FROM emp WHERE empname = OLD.empname;
            IF NOT FOUND THEN RETURN NULL; END IF;

            OLD.last_updated = now();
            INSERT INTO emp_audit VALUES('D', current_user, OLD.*);
            RETURN OLD;
        ELSIF (TG_OP = 'UPDATE') THEN
            UPDATE emp SET salary = NEW.salary WHERE empname =
 OLD.empname;
            IF NOT FOUND THEN RETURN NULL; END IF;

            NEW.last_updated = now();
            INSERT INTO emp_audit VALUES('U', current_user, NEW.*);
            RETURN NEW;
        ELSIF (TG_OP = 'INSERT') THEN
            INSERT INTO emp VALUES(NEW.empname, NEW.salary);

            NEW.last_updated = now();
            INSERT INTO emp_audit VALUES('I', current_user, NEW.*);
            RETURN NEW;
        END IF;
    END;
$$ LANGUAGE plpgsql;
```

```
CREATE TRIGGER emp_audit
INSTEAD OF INSERT OR UPDATE OR DELETE ON emp_view
    FOR EACH ROW EXECUTE FUNCTION update_emp_view();
```

#########1########################### ##
################ ##
##################################PL/pgSQL########## 41.6######

#41.6 ###############PL/pgSQL#####

#############Ralph Kimball###*The Data Warehouse Toolkit#Grocery Store*###########

```
--

-- time dimension#sales fact######
--
CREATE TABLE time_dimension (
    time_key                    integer NOT NULL,
    day_of_week                 integer NOT NULL,
    day_of_month                integer NOT NULL,
    month                       integer NOT NULL,
    quarter                     integer NOT NULL,
    year                        integer NOT NULL
);
CREATE UNIQUE INDEX time_dimension_key ON time_dimension(time_key);

CREATE TABLE sales_fact (
    time_key                    integer NOT NULL,
    product_key                 integer NOT NULL,
    store_key                   integer NOT NULL,
    amount_sold                 numeric(12,2) NOT NULL,
    units_sold                  integer NOT NULL,
    amount_cost                 numeric(12,2) NOT NULL
);
CREATE INDEX sales_fact_time ON sales_fact(time_key);

--

-- sales by time########
--
CREATE TABLE sales_summary_bytime (
    time_key                    integer NOT NULL,
    amount_sold                 numeric(15,2) NOT NULL,
    units_sold                  numeric(12) NOT NULL,
    amount_cost                 numeric(15,2) NOT NULL
);
CREATE UNIQUE INDEX sales_summary_bytime_key ON
 sales_summary_bytime(time_key);

--

-- ##################################
--
CREATE OR REPLACE FUNCTION maint_sales_summary_bytime() RETURNS TRIGGER
AS $maint_sales_summary_bytime$
    DECLARE
        delta_time_key          integer;
        delta_amount_sold       numeric(15,2);
        delta_units_sold        numeric(12);
        delta_amount_cost       numeric(15,2);
    BEGIN
```

```
        -- ###########
        IF (TG_OP = 'DELETE') THEN

            delta_time_key = OLD.time_key;
            delta_amount_sold = -1 * OLD.amount_sold;
            delta_units_sold = -1 * OLD.units_sold;
            delta_amount_cost = -1 * OLD.amount_cost;

        ELSIF (TG_OP = 'UPDATE') THEN


            -- time_key#############
            -- ### + ###################
            -- ###############
            IF ( OLD.time_key != NEW.time_key) THEN
                RAISE EXCEPTION 'Update of time_key : % -> % not allowed',
                                                        OLD.time_key,
NEW.time_key;
            END IF;

            delta_time_key = OLD.time_key;
            delta_amount_sold = NEW.amount_sold - OLD.amount_sold;
            delta_units_sold = NEW.units_sold - OLD.units_sold;
            delta_amount_cost = NEW.amount_cost - OLD.amount_cost;

        ELSIF (TG_OP = 'INSERT') THEN

            delta_time_key = NEW.time_key;
            delta_amount_sold = NEW.amount_sold;
            delta_units_sold = NEW.units_sold;
            delta_amount_cost = NEW.amount_cost;

        END IF;



        -- ##########################
        <<insert_update>>
        LOOP
            UPDATE sales_summary_bytime
                SET amount_sold = amount_sold + delta_amount_sold,
                    units_sold = units_sold + delta_units_sold,
                    amount_cost = amount_cost + delta_amount_cost
                WHERE time_key = delta_time_key;

            EXIT insert_update WHEN found;

            BEGIN
                INSERT INTO sales_summary_bytime (
                            time_key,
                            amount_sold,
                            units_sold,
                            amount_cost)
                    VALUES (
                            delta_time_key,
                            delta_amount_sold,
                            delta_units_sold,
                            delta_amount_cost
                           );

                EXIT insert_update;

            EXCEPTION
                WHEN UNIQUE_VIOLATION THEN
```

```
                    -- ######
            END;
        END LOOP insert_update;

        RETURN NULL;

    END;
$maint_sales_summary_bytime$ LANGUAGE plpgsql;

CREATE TRIGGER maint_sales_summary_bytime
AFTER INSERT OR UPDATE OR DELETE ON sales_fact
    FOR EACH ROW EXECUTE FUNCTION maint_sales_summary_bytime();

INSERT INTO sales_fact VALUES(1,1,1,10,3,15);
INSERT INTO sales_fact VALUES(1,2,1,20,5,35);
INSERT INTO sales_fact VALUES(2,2,1,40,15,135);
INSERT INTO sales_fact VALUES(2,3,1,10,1,13);
SELECT * FROM sales_summary_bytime;
DELETE FROM sales_fact WHERE product_key = 1;
SELECT * FROM sales_summary_bytime;
UPDATE sales_fact SET units_sold = units_sold * 2;
SELECT * FROM sales_summary_bytime;
```

AFTER## CREATE TRIGGER########1#
###2## # 41.7########

#41.7

41.4###1################
REFERENCING#################
###########################
############ (#####3###############TG_OP################)

```
CREATE TABLE emp (
    empname           text NOT NULL,
    salary            integer
);

CREATE TABLE emp_audit(
    operation         char(1)   NOT NULL,
    stamp             timestamp NOT NULL,
    userid            text      NOT NULL,
    empname           text      NOT NULL,
    salary            integer
);

CREATE OR REPLACE FUNCTION process_emp_audit() RETURNS TRIGGER AS
 $emp_audit$
    BEGIN
        --

        -- emp################emp_audit######
        -- ################TG_OP####
        --
        IF (TG_OP = 'DELETE') THEN
            INSERT INTO emp_audit
                SELECT 'D', now(), current_user, o.* FROM old_table o;
        ELSIF (TG_OP = 'UPDATE') THEN
            INSERT INTO emp_audit
                SELECT 'U', now(), current_user, n.* FROM new_table n;
        ELSIF (TG_OP = 'INSERT') THEN
            INSERT INTO emp_audit
```

```
                SELECT 'I', now(), current_user, n.* FROM new_table n;
        END IF;

        RETURN NULL; -- ###AFTER##############
    END;
$emp_audit$ LANGUAGE plpgsql;

CREATE TRIGGER emp_audit_ins
    AFTER INSERT ON emp
    REFERENCING NEW TABLE AS new_table
    FOR EACH STATEMENT EXECUTE FUNCTION process_emp_audit();
CREATE TRIGGER emp_audit_upd
    AFTER UPDATE ON emp
    REFERENCING OLD TABLE AS old_table NEW TABLE AS new_table
    FOR EACH STATEMENT EXECUTE FUNCTION process_emp_audit();
CREATE TRIGGER emp_audit_del
    AFTER DELETE ON emp
    REFERENCING OLD TABLE AS old_table
    FOR EACH STATEMENT EXECUTE FUNCTION process_emp_audit();
```

41.10.2.

PL/pgSQL################## ######################################
#event_trigger######PostgreSQL#######

PL/pgSQL## ########

TG_EVENT text

   ##############

TG_TAG text

   ################

41.8#PL/pgSQL#####################

#41.8 PL/pgSQL#########

##############################NOTICE###########

```
CREATE OR REPLACE FUNCTION snitch() RETURNS event_trigger AS $$
BEGIN
    RAISE NOTICE 'snitch: % %', tg_event, tg_tag;
END;
$$ LANGUAGE plpgsql;

CREATE EVENT TRIGGER snitch ON ddl_command_start EXECUTE FUNCTION
 snitch();
```

41.11. PL/pgSQL###

#####PL/pgSQL#######################

41.11.1.

PL/pgSQL####SQL############################## #####PL/pgSQL################
######## ################################ ##################################
####

```
INSERT INTO foo (foo) VALUES (foo(foo));
```

######foo######################## #######foo########################## 2###### ################################## ####3################################## #### ####PL/pgSQL#############

##################SQL###
(###########41.5.4##################################)

######################
############### #################

```
INSERT INTO dest (col) SELECT foo + bar FROM src;
```

#####dest###src################ ###col#dest#############foo###bar#########
#######src#########

######PL/pgSQL#SQL####################################### ###################
############PL/pgSQL##

######################### ####################PL/pgSQL#################### ####
##########v_*something*###########v_######################

################################ ######src.foo########################### #####
##(41.2##)# #########

```
<<block>>
DECLARE
    foo int;
BEGIN
    foo := ...;
    INSERT INTO dest (col) SELECT block.foo + bar FROM src;
```

###block.foo#src#foo######################## FOUND#########################
######### #######################################

PL/pgSQL##   #######PL/pgSQL####
###############(#####PostgreSQL   9.0####PL/pgSQL###########)####################
(Oracle#################)#############

#######################plpgsql.variable_conflict#######
#error#use_variable#use_column##########(error###############)# #########
##PL/pgSQL## #############PL/pgSQL
###

#######################1############################

```
#variable_conflict error
#variable_conflict use_variable
#variable_conflict use_column
```

##################################plpgsql.variable_conflict########## ######
####

```
CREATE FUNCTION stamp_user(id int, comment text) RETURNS void AS $$
    #variable_conflict use_variable
    DECLARE
        curtime timestamp := now();
    BEGIN
        UPDATE users SET last_modified = curtime, comment = comment
          WHERE users.id = id;
    END;
$$ LANGUAGE plpgsql;
```

UPDATE########curtime#comment###id##users#############################
#############WHERE#####users.id####################### ###UPDATE#
########comment#################### #######users############### #####
#variable_conflict########################

```
CREATE FUNCTION stamp_user(id int, comment text) RETURNS void AS $$
    <<fn>>
    DECLARE
        curtime timestamp := now();
    BEGIN
        UPDATE users SET last_modified = fn.curtime, comment =
 stamp_user.comment
          WHERE users.id = stamp_user.id;
    END;
$$ LANGUAGE plpgsql;
```

#####EXECUTE################################ ####################41.5.4######
####################USING#########

###########SELECT#INSERT#UPDATE#DELETE#####(EXPLAIN#CREATE TABLE ...
AS SELECT####)#####1################## ###SQL################################
##EXECUTE#####
##

41.11.2.

PL/pgSQL###
###########PL/pgSQL###################SQL##SQL###############

 ###SQL####################PL/pgSQL######SPI######SPI_prepare##############
######################### ### ###########
########
##################################### ##
############

PL/pgSQL######SPI## ###############
###PL/
pgSQL############ ##################################SPI#####################
############################## #############PL/pgSQL###########################
###################PREPARE##########

#####PL/pgSQL################################PL/pgSQL########SQL############
######################### ####SQL### ######
######################PL/pgSQL#EXECUTE################################

#################################### ##
################ ### #####EXECUTE######
###############

######2######################PL/pgSQL############################## ###########
#################################### ######################### #############
##############key########################

#################
##################

#######2##############

```
CREATE FUNCTION logfunc1(logtxt text) RETURNS void AS $$
    BEGIN
        INSERT INTO logtable VALUES (logtxt, 'now');
```

```
    END;
$$ LANGUAGE plpgsql;
```

###

```
CREATE FUNCTION logfunc2(logtxt text) RETURNS void AS $$
    DECLARE
        curtime timestamp;
    BEGIN
        curtime := 'now';
        INSERT INTO logtable VALUES (logtxt, curtime);
    END;
$$ LANGUAGE plpgsql;
```

logfunc1######PostgreSQL########INSERT#######logtable###### #'now'#timestamp#### ########INSERT####### #'now'#timestamp#####logfunc1### ##### ##### now()###current_timestamp###

logfunc2######PostgreSQL#######'now'### #####now########text## ######### curtime#####PL/pgSQL#####textout#timestamp_in## #######timestamp######### ##### ####### ##### #####now()###

41.12. PL/pgSQL#####

PL/pgSQL#####1#####1########psql### ####### ##########CREATE OR REPLACE FUNCTION### ####### ##### #####

```
CREATE OR REPLACE FUNCTION testfunc(integer) RETURNS integer AS $$
          ....
$$ LANGUAGE plpgsql;
```

psql#####

```
\i filename.sql
```

#####SQL###

PL/pgSQL########1#####GUI#### ####### ##pgAdmin### ##### #######

41.12.1.

PL/pgSQL######CREATE FUNCTION### ### ###

################################ ##

1#####

########### ##########

```
CREATE FUNCTION foo() RETURNS integer AS '
          ....
' LANGUAGE plpgsql;
```

#####################

2#####

################ ##########

```
a_output := ''Blah'';
SELECT * FROM users WHERE f_name=''foobar'';
```

######################

```
a_output := 'Blah';
SELECT * FROM users WHERE f_name='foobar';
```

####PL/pgSQL################

4#####

########################### ##########

```
a_output := a_output || '' AND name LIKE ''''foobar'''' AND xyz''
```

###a_output######## AND name LIKE 'foobar' AND xyz###

#######################

```
a_output := a_output || $$ AND name LIKE 'foobar' AND xyz$$
```

###############$$###################

6#####

################################## ##########

```
a_output := a_output || '' AND name LIKE ''''foobar''''''
```

###a_output######## AND name LIKE 'foobar'###

#####################

```
a_output := a_output || $$ AND name LIKE 'foobar'$$
```

10#####

####### 2##################8##################################2######### ###
###################### 41.10################ ##########

```
a_output := a_output || '' if v_'' ||
    referrer_keys.kind || '' like '''''''''''
    || referrer_keys.key_string || '''''''''''
    then return ''''''  || referrer_keys.referrer_type
    || ''''''; end if;'';
```

a_output#############

if v_... like ''...'' then return ''...''; end if;

####################

a_output := a_output || $$ if v_$$ || referrer_keys.kind || $$ like '$$
|| referrer_keys.key_string || $$'
then return '$$ || referrer_keys.referrer_type
|| $$'; end if;$$;

####################a_output################

41.12.2.

################################PL/PgSQL####*checks*####### #################
#########WARNING###ERROR########### WARNING##############################
#########################

############plpgsql.extra_warnings#plpgsql.extra_errors####"all"######
#######

###############plpgsql.extra_warnings######plpgsql.extra_errors#######
####### ######################"none"###"all"####### ######"none"### #######
###############

shadowed_variables

#########################

strict_multi_assignment

PL/pgSQL##########SELECT INTO#######2####################### PL/pgSQL
##########NULL## ##############PL/
pgSQL#######################WARNING###ERROR############

too_many_rows

##############PL/pgSQL#INTO####################2################# INTO##
##1##/#########################

#######plpgsql.extra_warnings#shadowed_variables############

```
SET plpgsql.extra_warnings TO 'shadowed_variables';

CREATE FUNCTION foo(f1 int) RETURNS int AS $$
DECLARE
f1 int;
BEGIN
RETURN f1;
END;
$$ LANGUAGE plpgsql;
WARNING:  variable "f1" shadows a previously defined variable
LINE 3: f1 int;
        ^
CREATE FUNCTION
```

#######plpgsql.extra_warnings#strict_multi_assignment############

```
SET plpgsql.extra_warnings TO 'strict_multi_assignment';

CREATE OR REPLACE FUNCTION public.foo()
```

```
 RETURNS void
 LANGUAGE plpgsql
AS $$
DECLARE
  x int;
  y int;
BEGIN
  SELECT 1 INTO x, y;
  SELECT 1, 2 INTO x, y;
  SELECT 1, 2, 3 INTO x, y;
END;
$$;

SELECT foo();
WARNING:  number of source and target fields in assignment does not match
DETAIL:  strict_multi_assignment check of extra_warnings is active.
HINT:  Make sure the query returns the exact list of columns.
WARNING:  number of source and target fields in assignment does not match
DETAIL:  strict_multi_assignment check of extra_warnings is active.
HINT:  Make sure the query returns the exact list of columns.

 foo
-----

(1 row)
```

41.13. Oracle PL/SQL#####

####Oracle®##PostgreSQL##########################PostgreSQL#PL/pgSQL###Oracle# PL/SQL###############

PL/pgSQL######PL/SQL####### ## ############# PL/SQL##PL/pgSQL################################

- SQL###PL/SQL########## ##############################PL/pgSQL#################################### 41.11.1###### #plpgsql.variable_conflict = use_column###############PL/SQL############################## ##variable_conflict####################
- PostgreSQL########################### ## ###### #41.12.1###########
- ################## #####Oracle#########varchar2################SQL#### PostgreSQL########varchar####text######### ####number##numeric######### ###########################
- ####################################
- ############################## ############## ################################
- REVERSE######FOR############ PL/SQL##########################PL/pgSQL##### ################### ##################################### ################### ######### #41.6.5.5#########
- #######FOR######################### ######################PL/SQL############### ## ####################################
- ###########################

41.13.1.

41.9#######PL/SQL##PL/pgSQL############

#41.9 ######PL/SQL##PL/pgSQL####

###Oracle PL/SQL######

```
CREATE OR REPLACE FUNCTION cs_fmt_browser_version(v_name varchar2,
                                                  v_version varchar2)
RETURN varchar2 IS
BEGIN
    IF v_version IS NULL THEN
        RETURN v_name;
    END IF;
    RETURN v_name || '/' || v_version;
END;
/
show errors;
```

#########PL/pgSQL############

- ##varchar2##varchar###text############ #######varchar################### ##########text##############
- #########RETURN##################PostgreSQL##RETURNS###### ###IS#AS###### PL/pgSQL#################LANGUAGE#########
- PostgreSQL################## ############################ ###/#####Oracle#### #####
- PostgreSQL##show errors########## #########################

####PostgreSQL############################

```
CREATE OR REPLACE FUNCTION cs_fmt_browser_version(v_name varchar,
                                                  v_version varchar)
RETURNS varchar AS $$
BEGIN
    IF v_version IS NULL THEN
        RETURN v_name;
    END IF;
    RETURN v_name || '/' || v_version;
END;
$$ LANGUAGE plpgsql;
```

41.10##

#41.10 #########PL/SQL#PL/pgSQL###

########SELECT###############IF##################

###Oracle####

```
CREATE OR REPLACE PROCEDURE cs_update_referrer_type_proc IS
    CURSOR referrer_keys IS
        SELECT * FROM cs_referrer_keys
        ORDER BY try_order;
    func_cmd VARCHAR(4000);
BEGIN
    func_cmd := 'CREATE OR REPLACE FUNCTION cs_find_referrer_type(v_host
 IN VARCHAR2,
                 v_domain IN VARCHAR2, v_url IN VARCHAR2) RETURN VARCHAR2
 IS BEGIN';

    FOR referrer_key IN referrer_keys LOOP
```

```
        func_cmd := func_cmd ||
          ' IF v_' || referrer_key.kind
          || ' LIKE ''' || referrer_key.key_string
          || ''' THEN RETURN ''' || referrer_key.referrer_type
          || '''; END IF;';
    END LOOP;

    func_cmd := func_cmd || ' RETURN NULL; END;';

    EXECUTE IMMEDIATE func_cmd;
END;
/
show errors;
```

#####PostgreSQL##############

```
CREATE OR REPLACE PROCEDURE cs_update_referrer_type_proc() AS $func$
DECLARE
    referrer_keys CURSOR IS
        SELECT * FROM cs_referrer_keys
        ORDER BY try_order;
    func_body text;
    func_cmd text;
BEGIN
    func_body := 'BEGIN';

    FOR referrer_key IN referrer_keys LOOP
        func_body := func_body ||
          ' IF v_' || referrer_key.kind
          || ' LIKE ' || quote_literal(referrer_key.key_string)
          || ' THEN RETURN ' || quote_literal(referrer_key.referrer_type)
          || '; END IF;' ;
    END LOOP;

    func_body := func_body || ' RETURN NULL; END;';

    func_cmd :=
      'CREATE OR REPLACE FUNCTION cs_find_referrer_type(v_host varchar,
                                                        v_domain varchar,
                                                        v_url varchar)
        RETURNS varchar AS '
      || quote_literal(func_body)
      || ' LANGUAGE plpgsql;' ;

    EXECUTE func_cmd;
END;
$func$ LANGUAGE plpgsql;
```

##############quote_literal############################ ################ ########################### ###referrer_key.key_string################## ########### #referrer_key.kind#######host#domain#url################ #referrer_key.key_string######################## #####Oracle############ ## ###referrer_key.key_string###referrer_key.referrer_type############ ##############

41.11##OUT############################ PostgreSQL######instr############## ############## 41.13.3#############instr#PL/pgSQL##########

#41.11 #########OUT########PL/SQL######PL/pgSQL####

###Oracle PL/SQL#######URL##############################

###Oracle####

```
CREATE OR REPLACE PROCEDURE cs_parse_url(
    v_url IN VARCHAR2,

    v_host OUT VARCHAR2,  -- #########
    v_path OUT VARCHAR2,  -- #########
    v_query OUT VARCHAR2) -- #########
IS
    a_pos1 INTEGER;
    a_pos2 INTEGER;
BEGIN
    v_host := NULL;
    v_path := NULL;
    v_query := NULL;
    a_pos1 := instr(v_url, '//');

    IF a_pos1 = 0 THEN
        RETURN;
    END IF;
    a_pos2 := instr(v_url, '/', a_pos1 + 2);
    IF a_pos2 = 0 THEN
        v_host := substr(v_url, a_pos1 + 2);
        v_path := '/';
        RETURN;
    END IF;

    v_host := substr(v_url, a_pos1 + 2, a_pos2 - a_pos1 - 2);
    a_pos1 := instr(v_url, '?', a_pos2 + 1);

    IF a_pos1 = 0 THEN
        v_path := substr(v_url, a_pos2);
        RETURN;
    END IF;

    v_path := substr(v_url, a_pos2, a_pos1 - a_pos2);
    v_query := substr(v_url, a_pos1 + 1);
END;
/
show errors;
```

PL/pgSQL##################

```
CREATE OR REPLACE FUNCTION cs_parse_url(
    v_url IN VARCHAR,

    v_host OUT VARCHAR,  -- #########
    v_path OUT VARCHAR,  -- #########
    v_query OUT VARCHAR) -- #########
AS $$
DECLARE
    a_pos1 INTEGER;
    a_pos2 INTEGER;
BEGIN
    v_host := NULL;
    v_path := NULL;
    v_query := NULL;
    a_pos1 := instr(v_url, '//');

    IF a_pos1 = 0 THEN
        RETURN;
    END IF;
    a_pos2 := instr(v_url, '/', a_pos1 + 2);
    IF a_pos2 = 0 THEN
        v_host := substr(v_url, a_pos1 + 2);
```

```
        v_path := '/';
        RETURN;
    END IF;

    v_host := substr(v_url, a_pos1 + 2, a_pos2 - a_pos1 - 2);
    a_pos1 := instr(v_url, '?', a_pos2 + 1);

    IF a_pos1 = 0 THEN
        v_path := substr(v_url, a_pos2);
        RETURN;
    END IF;

    v_path := substr(v_url, a_pos2, a_pos1 - a_pos2);
    v_query := substr(v_url, a_pos1 + 1);
END;
$$ LANGUAGE plpgsql;
```

##################

```
SELECT * FROM cs_parse_url('http://foobar.com/query.cgi?baz');
```

41.12##Oracle##############################

#41.12 PL/SQL#######PL/pgSQL####

###Oracle####

```
CREATE OR REPLACE PROCEDURE cs_create_job(v_job_id IN INTEGER) IS
    a_running_job_count INTEGER;
BEGIN
    LOCK TABLE cs_jobs IN EXCLUSIVE MODE;

    SELECT count(*) INTO a_running_job_count FROM cs_jobs WHERE end_stamp
 IS NULL;

    IF a_running_job_count > 0 THEN

        COMMIT; -- ######
        raise_application_error(-20000,
                 'Unable to create a new job: a job is currently
 running.');
    END IF;

    DELETE FROM cs_active_job;
    INSERT INTO cs_active_job(job_id) VALUES (v_job_id);

    BEGIN
        INSERT INTO cs_jobs (job_id, start_stamp) VALUES (v_job_id,
 now());
    EXCEPTION

        WHEN dup_val_on_index THEN NULL; -- ###########
    END;
    COMMIT;
END;
/
show errors
```

##############PL/pgSQL######################

```
CREATE OR REPLACE PROCEDURE cs_create_job(v_job_id integer) AS $$
DECLARE
    a_running_job_count integer;
```

```
BEGIN
    LOCK TABLE cs_jobs IN EXCLUSIVE MODE;

    SELECT count(*) INTO a_running_job_count FROM cs_jobs WHERE end_stamp
 IS NULL;

    IF a_running_job_count > 0 THEN

        COMMIT; -- ######
        RAISE EXCEPTION 'Unable to create a new job: a job is currently
 running'; -- 1
    END IF;

    DELETE FROM cs_active_job;
    INSERT INTO cs_active_job(job_id) VALUES (v_job_id);

    BEGIN
        INSERT INTO cs_jobs (job_id, start_stamp) VALUES (v_job_id,
 now());
    EXCEPTION
        WHEN unique_violation THEN -- 2

            -- ###########
    END;
    COMMIT;
END;
$$ LANGUAGE plpgsql;
```

1 ###RAISE *exception_name*################RAISE###Oracle##############

2 PL/pgSQL##############Oracle###### ######################A########## #### ################### ##############SQLSTATE############

41.13.2.

#####Oracle PL/SQL###PostgreSQL#######################

41.13.2.1.

PL/pgSQL####EXCEPTION###########BEGIN################################### # ####Oracle######################

```
BEGIN
    SAVEPOINT s1;
    ... code here ...
EXCEPTION
    WHEN ... THEN
        ROLLBACK TO s1;
        ... code here ...
    WHEN ... THEN
        ROLLBACK TO s1;
        ... code here ...
END;
```

########SAVEPOINT#ROLLBACK TO#####Oracle################# # #SAVEPOINT#ROLLBACK TO############### #########SAVEPOINT#ROLLBACK TO### ################################

41.13.2.2. EXECUTE

PL/pgSQL#EXECUTE#PL/SQL######### ###41.5.4####### #quote_literal#quote_ident################### ##############EXECUTE ''SELECT * from $1'';###################

41.13.2.3. PL/pgSQL######

PostgreSQL#############2############## ######################################
######NULL#######NULL######## ###CREATE FUNCTION#########

```
##################CREATE FUNCTION###########


CREATE FUNCTION foo(...) RETURNS integer AS $$
...
$$ LANGUAGE plpgsql STRICT IMMUTABLE;
```

41.13.3.

```
#######################Oracle###instr###########



--
-- instr functions that mimic Oracle's counterpart
-- Syntax: instr(string1, string2 [, n [, m]])
-- where [] denotes optional parameters.
-- Oracle###########instr##
-- ##: instr(string1, string2 [, n [, m]])
-- ####[]###########
--
-- Search string1, beginning at the nth character, for the mth occurrence
-- of string2.  If n is negative, search backwards, starting at the
 abs(n)'th
-- character from the end of string1.
-- If n is not passed, assume 1 (search starts at first character).
-- If m is not passed, assume 1 (find first occurrence).
-- Returns starting index of string2 in string1, or 0 if string2 is not
 found.
-- string1##n###########m###string2######
-- n######string1######abs(n)####################
-- n############1######################
-- m############1#######################
-- string1##string2#######string2#########0######
--

CREATE FUNCTION instr(varchar, varchar) RETURNS integer AS $$
BEGIN
    RETURN instr($1, $2, 1);
END;
$$ LANGUAGE plpgsql STRICT IMMUTABLE;


CREATE FUNCTION instr(string varchar, string_to_search_for varchar,
                      beg_index integer)
RETURNS integer AS $$
DECLARE
    pos integer NOT NULL DEFAULT 0;
    temp_str varchar;
    beg integer;
    length integer;
    ss_length integer;
BEGIN
    IF beg_index > 0 THEN
        temp_str := substring(string FROM beg_index);
        pos := position(string_to_search_for IN temp_str);

        IF pos = 0 THEN
            RETURN 0;
```

```
        ELSE
            RETURN pos + beg_index - 1;
        END IF;
    ELSIF beg_index < 0 THEN
        ss_length := char_length(string_to_search_for);
        length := char_length(string);
        beg := length + 1 + beg_index;

        WHILE beg > 0 LOOP
            temp_str := substring(string FROM beg FOR ss_length);
            IF string_to_search_for = temp_str THEN
                RETURN beg;
            END IF;

            beg := beg - 1;
        END LOOP;

        RETURN 0;
    ELSE
        RETURN 0;
    END IF;
END;
$$ LANGUAGE plpgsql STRICT IMMUTABLE;


CREATE FUNCTION instr(string varchar, string_to_search_for varchar,
                      beg_index integer, occur_index integer)
RETURNS integer AS $$
DECLARE
    pos integer NOT NULL DEFAULT 0;
    occur_number integer NOT NULL DEFAULT 0;
    temp_str varchar;
    beg integer;
    i integer;
    length integer;
    ss_length integer;
BEGIN
    IF occur_index <= 0 THEN
        RAISE 'argument ''%'' is out of range', occur_index
          USING ERRCODE = '22003';
    END IF;

    IF beg_index > 0 THEN
        beg := beg_index - 1;
        FOR i IN 1..occur_index LOOP
            temp_str := substring(string FROM beg + 1);
            pos := position(string_to_search_for IN temp_str);
            IF pos = 0 THEN
                RETURN 0;
            END IF;
            beg := beg + pos;
        END LOOP;

        RETURN beg;
    ELSIF beg_index < 0 THEN
        ss_length := char_length(string_to_search_for);
        length := char_length(string);
        beg := length + 1 + beg_index;

        WHILE beg > 0 LOOP
            temp_str := substring(string FROM beg FOR ss_length);
            IF string_to_search_for = temp_str THEN
                occur_number := occur_number + 1;
                IF occur_number = occur_index THEN
```

```
                    RETURN beg;
                END IF;
            END IF;

            beg := beg - 1;
        END LOOP;

        RETURN 0;
    ELSE
        RETURN 0;
    END IF;
END;
$$ LANGUAGE plpgsql STRICT IMMUTABLE;
```

#42# PL/Tcl — Tcl#####

PL/Tcl###PostgreSQL#######################Tcl##[1]####PostgreSQL#################

42.1.

PL/Tcl###############C####################### ####Tcl##########################
##

####Tcl### ###Tcl#############
##SPI#######################elog()## C
##########PL/Tcl#######################PostgreSQL#############OS###############
##################

##################Tcl##################################

##########Tcl#############Tcl######Tcl#################### ########PL/TclU####
###Tcl####PL/Tcl########## #######Tcl############################## *PL/TclU######*
################################## ##
PL/TclU###
#####################

########Tcl#################PL/Tcl#PL/TclU##########################
##PostgreSQL####################### PL/Tcl###PL/TclU##########################
#####CREATE EXTENSION############# ####CREATE EXTENSION pltcl####CREATE
EXTENSION pltclu###

42.2. PL/Tcl#####

PL/Tcl####################CREATE FUNCTION############

```
CREATE FUNCTION funcname (argument-types) RETURNS return-type AS $$

    # PL/Tcl####
$$ LANGUAGE pltcl;
```

PL/TclU######pltclu####################

###########Tcl######## ################Tcl######1 … *n*################ ######
#return######Tcl########### #########Tcl#################

####2#############################

```
CREATE FUNCTION tcl_max(integer, integer) RETURNS integer AS $$
    if {$1 > $2} {return $1}
    return $2
$$ LANGUAGE pltcl STRICT;
```

STRICT########## ###############NULL########################## NULL########
###############NULL###############

###strict#################NULL##########$*n*############### #####NULL###########
##argisnull########### ##########NULL######NULL#####NULL######NULL######
##tcl_max################

[1] https://www.tcl.tk/

```
CREATE FUNCTION tcl_max(integer, integer) RETURNS integer AS $$
    if {[argisnull 1]} {
        if {[argisnull 2]} { return_null }
        return $2
    }
    if {[argisnull 2]} { return $1 }
    if {$1 > $2} {return $1}
    return $2
$$ LANGUAGE pltcl;
```

########NULL##PL/Tcl###########`return_null`########## ########################
######

########Tcl############## ################ #########NULL#################### ##
########

```
CREATE TABLE employee (
    name text,
    salary integer,
    age integer
);

CREATE FUNCTION overpaid(employee) RETURNS boolean AS $$
    if {200000.0 < $1(salary)} {
        return "t"
    }
    if {$1(age) < 30 && 100000.0 < $1(salary)} {
        return "t"
    }
    return "f"
$$ LANGUAGE pltcl;
```

PL/Tcl################### #######Tcl###################/################### ###
##############NULL######################## #######

```
CREATE FUNCTION square_cube(in int, out squared int, out cubed int) AS $$
    return [list squared [expr {$1 * $1}] cubed [expr {$1 * $1 * $1}]]
$$ LANGUAGE pltcl;
```

##############################

```
CREATE PROCEDURE tcl_triple(INOUT a integer, INOUT b integer) AS $$
    return [list a [expr {$1 * 3}] b [expr {$2 * 3}]]
$$ LANGUAGE pltcl;

CALL tcl_triple(5, 10);
```

> ###
>
> `array get` Tcl##
>
> ```
> CREATE FUNCTION raise_pay(employee, delta int) RETURNS employee AS $
> $
>  set 1(salary) [expr {$1(salary) + $2}]
>  return [array get 1]
> $$ LANGUAGE pltcl;
> ```

PL/Tcl################ ######Tcl#####`return_next`################ ###############
################/############# ###############

```
CREATE FUNCTION sequence(int, int) RETURNS SETOF int AS $$
    for {set i $1} {$i < $2} {incr i} {
        return_next $i
    }
$$ LANGUAGE pltcl;
```

#############

```
CREATE FUNCTION table_of_squares(int, int) RETURNS TABLE (x int, x2 int)
 AS $$
    for {set i $1} {$i < $2} {incr i} {
        return_next [list x $i x2 [expr {$i * $i}]]
    }
$$ LANGUAGE pltcl;
```

42.3. PL/Tcl##########

PL/Tcl#########################SELECT############################## #
##return#####return_next###
###############

42.4. PL/Tcl#############

###PL/Tcl
################################

#############PL/Tcl#####SQL#######################Tcl############ ##########
##################PL/Tcl#################### #########################Tcl####
#######SQL#################PL/Tcl################## #########SECURITY
DEFINER#####SET ROLE#########SQL#####################PL/Tcl#############
######################### ##SECURITY
DEFINER#################### ##
##

###############PL/TclU########PL/Tcl##################Tcl############## ####
PL/TclU####################### ####PL/TclU##################################
######################

PL/Tcl############################upvar#####################################
######## #########################GD###### ###########GD############## ########
################Tcl################# #GD################################## ##
################################

###spi_execp#####GD#########

42.5. PL/Tcl##############

#########Tcl### #########PL/Tcl###########
###################

spi_exec ?-count *n*? ?-array *name*? *command* ?*loop-body*?

> ###########SQL########### ###################### #########spi_exec#######
> ### ########SELECT#####
> ##############Tcl##########
>
> ######-count###spi_exec#########LIMIT##############*n*################
> *n*#0#########################-count#############

#####SELECT##########################Tcl######### -array##############
#######################
##.tupno###

#######SELECT#####*loop-body*###########################Tcl##############
#spi_exec######
#################### #########

```
spi_exec "SELECT count(*) AS cnt FROM pg_proc"
```

####$cnt Tcl####pg_proc################

loop-body##Tcl######
#*loop-body*#SELECT####################### ################Tcl##########
####### #########

```
spi_exec -array C "SELECT * FROM pg_class" {
    elog DEBUG "have table $C(relname)"
}
```

####pg_class################### ######Tcl################ #######
#continue#break##########

###########NULL################################

spi_prepare *query typelist*

########################## ############################

#############$1 …
$*n*####################### ##################Tcl###########################
##############*typelist*################

spi_prepare#########ID### ##ID###spi_execp############# ######
#spi_execp#########

spi_execp ?-count *n*? ?-array *name*? ?-nulls *string*? *queryid* ?*value-list*? ?*loop-body*?

spi_prepare#################### *queryid*#spi_prepare######ID### ###
################*value-list*########### ##################Tcl###### ####
########spi_prepare########################## ################*value-list*#########

-nulls#############'n'#############spi_execp########NULL####### ####
############*value-list*#################### ######################NULL#
##

##############################spi_execp#spi_exec######### -count#-array#*loop-body*##################

################PL/Tcl##########

```
CREATE FUNCTION t1_count(integer, integer) RETURNS integer AS $$
    if {![ info exists GD(plan) ]} {

        # #######################
        set GD(plan) [ spi_prepare \
                "SELECT count(*) AS cnt FROM t1 WHERE num >= \$1 AND
 num <= \$2" \
                [ list int4 int4 ] ]
```

```
        }
        spi_execp -count 1 $GD(plan) [ list $1 $2 ]
        return $cnt
    $$ LANGUAGE pltcl;
```

spi_prepare#################$*n*#########spi_prepare#####Tcl########### ######################

subtransaction *command*

command#####Tcl######SQL################## ################Tcl########## ######################## #############42.9#########

quote *string*

#################################### spi_exec#spi_prepare######SQL##### ## #######################

```
"SELECT '$val' AS ret"
```

####val Tcl###doesn't################ ###########################

```
SELECT 'doesn't' AS ret
```

#####spi_exec###spi_prepare################### ######################## ##############

```
SELECT 'doesn''t' AS ret
```

####PL/Tcl##################

```
"SELECT '[ quote $val ]' AS ret"
```

spi_execp###1###########SQL## #############

elog *level msg*

################### #########DEBUG#LOG#INFO#NOTICE#WARNING#ERROR## ##FATAL### ERROR############ #######Tcl########################## ### ######Tcl#error# ######## FATAL############################## #PL/Tcl############ ##################################### ############################ log_min_messages#client_min_messages## ################### ########19####42.8#########

42.6. PL/Tcl######

######PL/Tcl############ PostgreSQL##################trigger################# ###########

#####################################

$TG_name

CREATE TRIGGER########

$TG_relid

############################ID#

$TG_table_name

##########################

$TG_table_schema

#############################

$TG_relatts

#####################Tcl#### Tcl#lsearch################################1# ############ ####PostgreSQL################ ############################## ##############

$TG_when

##############BEFORE#AFTER###INSTEAD OF######

$TG_level

##############ROW###STATEMENT######

$TG_op

##############INSERT#UPDATE#DELETE####TRUNCATE#####

$NEW

INSERT/UPDATE#################DELETE############## ############## NULL ############### ####################

$OLD

UPDATE/DELETE################INSERT############## ############## NULL# ############## ####################

$args

CREATE TRIGGER##############Tcl#### ###########$1 ... $*n*################

############OK######SKIP########/##########1########### ####OK############ ##INSERT/UPDATE/DELETE########### SKIP################################## ## ###########PL/Tcl### #### ###########NULL###### ############$NEW############################BEFORE INSERT###UPDATE############INSERT RETURNING###UPDATE RETURNING##### #############INSTEAD OF INSERT###UPDATE########## #####BEFORE DELETE## #INSTEAD OF DELETE##################OK######################## ######### ############

###
######array get Tcl##############################

############1########

```
CREATE FUNCTION trigfunc_modcount() RETURNS trigger AS $$
    switch $TG_op {
        INSERT {
```

```
            set NEW($1) 0
        }
        UPDATE {
            set NEW($1) $OLD($1)
            incr NEW($1)
        }
        default {
            return OK
        }
    }
    return [array get NEW]
$$ LANGUAGE pltcl;

CREATE TABLE mytab (num integer, description text, modcnt integer);

CREATE TRIGGER trig_mytab_modcount BEFORE INSERT OR UPDATE ON mytab
    FOR EACH ROW EXECUTE FUNCTION trigfunc_modcount('modcnt');
```

############################ ################## ##############################
###

42.7. PL/Tcl##############

##########PL/Tcl############ PostgreSQL######################event_trigger###
#########################

####################################

$TG_event

   #############

$TG_tag

   ##############

################

####################NOTICE#############################

```
CREATE OR REPLACE FUNCTION tclsnitch() RETURNS event_trigger AS $$
  elog NOTICE "tclsnitch: $TG_event $TG_tag"
$$ LANGUAGE pltcl;

CREATE EVENT TRIGGER tcl_a_snitch ON ddl_command_start EXECUTE FUNCTION
 tclsnitch();
```

42.8. PL/Tcl######

PL/Tcl########PL/Tcl#######Tcl###################Tcl#error#####PL/Tcl#elog##
################################## #######Tcl#catch#######Tcl############## ###
########PL/Tcl##SQ############

###PL/Tcl#spi_exec#spi_prepare#spi_execp#########SQL#####Tcl##############
#######Tcl#catch############## ####PL/Tcl##################SQL###############
############################## ##############################SQL########

Tcl##Tcl##############################errorCode######## ######Tcl#######1####
PL/Tcl############
############1#####POSTGRES#2#####PostgreSQL##################################

```
######  #####SQLSTATE#condition####message############2####A#############
###                                                        ############
##detail#hint#context#schema#table#column#datatype#constraint#statement#cursor_po
##funcname######
```

PL/Tcl#errorCode## #########
###########

```
if {[catch { spi_exec $sql_command }]} {
    if {[lindex $::errorCode 0] == "POSTGRES"} {
        array set errorArray $::errorCode
        if {$errorArray(condition) == "undefined_table"} {
            # deal with missing table
        } else {
            # deal with some other type of SQL error
        }
    }
}
```

#######errorCode#######################

42.9. PL/Tcl##################

42.8###
################################## PL/Tcl#################################

2#######################

```
CREATE FUNCTION transfer_funds() RETURNS void AS $$
    if [catch {
        spi_exec "UPDATE accounts SET balance = balance - 100 WHERE
 account_name = 'joe'"
        spi_exec "UPDATE accounts SET balance = balance + 100 WHERE
 account_name = 'mary'"
    } errormsg] {
        set result [format "error transferring funds: %s" $errormsg]
    } else {
        set result "funds transferred successfully"
    }
    spi_exec "INSERT INTO operations (result) VALUES ('[quote $result]')"
$$ LANGUAGE pltcl;
```

#####UPDATE##UPDATE######### #####
##Joe##################Mary############### #########spi_exec##############
#####################################

PL/Tcl#########
###subtransaction############

```
CREATE FUNCTION transfer_funds2() RETURNS void AS $$
    if [catch {
        subtransaction {
            spi_exec "UPDATE accounts SET balance = balance - 100 WHERE
 account_name = 'joe'"
            spi_exec "UPDATE accounts SET balance = balance + 100 WHERE
 account_name = 'mary'"
        }
    } errormsg] {
        set result [format "error transferring funds: %s" $errormsg]
    } else {
```

```
        set result "funds transferred successfully"
    }
    spi_exec "INSERT INTO operations (result) VALUES ('[quote $result]')"
$$ LANGUAGE pltcl;
```

#########catch################# ################################operations## ################# subtransaction############### ############################ ############################

#######################Tcl## ### ##subtransaction###############Tcl############################ ####Tcl###### ##############return######################

42.10.

##################################DO##################################### ### #################commit######### ######################rollback######## ## #SQL#####COMMIT#ROLLBACK#spi_exec#################################### ######## ##

##########

```
CREATE PROCEDURE transaction_test1()
LANGUAGE pltcl
AS $$
for {set i 0} {$i < 10} {incr i} {
    spi_exec "INSERT INTO test1 (a) VALUES ($i)"
    if {$i % 2 == 0} {
        commit
    } else {
        rollback
    }
}
$$;

CALL transaction_test1();
```

##

42.11. PL/Tcl###

######PL/Tcl###################

pltcl.start_proc(string)

#######################PL/Tcl#######Tcl###########################PL/Tcl####### ####################### ###########Tcl############Tcl################### ##### ############PL/Tcl#################PL/Tcl#################################### ###########Tcl#############

##########pltcl###############SECURITY DEFINER################## ####### ### ################################## #####

###Tcl###Tcl#####################

###Tcl###############

`pltclu.start_proc(string)`

########PL/TclU###########`pltcl.start_proc`########### ########`pltclu`### ##############

42.12. Tcl#######

PostgreSQL## ####Tcl######### ########## PL/Tcl##############`pg_proc`#####################ID#########Tcl#### ##################### ##########################PostgreSQL#######Tcl############ PL/Tcl##

#43# PL/Perl — Perl#####

PL/Perl#Perl#########[1]#####PostgreSQL##################################

PL/Perl###########################Perl################################## ####### ###PL/pgSQL#######################Perl###################

PL/Perl####################CREATE EXTENSION plperl#########

> ###
>
> ###template1##

> ##
>
> ###########################PL/Perl################### ##########17######### ### ############################PL/Perl###############

43.1. PL/Perl#####

PL/Perl###################CREATE FUNCTION############

```
CREATE FUNCTION funcname (argument-types)
RETURNS return-type

-- ###########
AS $$

    # PL/Perl##########
$$ LANGUAGE plperl;
```

########Perl####### ###PL/Perl############Perl################# PL/Perl############## ########################### ##

PL/Perl#########Perl####################

##PL/Perl#DO#########################

```
DO $$
    # PL/Perl code
$$ LANGUAGE plperl;
```

################## ##################### ###############

> ##
>
> Perl### PL/Perl######## ### ############################## ######### ###perldiag#########Variable "%s" will not stay shared##

[1] https://www.perl.org

> #Variable "%s" is not available######### ###########perl nested named subroutine##########

CREATE FUNCTION#################################### ################ #4.1.2.4################## #########E''############################'####### ###\##################4.1.2.1#####

#######Perl############### ###@_##########return##########################

####2############################

```
CREATE FUNCTION perl_max (integer, integer) RETURNS integer AS $$
    if ($_[0] > $_[1]) { return $_[0]; }
    return $_[1];
$$ LANGUAGE plperl;
```

> ##
>
> PL/Perl#########################UTF-8############UTF-8################ ####

SQL#NULL#################Perl################ ########NULL############### ############################## STRICT###########PostgreSQL################## ## NULL#######################NULL############### ######################## ###### ####perl_max######NULL######NULL#####NULL#####NULL############## ##

```
CREATE FUNCTION perl_max (integer, integer) RETURNS integer AS $$
    my ($x, $y) = @_;
    if (not defined $x) {
        return undef if not defined $y;
        return $y;
    }
    return $x if not defined $y;
    return $x if $x > $y;
    return $y;
$$ LANGUAGE plperl;
```

#######PL/Perl####SQL#NULL######################## ######################### #####

###########################PostgreSQL####################### ##############Perl# ################################ ################Perl####################### ## ##decode_bytea###bytea#################################

####PostgreSQL############################## ####bytea#################### #encode_bytea############

#######1####### ###########bool###########Perl############'t'###'f'####### ##### Perl#'f'######################## ####(CREATE TRANSFORM########)###### ############ ######bool_plperl########### #################

```
CREATE EXTENSION bool_plperl;  -- PL/PerlU#####bool_plperlu
```

##bool############PL/Perl#####TRANSFORM####################

```
CREATE FUNCTION perl_and(bool, bool) RETURNS bool
TRANSFORM FOR TYPE bool
AS $$
  my ($a, $b) = @_;
  return $a && $b;
$$ LANGUAGE plperl;
```

###########bool###Perl###1#################### #######bool###Perl########### ################## ############################SPI########(43.3.1)#########

Perl##PostgreSQL####Perl################## #########

```
CREATE OR REPLACE function returns_array()
RETURNS text[][] AS $$
    return [['a"b','c,d'],['e\\f','g']];
$$ LANGUAGE plperl;

select returns_array();
```

Perl#PostgreSQL####bless###PostgreSQL::InServer::ARRAY############## 9.1##### PostgreSQL######Perl## ###### ####

```
CREATE OR REPLACE FUNCTION concat_array_elements(text[]) RETURNS TEXT AS $
$
    my $arg = shift;
    my $result = "";
    return undef if (!defined $arg);


    # #########
    for (@$arg) {
        $result .= $_;
    }


    # #########
    $result .= $arg;

    return $result;
$$ LANGUAGE plperl;

SELECT concat_array_elements(ARRAY['PL','/','Perl']);
```

##
Perl##

########################## ################## #########

```
CREATE TABLE employee (
    name text,
    basesalary integer,
    bonus integer
);

CREATE FUNCTION empcomp(employee) RETURNS integer AS $$
    my ($emp) = @_;
```

```
    return $emp->{basesalary} + $emp->{bonus};
$$ LANGUAGE plperl;

SELECT name, empcomp(employee.*) FROM employee;

##########################PL/Perl#################### ##########


CREATE TYPE testrowperl AS (f1 integer, f2 text, f3 text);

CREATE OR REPLACE FUNCTION perl_row() RETURNS testrowperl AS $$
    return {f2 => 'hello', f1 => 1, f3 => 'world'};
$$ LANGUAGE plperl;

SELECT * FROM perl_row();

################################NULL##########

################################


CREATE PROCEDURE perl_triple(INOUT a integer, INOUT b integer) AS $$
    my ($a, $b) = @_;
    return {a => $a * 3, b => $b * 3};
$$ LANGUAGE plperl;

CALL perl_triple(5, 10);

###PL/Perl###########################  ########################################
##1##1############  #####return_next##################  ###return_next##
##return###return undef##########################


CREATE OR REPLACE FUNCTION perl_set_int(int)
RETURNS SETOF INTEGER AS $$
    foreach (0..$_[0]) {
        return_next($_);
    }
    return undef;
$$ LANGUAGE plperl;

SELECT * FROM perl_set_int(5);

CREATE OR REPLACE FUNCTION perl_set()
RETURNS SETOF testrowperl AS $$
    return_next({ f1 => 1, f2 => 'Hello', f3 => 'World' });
    return_next({ f1 => 2, f2 => 'Hello', f3 => 'PostgreSQL' });
    return_next({ f1 => 3, f2 => 'Hello', f3 => 'PL/Perl' });
    return undef;
$$ LANGUAGE plperl;

########################################################################## #######
##############################


CREATE OR REPLACE FUNCTION perl_set_int(int) RETURNS SETOF INTEGER AS $$
    return [0..$_[0]];
$$ LANGUAGE plperl;

SELECT * FROM perl_set_int(5);

CREATE OR REPLACE FUNCTION perl_set() RETURNS SETOF testrowperl AS $$
    return [
        { f1 => 1, f2 => 'Hello', f3 => 'World' },
        { f1 => 2, f2 => 'Hello', f3 => 'PostgreSQL' },
```

```
        { f1 => 3, f2 => 'Hello', f3 => 'PL/Perl' }
    ];
$$ LANGUAGE plperl;

SELECT * FROM perl_set();
```

#####strict########################## ###############SET plperl.use_strict############ ############PL/Perl########################## ################ ################postgresql.conf#####plperl.use_strict# #######

```
use strict;
```

#################

###Perl#######5.10.0######use#feature###########

43.2. PL/Perl#########

PL/Perl#################(SELECT############)################## ###return## #return_next##

#######################bool######################## #############PostgreSQL### #########

43.3.

43.3.1. PL/Perl##############

Perl################################

spi_exec_query(*query* [, *limit*])

spi_exec_query#SQL## *limit*#### ##############spi_exec_query#######LIMIT################*limit*######## *limit*##########################

#SELECT############

```
$rv = spi_exec_query('SELECT * FROM my_table', 5);
```

###my_table######5######### my_table#my_column###########$i############# #########

```
$foo = $rv->{rows}[$i]->{my_column};
```

SELECT##############################

```
$nrows = $rv->{processed}
```

####################

```
$query = "INSERT INTO my_table VALUES (1, 'test')";
$rv = spi_exec_query($query);

######################SPI_OK_INSERT###############


$res = $rv->{status};

######################


$nrows = $rv->{processed};

#############


CREATE TABLE test (
    i int,
    v varchar
);

INSERT INTO test (i, v) VALUES (1, 'first line');
INSERT INTO test (i, v) VALUES (2, 'second line');
INSERT INTO test (i, v) VALUES (3, 'third line');
INSERT INTO test (i, v) VALUES (4, 'immortal');

CREATE OR REPLACE FUNCTION test_munge() RETURNS SETOF test AS $$
    my $rv = spi_exec_query('select i, v from test;');
    my $status = $rv->{status};
    my $nrows = $rv->{processed};
    foreach my $rn (0 .. $nrows - 1) {
        my $row = $rv->{rows}[$rn];
        $row->{i} += 200 if defined($row->{i});
        $row->{v} =~ tr/A-Za-z/a-zA-Z/ if (defined($row->{v}));
        return_next($row);
    }
    return undef;
$$ LANGUAGE plperl;

SELECT * FROM test_munge();
```

spi_query(*command*)
spi_fetchrow(*cursor*)
spi_cursor_close(*cursor*)

spi_query###spi_fetchrow##
spi_fetchrow#spi_query##############

```
CREATE TYPE foo_type AS (the_num INTEGER, the_text TEXT);

CREATE OR REPLACE FUNCTION lotsa_md5 (INTEGER) RETURNS SETOF foo_type
 AS $$
    use Digest::MD5 qw(md5_hex);
    my $file = '/usr/share/dict/words';
    my $t = localtime;
    elog(NOTICE, "opening file $file at $t" );
    open my $fh, '<', $file # ooh, it's a file access!
        or elog(ERROR, "cannot open $file for reading: $!");
    my @words = <$fh>;
    close $fh;
    $t = localtime;
    elog(NOTICE, "closed file $file at $t");
    chomp(@words);
    my $row;
```

```
    my $sth = spi_query("SELECT * FROM generate_series(1,$_[0]) AS
 b(a)");
    while (defined ($row = spi_fetchrow($sth))) {
        return_next({
            the_num => $row->{a},
            the_text => md5_hex($words[rand @words])
        });
    }
    return;
$$ LANGUAGE plperlu;

SELECT * from lotsa_md5(500);
```

##spi_fetchrow#################undef################
spi_fetchrow#undef####spi_query################### ##################
##spi_cursor_close################## ########################

spi_prepare(*command*, *argument types*)
spi_query_prepared(*plan*, *arguments*)
spi_exec_prepared(*plan* [, *attributes*], *arguments*)
spi_freeplan(*plan*)

spi_prepare#spi_query_prepared#spi_exec_prepared#spi_freeplan#### ##################### spi_prepare#############($1#$2##)################## #############

```
$plan = spi_prepare('SELECT * FROM test WHERE id > $1 AND name = $2',
                                                        'INTEGER',
 'TEXT');
```

spi_prepare######################spi_exec_query################ #spi_exec_prepared#spi_query##############spi_query_prepared(##### ####spi_fetchrow#########)################################ spi_exec_prepared######################## ############################## #####limit### limit#########################

##############1## #########
##spi_freeplan####################

```
CREATE OR REPLACE FUNCTION init() RETURNS VOID AS $$
        $_SHARED{my_plan} = spi_prepare('SELECT (now() + $1)::date AS
 now',
                                        'INTERVAL');
$$ LANGUAGE plperl;

CREATE OR REPLACE FUNCTION add_time( INTERVAL ) RETURNS TEXT AS $$
        return spi_exec_prepared(
                $_SHARED{my_plan},
                $_[0]
        )->{rows}->[0]->{now};
$$ LANGUAGE plperl;

CREATE OR REPLACE FUNCTION done() RETURNS VOID AS $$
        spi_freeplan( $_SHARED{my_plan});
        undef $_SHARED{my_plan};
$$ LANGUAGE plperl;

SELECT init();
SELECT add_time('1 day'), add_time('2 days'), add_time('3 days');
SELECT done();

  add_time  |  add_time  |  add_time
```

```
------------+------------+------------
 2005-12-10 | 2005-12-11 | 2005-12-12
```

spi_prepare#########$1#$2#$3###################### ######################
##############################

#####spi_exec_prepared##########################

```
CREATE TABLE hosts AS SELECT id, ('192.168.1.'||id)::inet AS address
                      FROM generate_series(1,3) AS id;

CREATE OR REPLACE FUNCTION init_hosts_query() RETURNS VOID AS $$
        $_SHARED{plan} = spi_prepare('SELECT * FROM hosts
                                      WHERE address << $1', 'inet');
$$ LANGUAGE plperl;

CREATE OR REPLACE FUNCTION query_hosts(inet) RETURNS SETOF hosts AS $$
        return spi_exec_prepared(
                $_SHARED{plan},
                {limit => 2},
                $_[0]
        )->{rows};
$$ LANGUAGE plperl;

CREATE OR REPLACE FUNCTION release_hosts_query() RETURNS VOID AS $$
        spi_freeplan($_SHARED{plan});
        undef $_SHARED{plan};
$$ LANGUAGE plperl;

SELECT init_hosts_query();
SELECT query_hosts('192.168.1.0/30');
SELECT release_hosts_query();

    query_hosts
-----------------
 (1,192.168.1.1)
 (2,192.168.1.2)
(2 rows)
```

spi_commit()
spi_rollback()

############################## ####################################DO########
############## #SQL#####COMMIT#ROLLBACK#spi_exec_query###############
################################ ##
###############

##########

```
CREATE PROCEDURE transaction_test1()
LANGUAGE plperl
AS $$
foreach my $i (0..9) {
    spi_exec_query("INSERT INTO test1 (a) VALUES ($i)");
    if ($i % 2 == 0) {
        spi_commit();
    } else {
        spi_rollback();
    }
}
$$;
```

```
CALL transaction_test1();
```

43.3.2. PL/Perl##########

`elog(level, msg)`

################### ##########DEBUG#LOG#INFO#NOTICE#WARNING####ERROR# ## ERROR############ #####Perl## ############################### #####Perl#die######### ################### ####### log_min_messages#client_min_messages################################ ############################ ####19##########

`quote_literal(string)`

##########SQL#################################### ##################### ########## quote_literal####undef###undef############### ###undef########## ##quote_nullable#########

`quote_nullable(string)`

##########SQL##################################### ###undef############## #"NULL"###### ################################

`quote_ident(string)`

##########SQL################################## #####(##################### ##################)############ ####################

`decode_bytea(string)`

################################## ###bytea#############

`encode_bytea(string)`

##########################bytea######

`encode_array_literal(array)`
`encode_array_literal(array, delimiter)`

################################(8.15.2##)# ######################### ######### #################undef#########", "###

`encode_typed_literal(value, typename)`

Perl###2################################### #####################

`encode_array_constructor(array)`

##############################(4.2.12##)# #####quote_nullable############## ############quote_nullable######################

`looks_like_number(string)`

############Perl############################## ###undef###undef###### ######### ######### Inf###Infinity#########

`is_array_ref(argument)`

################################ref#ARRAY## #PostgreSQL::InServer::ARRAY########## ############

43.4. PL/Perl###########

##%_SHARED################

#####################

```
CREATE OR REPLACE FUNCTION set_var(name text, val text) RETURNS text AS $$
    if ($_SHARED{$_[0]} = $_[1]) {
        return 'ok';
    } else {
        return "cannot set shared variable $_[0] to $_[1]";
    }
$$ LANGUAGE plperl;

CREATE OR REPLACE FUNCTION get_var(name text) RETURNS text AS $$
    return $_SHARED{$_[0]};
$$ LANGUAGE plperl;

SELECT set_var('sample', 'Hello, PL/Perl!  How''s tricks?');
SELECT get_var('sample');
```

#######################

```
CREATE OR REPLACE FUNCTION myfuncs() RETURNS void AS $$
    $_SHARED{myquote} = sub {
        my $arg = shift;
        $arg =~ s/(['\\])/\\$1/g;
        return "'$arg'";
    };
$$ LANGUAGE plperl;


SELECT myfuncs(); /* ###### */


/* ############### */

CREATE OR REPLACE FUNCTION use_quote(TEXT) RETURNS text AS $$
    my $text_to_quote = shift;
    my $qfunc = $_SHARED{myquote};
    return &$qfunc($text_to_quote);
$$ LANGUAGE plperl;
```

#############return $_SHARED{myquote}->($_[0]);###1#################

#############PL/Perl####SQL#######################Perl############# ########## ##################PL/Perl#################### ##################%_SHARED###### ######### #########SQL###################PL/Perl#####%_SHARED######## ######## #####SQL##########SECURITY DEFINER###SET ROLE#######################PL/Perl###%_SHARED################################ ############################## ##########SECURITY DEFINER######################### ######################## #########################

43.5. #####PL/Perl#########PL/Perl

###PL/Perl#plperl################################## ####################Perl## ############ ######################### ###############require#use######### ####### C#########Perl#######################################OS################## ## ######################################

###

```
CREATE FUNCTION badfunc() RETURNS integer AS $$
    my $tmpfile = "/tmp/badfile";
    open my $fh, '>', $tmpfile
        or elog(ERROR, qq{could not open the file "$tmpfile": $!});
    print $fh "Testing writing to a file\n";
    close $fh or elog(ERROR, qq{could not close the file "$tmpfile": $!});
    return 1;
$$ LANGUAGE plperl;
```

##

#####Perl################# ####Perl ###################### ##############PL/Perl### ###########PL/PerlU######################## ########Perl############### ######### ######plperlu################PL/Perl###########

PL/PerlU## ############################## ################ ##

#################plperlu#########################

#########plperl####plperlu#########Perl################################### ## ########################

> ##
>
> PL/Perl###SQL########Perl############################PL/PerlU##########Perl #######PL/Perl##################################### #####PL/PerlU########### ########### ###PL/Perl###PL/PerlU#################

> ##
>
> Perl###########usemultiplicity###useithreads######################### ###################### ##################usemultiplicity##### ### perlembed################## PL/Perl##############Perl######################## ##Perl################## ###############PL/PerlU##############SQL######## ##PL/Perl#####################

43.6. PL/Perl###

PL/Perl###################### ########$_TD#################################### $_TD################################### ###$_TD######################

```
$_TD->{new}{foo}

   NEW#foo###

$_TD->{old}{foo}

   OLD#foo###

$_TD->{name}

   #############
```

```
$_TD->{event}

   ######## INSERT#UPDATE#DELETE#TRUNCATE#####UNKNOWN#

$_TD->{when}

   ############## BEFORE#AFTER#INSTEAD OF####UNKNOWN#

$_TD->{level}

   ######## ROW#STATEMENT#####UNKNOWN#

$_TD->{relid}

   ############OID#

$_TD->{table_name}

   ###############

$_TD->{relname}

   ############### ############################### ####$_TD->{table_name}########
   ##

$_TD->{table_schema}

   ########################

$_TD->{argc}

   ###########

@{$_TD->{args}}

   ######### $_TD->{argc}#0###########

##########################

return;

   #########

"SKIP"

   ##########

"MODIFY"

   #########NEW###############

################################


CREATE TABLE test (
    i int,
    v varchar
);

CREATE OR REPLACE FUNCTION valid_id() RETURNS trigger AS $$
    if (($_TD->{new}{i} >= 100) || ($_TD->{new}{i} <= 0)) {

        return "SKIP";    # INSERT/UPDATE###########
```

```
    } elsif ($_TD->{new}{v} ne "immortal") {
        $_TD->{new}{v} .= "(modified by trigger)";

        return "MODIFY";  # ######INSERT/UPDATE###########
    } else {

        return;           # INSERT/UPDATE###########
    }
$$ LANGUAGE plperl;

CREATE TRIGGER test_valid_id_trig
    BEFORE INSERT OR UPDATE ON test
    FOR EACH ROW EXECUTE FUNCTION valid_id();
```

43.7. PL/Perl#######

PL/Perl########################## ###########$_TD############################## ######## $_TD## ###$_TD################## #####

$_TD->{event}

   #############

$_TD->{tag}

   #############

################

#################################

```
CREATE OR REPLACE FUNCTION perlsnitch() RETURNS event_trigger AS $$
  elog(NOTICE, "perlsnitch: " . $_TD->{event} . " " . $_TD->{tag} . " ");
$$ LANGUAGE plperl;

CREATE EVENT TRIGGER perl_a_snitch
    ON ddl_command_start
    EXECUTE FUNCTION perlsnitch();
```

43.8. PL/Perl###

43.8.1.

####PL/Perl##################

plperl.on_init(string)

   Perl################plperl###plperlu######################perl########## ## ############SPI########### ## ####################################

   ##Perl################ #############on_init################ ##########

```
plperl.on_init = 'require "plperlinit.pl"'
plperl.on_init = 'use lib "/my/app"; use MyApp::PgInit;'
```

   plperl.on_init##########################plperl############ ########### ################ ##################################

```
DO 'elog(WARNING, join ", ", sort keys %INC)' LANGUAGE plperl;
```

plperl######shared_preload_libraries#############postmaster######## ### ##postmaster########################## ################## ##plperl.on_init######Perl######postmaster######################## ### ######################## Perl#######PL/PerlU###PL/Perl#########SQL#####PL/Perl######################### ############ #####################Perl#############plperl.on_init####### ##### ###postmaster##########Perl######################Windows############### ###############

#######postgresql.conf##########################

plperl.on_plperl_init (string)
plperl.on_plperlu_init (string)

##############plperl###plperlu##Perl#################Perl######### ###### ##########PL/Perl###PL/PerlU##################################SQL####PL/Perl## ####plperl.on_init########## ## ############SPI########## plperl.on_plperl_init##Perl################ ########### ####Perl######################

Perl###################

############################ ## ######Perl################

plperl.use_strict (boolean)

#########PL/Perl########strict############ ############################## ####

43.8.2.

###########PL/Perl###### ############

- PL/Perl####################
- SPI###############
- spi_exec_query## ### #######spi_query/spi_fetchrow#####################

  ################return####PostgreSQL################# ######### #return_next######################

- ##END########### ################# #### ##################### ####################

#44# PL/Python — Python#####

PL/Python#########PostgreSQL##########Python##[1]#######

PL/Python####################CREATE EXTENSION plpython3u#########

> ###
>
> ###template1##

PL/Python## ######plpython3u# ############ Python#########################plpython################## ###### ##PL/Python######################### ##################### #########plpython3u#########################

> ##
>
> ##############################PL/Python######################## ########### ################ ############################PL/Python###############

44.1. PL/Python##

PL/Python############CREATE FUNCTION#########

```
CREATE FUNCTION funcname (argument-list)
  RETURNS return-type
AS $$
  # PL/Python function body
$$ LANGUAGE plpython3u;
```

########Python######## ##############args[]############## ################Python ########### ###################### ####Python##########return###yield###### ############## ###########Python######None##### PL/Python#Python#None#SQL# NULL####### #########Python#########None################return############## ###return################### ##############

#####2##################################

```
CREATE FUNCTION pymax (a integer, b integer)
  RETURNS integer
AS $$
  if a > b:
    return a
  return b
$$ LANGUAGE plpython3u;
```

###############Python#####Python########## #################

```
def __plpython_procedure_pymax_23456():
```

[1] https://www.python.org

```
  if a > b:
    return a
  return b
```

####23456#PostgreSQL##############OID###

################### Python## #############

```
CREATE FUNCTION pystrip(x text)
  RETURNS text
AS $$
  x = x.strip()  # error
  return x
$$ LANGUAGE plpython3u;
```

x######x##################################x#PL/Python##x######### global####################

```
CREATE FUNCTION pystrip(x text)
  RETURNS text
AS $$
  global x
  x = x.strip()  # ok now
  return x
$$ LANGUAGE plpython3u;
```

####PL/Python############################ ##########################

44.2.

########PL/Python####PostgreSQL#Python######################### ################## ######

44.2.1.

PL/Python####################PostgreSQL########Python#########

- PostgreSQL#boolean#Python#bool########
- PostgreSQL#smallint#int#bigint###oid#Python#int########
- PostgreSQL#real###double#Python#float########
- PostgreSQL#numeric#Python#Decimal######## ########cdecimal############## ## ###############decimal.Decimal######## cdecimal#decimal####### ### Python 3.3###cdecimal#decimal#################################
- PostgreSQL#bytea##Python#bytes########
- PostgreSQL######################Python#str#(####Python#######Unicode#)#######
- ###################

PL/Python######################PostgreSQL####################

- PostgreSQL#######boolean########*Python*################### ####0######### 'f'# ################

- PostgreSQL#######bytea########Python###########Python#bytes###########bytea#
  #######

- ####PostgreSQL##############Python######str##################PostgreSQL######
  ######### (Python###float##################str#####repr###############)

  ####PostgreSQL###########PostgreSQL################

- ###################

#####PostgreSQL###################Python################################ ######
##########

44.2.2. Null#None

SQL#NULL################Python##None##### ####44.1#####pymax########NULL
################## #####STRICT#####PostgreSQL##NULL#########################
NULL########################### ########NULL##############

```
CREATE FUNCTION pymax (a integer, b integer)
  RETURNS integer
AS $$
  if (a is None) or (b is None):
    return None
  if a > b:
    return a
  return b
$$ LANGUAGE plpython3u;
```

#########PL/Python####SQL NULL#######None############ #######################
###########

44.2.3.

SQL####Python#######PL/Python####### PL/Python######SQL###########Python########
##

```
CREATE FUNCTION return_arr()
  RETURNS int[]
AS $$
return [1, 2, 3, 4, 5]
$$ LANGUAGE plpython3u;

SELECT return_arr();
 return_arr
-------------
 {1,2,3,4,5}
(1 row)
```

######PL/Python#####Python############ ####2############## PL/Python######SQL###
##################################### ####

```
CREATE FUNCTION test_type_conversion_array_int4(x int4[]) RETURNS int4[]
 AS $$
plpy.info(x, type(x))
return x
$$ LANGUAGE plpython3u;

SELECT * FROM test_type_conversion_array_int4(ARRAY[[1,2,3],[4,5,6]]);
```

```
INFO:  ([[1, 2, 3], [4, 5, 6]], <type 'list'>)
 test_type_conversion_array_int4
---------------------------------
 {{1,2,3},{4,5,6}}
(1 row)
```

#########Python########PostgreSQL#####9.6#################################### ####### ####################1############# ################################### ###########

Python########################## #######################Python################

```
CREATE FUNCTION return_str_arr()
  RETURNS varchar[]
AS $$
return "hello"
$$ LANGUAGE plpython3u;

SELECT return_str_arr();
 return_str_arr
----------------
 {h,e,l,l,o}
(1 row)
```

44.2.4.

#######Python############ ################# ##########NULL##########None###### #### ##########

```
CREATE TABLE employee (
  name text,
  salary integer,
  age integer
);

CREATE FUNCTION overpaid (e employee)
  RETURNS boolean
AS $$
  if e["salary"] > 200000:
    return True
  if (e["age"] < 30) and (e["salary"] > 100000):
    return True
  return False
$$ LANGUAGE plpython3u;
```

Python######################## ###############

```
CREATE TYPE named_value AS (
  name   text,
  value  integer
);
```

###################

##################################set####

```
    ################################################  0#######################
    ##1################ ##########

    CREATE FUNCTION make_pair (name text, value integer)
```

```
      RETURNS named_value
    AS $$
      return ( name, value )

      # ###############: return [ name, value ]
    $$ LANGUAGE plpython3u;

    #####SQL NULL#############None#######

    ############Python##########################################################
    ##

#######

    ################################ ##########


    CREATE FUNCTION make_pair (name text, value integer)
      RETURNS named_value
    AS $$
      return { "name": name, "value": value }
    $$ LANGUAGE plpython3u;

    ######################## ################### #####SQL NULL################
    ###None#########

#######__getattr__##################

    ################## ##########


    CREATE FUNCTION make_pair (name text, value integer)
      RETURNS named_value
    AS $$
      class named_value:
        def __init__ (self, n, v):
          self.name = n
          self.value = v
      return named_value(name, value)

      # or simply
      class nv: pass
      nv.name = name
      nv.value = value
      return nv
    $$ LANGUAGE plpython3u;

OUT######################## ##########


CREATE FUNCTION multiout_simple(OUT i integer, OUT j integer) AS $$
return (1, 2)
$$ LANGUAGE plpython3u;

SELECT * FROM multiout_simple();

######################## ##########


CREATE PROCEDURE python_triple(INOUT a integer, INOUT b integer) AS $$
return (a * 3, b * 3)
$$ LANGUAGE plpython3u;

CALL python_triple(5, 10);
```

44.2.5.

###PL/Python######################### ##
#########################

```
CREATE TYPE greeting AS (
  how text,
  who text
);
```

#################

##################

```
   CREATE FUNCTION greet (how text)
     RETURNS SETOF greeting
   AS $$
     # return tuple containing lists as composite types
     # all other combinations work also
     return ( [ how, "World" ], [ how, "PostgreSQL" ], [ how, "PL/
   Python" ] )
   $$ LANGUAGE plpython3u;
```

######__iter__#####next#################

```
   CREATE FUNCTION greet (how text)
     RETURNS SETOF greeting
   AS $$
     class producer:
       def __init__ (self, how, who):
         self.how = how
         self.who = who
         self.ndx = -1

       def __iter__ (self):
         return self

       def next (self):
         self.ndx += 1
         if self.ndx == len(self.who):
           raise StopIteration
         return ( self.how, self.who[self.ndx] )

     return producer(how, [ "World", "PostgreSQL", "PL/Python" ])
   $$ LANGUAGE plpython3u;
```

######(yield)

```
   CREATE FUNCTION greet (how text)
     RETURNS SETOF greeting
   AS $$
     for who in [ "World", "PostgreSQL", "PL/Python" ]:
       yield ( how, who )
   $$ LANGUAGE plpython3u;
```

#RETURNS SETOF record######OUT######################## ##########

```
CREATE FUNCTION multiout_simple_setof(n integer, OUT integer, OUT integer)
 RETURNS SETOF record AS $$
```

```
return [(1, 2)] * n
$$ LANGUAGE plpython3u;

SELECT * FROM multiout_simple_setof(3);
```

44.3.

######SD## ######GD#########
############Python##########################

#####Python#################### #####myfunc#################myfunc2########
##########GD############

44.4.

PL/Python#DO#########################

```
DO $$
    # PL/Python code
$$ LANGUAGE plpython3u;
```

#################################### ###################

44.5.

###############TD###################

TD["event"]

   ################### INSERT#UPDATE#DELETE#TRUNCATE

TD["when"]

   BEFORE#AFTER####INSTEAD OF############

TD["level"]

   ROW###STATEMENT#######

TD["new"]
TD["old"]

   ##

TD["name"]

   ###########

TD["table_name"]

   ######################

TD["table_schema"]

   ########################

TD["relid"]

   ############OID#######

TD["args"]

CREATE TRIGGER#################TD["args"][0]##TD["args"][*n*-1]######
##########

TD["when"]#BEFORE###INSTEAD OF#####TD["level"]#ROW####Python##########
#######None###"OK"##############"SKIP"########## ###TD["event"]#INSERT##
#UPDATE##############"MODIFY"########## #################

44.6.

PL/Python###########plpy###Python############## ##############plpy.*foo*######
##Python################

44.6.1.

plpy##################################

plpy.execute(*query* [, *limit*])

plpy.execute###
#######

limit###############plpy.execute#######LIMIT###############*limit*####
limit##########################

################################## ################################ #######

rv = plpy.execute("SELECT * FROM my_table", 5)

####my_table##5######## my_table#my_column##########################

foo = rv[i]["my_column"]

##########len#############

#########################

nrows()

################# ######################### ####UPDATE################
##############RETURNING############

status()

SPI_execute()###########

colnames()
coltypes()
coltypmods()

############OID##########################

RETURNING#####UPDATE#DROP TABLE##################################
###################### #######################################

__str__()

###__str__###########################plpy.debug(rv)############

###############

plpy.execute## #################
################## ####################################plpy.execute###
#plpy.cursor#########

plpy.prepare(*query* [, *argtypes*])
plpy.execute(*plan* [, *arguments* [, *limit*]])

plpy.prepare################ ##
####### #######

plan = plpy.prepare("SELECT last_name FROM my_users WHERE first_name =
 $1", ["text"])

text#$1############# ##################2############

###################plpy.execute#########

rv = plpy.execute(plan, ["name"], 5)

################################## ################2############### #######
##########2############ 3#########################

#############execute################

rv = plan.execute(["name"], 5)

######################44.2#######PostgreSQL#Python##############

PL/Python########################## ################SPI#####45########### #
####################################SD###GD#44.3#################### #####
##

CREATE FUNCTION usesavedplan() RETURNS trigger AS $$
    if "plan" in SD:
        plan = SD["plan"]
    else:
        plan = plpy.prepare("SELECT 1")
        SD["plan"] = plan
    # rest of function
$$ LANGUAGE plpython3u;

plpy.cursor(*query*)
plpy.cursor(*plan* [, *arguments*])

plpy.cursor###plpy.execute#################################### ######
############################## plpy.execute##############################
##########################cursor############

##################################fetch########## fetch################
############## ######################### #############fetch#############
######## ##[2]####### ###########
##

################2###########

CREATE FUNCTION count_odd_iterator() RETURNS integer AS $$
odd = 0

[2] https://docs.python.org/library/stdtypes.html#iterator-types

```
for row in plpy.cursor("select num from largetable"):
    if row['num'] % 2:
         odd += 1
return odd
$$ LANGUAGE plpython3u;

CREATE FUNCTION count_odd_fetch(batch_size integer) RETURNS integer AS
 $$
odd = 0
cursor = plpy.cursor("select num from largetable")
while True:
    rows = cursor.fetch(batch_size)
    if not rows:
        break
    for row in rows:
        if row['num'] % 2:
            odd += 1
return odd
$$ LANGUAGE plpython3u;

CREATE FUNCTION count_odd_prepared() RETURNS integer AS $$
odd = 0
plan = plpy.prepare("select num from largetable where num % $1 <> 0",
 ["integer"])

rows = list(plpy.cursor(plan, [2]))  # ### = list(plan.cursor([2]))

return len(rows)
$$ LANGUAGE plpython3u;
```

###############################close############# ######### ##################

###

plpy.cursor################Python######API##[3]#########DB-API####### ########## ##############

44.6.2.

```
#################################################          plpy.execute##
#plpy.prepare################plpy.SPIError######################### #####
##try/except########Python############### #######
```

```
CREATE FUNCTION try_adding_joe() RETURNS text AS $$
    try:
        plpy.execute("INSERT INTO users(username) VALUES ('joe')")
    except plpy.SPIError:

        "#########" ###
    else:

        "Joe######" ###
$$ LANGUAGE plpython3u;
```

```
##################################       #       A.1##########################
plpy.spiexceptions######PostgreSQL#################################   ##
#division_by_zero#DivisionByZero#unique_violation#UniqueViolation##fdw_error#FdwE
```

[3] https://www.python.org/dev/peps/pep-0249/

########## ##############SPIError########## ################################## #### ##########

```
CREATE FUNCTION insert_fraction(numerator int, denominator int) RETURNS
 text AS $$
from plpy import spiexceptions
try:
    plan = plpy.prepare("INSERT INTO fractions (frac) VALUES ($1 / $2)",
 ["int", "int"])
    plpy.execute(plan, [numerator, denominator])
except spiexceptions.DivisionByZero:
    return "denominator cannot equal zero"
except spiexceptions.UniqueViolation:
    return "already have that fraction"
except plpy.SPIError as e:
    return "other error, SQLSTATE %s" % e.sqlstate
else:
    return "fraction inserted"
$$ LANGUAGE plpython3u;
```

plpy.spiexceptions#############SPIError##############except############ ####################

######################SPIError################sqlstate###########except# ################### ######SQLSTATE################ ##################

44.7.

44.6.2## ################################ PL/Python################################

44.7.1.

2#########################

```
CREATE FUNCTION transfer_funds() RETURNS void AS $$
try:
    plpy.execute("UPDATE accounts SET balance = balance - 100 WHERE
 account_name = 'joe'")
    plpy.execute("UPDATE accounts SET balance = balance + 100 WHERE
 account_name = 'mary'")
except plpy.SPIError as e:
    result = "error transferring funds: %s" % e.args
else:
    result = "funds transferred correctly"
plan = plpy.prepare("INSERT INTO operations (result) VALUES ($1)",
 ["text"])
plpy.execute(plan, [result])
$$ LANGUAGE plpython3u;
```

2###UPDATE##UPDATE########## #### ######################################

############plpy.execute############################ plpy##### ##plpy.subtransaction()## ####### ##############################[4]###### ## ##

[4] https://docs.python.org/library/stdtypes.html#context-manager-types

```
CREATE FUNCTION transfer_funds2() RETURNS void AS $$
try:
    with plpy.subtransaction():
        plpy.execute("UPDATE accounts SET balance = balance - 100 WHERE
 account_name = 'joe'")
        plpy.execute("UPDATE accounts SET balance = balance + 100 WHERE
 account_name = 'mary'")
except plpy.SPIError as e:
    result = "error transferring funds: %s" % e.args
else:
    result = "funds transferred correctly"
plan = plpy.prepare("INSERT INTO operations (result) VALUES ($1)",
 ["text"])
plpy.execute(plan, [result])
$$ LANGUAGE plpython3u;
```

try/except#################### #######Python####################PostgreSQL#### ###########operations########################### ########################### ####### ## ######### ## ########### #################Python###########################

44.8.

#################################DO#################################### ######## ############plpy.commit()###### #############plpy.rollback()###### #SQL## ###COMMIT#ROLLBACK#plpy.execute## #### ##

#########

```
CREATE PROCEDURE transaction_test1()
LANGUAGE plpython3u
AS $$
for i in range(0, 10):
    plpy.execute("INSERT INTO test1 (a) VALUES (%d)" % i)
    if i % 2 == 0:
        plpy.commit()
    else:
        plpy.rollback()
$$;

CALL transaction_test1();
```

##################################

44.9.

plpy####################

plpy.debug(*msg, **kwargs*)
plpy.log(*msg, **kwargs*)
plpy.info(*msg, **kwargs*)
plpy.notice(*msg, **kwargs*)
plpy.warning(*msg, **kwargs*)
plpy.error(*msg, **kwargs*)
plpy.fatal(*msg, **kwargs*)

plpy.error###plpy.fatal#####Python########## ######################### ## raise plpy.Error(*msg*)##

#raise plpy.Fatal(*msg*)######plpy.error(*msg*)###plpy.fatal(*msg*)########
###raise#################### ##########################
log_min_messages#client_min_messages##
############# ####19###########

msg################ ##########2################### ###########################
##############

####################

detail
hint
sqlstate
schema_name
table_name
column_name
datatype_name
constraint_name

####

```
CREATE FUNCTION raise_custom_exception() RETURNS void AS $$
plpy.error("custom exception message",
           detail="some info about exception",
           hint="hint for users")
$$ LANGUAGE plpython3u;

=# SELECT raise_custom_exception();
ERROR:  plpy.Error: custom exception message
DETAIL:  some info about exception
HINT:  hint for users
CONTEXT:  Traceback (most recent call last):
  PL/Python function "raise_custom_exception", line 4, in <module>
    hint="hint for users")
PL/Python function "raise_custom_exception"
```

###############
#plpy.quote_literal(*string*)#plpy.quote_nullable(*string*)##
#plpy.quote_ident(*string*)###### ####9.4#################### ##############
############# # 41.1###SQL####PL/Python########

```
plpy.execute("UPDATE tbl SET %s = %s WHERE key = %s" % (
    plpy.quote_ident(colname),
    plpy.quote_nullable(newvalue),
    plpy.quote_literal(keyvalue)))
```

44.10. Python 2#Python 3

PL/Python#Python 3########## PostgreSQL###########plpythonu#plpython2u#######
####Python 2############

44.11.

Python##########################PL/Python######################## ################
##PostgreSQL######################## ##########Python############# #######Python##
########## #########Python##############################PL/Python#########

- PYTHONHOME

- PYTHONPATH
- PYTHONY2K
- PYTHONOPTIMIZE
- PYTHONDEBUG
- PYTHONVERBOSE
- PYTHONCASEOK
- PYTHONDONTWRITEBYTECODE
- PYTHONIOENCODING
- PYTHONUSERBASE
- PYTHONHASHSEED

#python##Python################ PL/Python#######Python##############

#45#

################SPI########C#############SQL#################### SPI###### ################################### ###SPI###############

> ##
>
> ###############SQL#################### ##########SPI################## #######################

#####SPI################C################### ################################## ################ ####SPI#################################### ###########SPI#### ####################### ##########SPI### ####

SPI#######################`SPI_result`############### #############`NULL`######

SPI################`executor/spi.h`########include###########

45.1.

SPI_connect

SPI_connect, SPI_connect_ext — SPI######C#######

##

```
int SPI_connect(void)
```

```
int SPI_connect_ext(int options)
```

##

SPI_connect#C########SPI############## SPI################################## SPI#################C#################

SPI_connect_ext############################## ###################

SPI_OPT_NONATOMIC

SPI##################################(SPI_commit#SPI_rollback)######## ###### ##############################

SPI_connect()#SPI_connect_ext(0)######

###

SPI_OK_CONNECT

#######

SPI_ERROR_CONNECT

###########

SPI_finish

SPI_finish — C###SPI###########

##

```
int SPI_finish(void)
```

##

SPI_finish####SPI############### C################SPI######################### ####elog(ERROR)### #####SPI###### #######

###

SPI_OK_FINISH

###########

SPI_ERROR_UNCONNECTED

####C############

SPI_execute

SPI_execute — #########

##

```
int SPI_execute(const char * command, bool read_only, long count)
```

##

SPI_execute#####SQL######*count*######## *read_only*#true#################### ##########################

##########C###############

count#0############################## *count*#0######*count*################ ## ####LIMIT############count############### ####

```
SPI_execute("SELECT * FROM foo", true, 5);
```

############5########### ##################################### ###

```
SPI_execute("INSERT INTO foo SELECT * FROM bar", false, 5);
```

##*count*###########bar############## ####

```
SPI_execute("INSERT INTO foo SELECT * FROM bar RETURNING *", false, 5);
```

##5###RETURNING##########################5###########

########1################## SPI_execute#################### *count*############ ############################## ################################

read_only#false####################SPI_execute########################### #### ############################SERIALIZABLE###REPEATABLE READ############ ####READ COMMITTED## ###### ####################################

read_only#true#####SPI_execute######################### #######SELECT##### ###################### ## ###### ### ##############*stable*################# #### ##

#####SPI#####1## ########################### ##

#####[90.315315]######################SPI_processed############## ###### #SPI_OK_SELECT#SPI_OK_INSERT_RETURNING#SPI_OK_DELETE_RETURNING### #SPI_OK_UPDATE_RETURNING####SPITupleTable *SPI_tuptable############### ################## ##################EXPLAIN####################SPI_tuptable# ############ ##############COPY, CREATE TABLE AS############ ### #SPI_tuptable#NULL####SPI_processed############

SPITupleTable##################

```
typedef struct SPITupleTable
{

    /* ##### */
    TupleDesc   tupdesc;        /* ###### */
    HeapTuple  *vals;           /* ###### */
    uint64      numvals;        /* ######## */


    /* ######################### */
    uint64      alloced;        /* vals############ */
    MemoryContext tuptabcxt;    /* ################ */
    slist_node  next;           /* ########### */
    SubTransactionId subid;     /* SPITupleTable################ */
} SPITupleTable;
```

#####tupdesc#vals#numvals#SPI##########################vals##############
###numvals######(##################SPI_processed######)# tupdesc#####SPI
#################

SPI_finish#####C#########SPITupleTable#########SPI_freetuptable######
##############################

##

const char * *command*

##############

bool *read_only*

###########true#

long *count*

##############0#

###

##################################

SPI_OK_SELECT

SELECT#SELECT INTO############

SPI_OK_SELINTO

SELECT INTO########

SPI_OK_INSERT

INSERT########

SPI_OK_DELETE

DELETE########

SPI_OK_UPDATE

UPDATE########

SPI_OK_MERGE

MERGE########

SPI_OK_INSERT_RETURNING

INSERT RETURNING########

SPI_OK_DELETE_RETURNING

DELETE RETURNING########

SPI_OK_UPDATE_RETURNING

UPDATE RETURNING########

SPI_OK_MERGE_RETURNING

#####[78.846154]#DELETE RETURNING######## #########RETURNING########

SPI_OK_UTILITY

###########CREATE TABLE###########

SPI_OK_REWRITTEN

###########UPDATE#INSERT##################################

#######################

SPI_ERROR_ARGUMENT

command#NULL#####*count*#0######

SPI_ERROR_COPY

COPY TO stdout####COPY FROM stdin########

SPI_ERROR_TRANSACTION

################BEGIN#COMMIT#ROLLBACK#SAVEPOINT#PREPARE TRANSACTION#COMMIT PREPARED#ROLLBACK PREPARED##################

SPI_ERROR_OPUNKNOWN

#######################

SPI_ERROR_UNCONNECTED

####C###########

##

SPI############SPI_processed#SPI_tuptable##############################
SPI_exec##2####################
#########

SPI_exec

SPI_exec — ############

##

```
int SPI_exec(const char * command, long count)
```

##

SPI_exec####*read_only*######false###SPI_execute######

##

const char * *command*

##############

long *count*

##############0#

###

SPI_execute#########

SPI_execute_extended

SPI_execute_extended — ##################

##

```
int SPI_execute_extended(const char *command,
                         const SPIExecuteOptions * options)
```

##

SPI_execute_extended################################ ##############$*n*#####
##*options->params*#######(######)###################### ###########*options*###
########

##############################*options->params*##############
#PARAM_FLAG_CONST###########

options->dest#NULL############SPI_tuptable############################
########### ####################################DestReceiver#############
#########################

##

const char * *command*

#######

const SPIExecuteOptions * *options*

##############

########*options*################################## ##########################
#######*options*#############

ParamListInfo *params*

##########################NULL

bool *read_only*

###########true

bool *allow_nonatomic*

true#CALL#DO##################SPI_OPT_NONATOMIC###
#SPI_connect_ext########################

bool *must_return_tuples*

true##################################(#######0################)

uint64 *tcount*

##################0

DestReceiver * *dest*

###############DestReceiver###### NULL###SPI_execute#########
#SPI_tuptable##########

ResourceOwner *owner*

########SPI_execute_plan_extended####################
##SPI_execute_extended###################

###

####SPI_execute######

options->dest#NULL#####SPI_processed#SPI_tuptable#SPI_execute#######
options->dest#NULL######SPI_processed########SPI_tuptable#NULL####
#################DestReceiver###################

SPI_execute_with_args

SPI_execute_with_args — ##################

##

```
int SPI_execute_with_args(const char *command,
                          int nargs, Oid *argtypes,
                          Datum *values, const char *nulls,
                          bool read_only, long count)
```

##

SPI_execute_with_args############################## ##############$*n*#### ############################ *read_only*#*count*#SPI_execute#########

SPI_execute##SQL# ##################

##SPI_execute_plan####SPI_prepare## ######################## ########################### ##################### ##

##

const char * *command*

#######

int *nargs*

########$1#$2#####

Oid * *argtypes*

###########OID####*nargs*####

Datum * *values*

###########*nargs*####

const char * *nulls*

########null#######*nargs*####

nulls#NULL#####SPI_execute_with_args########null######### #####
##*nulls*################NULL###' '##########NULL###'n'### #####
##*values*################ *nulls*############################## '\0'#######
####

bool *read_only*

############true

long *count*

###############0#

###

```
    ####SPI_execute######

    ######SPI_processed#SPI_tuptable#SPI_execute###########
```

SPI_prepare

SPI_prepare — ################

##

```
SPIPlanPtr SPI_prepare(const char * command, int nargs, Oid * argtypes)
```

##

SPI_prepare############################ ################ ###### #SPI_execute_plan###############

SPI_prepare################################ ################

###########################$1#$2############################ ############ ##SPI_execute_plan############### ## #########

SPI_finish#######################SPI_prepare#########C################## ####### ######SPI_keepplan#SPI_saveplan####################

##

const char * *command*

#######

int *nargs*

#######$1#$2#####

Oid * *argtypes*

##########OID##########

###

SPI_prepare#SPIPlan###NULL########## ###SPIPlan########opaque##### ##### ##NULL#####SPI_execute#################1##SPI_result####### ## ##*command*#NULL#####*nargs*#0######*nargs*#0######*argtypes*#NULL### ##SPI_ERROR_ARGUMENT#######

##

################SPI_execute_plan#################################### #### ############SPI_execute_plan############################### ############### ###SPI_execute_plan## #### ################SPI_prepare_cursor#CURSOR_OPT_GENERIC_PLAN## #CURSOR_OPT_CUSTOM_PLAN##

##PostgreSQL########################### ####################DDL############################## ##########search_path# ##############search_path########### ########PostgreSQL 9.3########### ##### ########PREPARE#########

##########C###############

SPIPlanPtr#spi.h##opaque#################### ###############PostgreSQL#####
##################################

###########################SPIPlanPtr##################

SPI_prepare_cursor

SPI_prepare_cursor — ###############

##

```
SPIPlanPtr SPI_prepare_cursor(const char * command, int nargs,
                              Oid * argtypes, int cursorOptions)
```

##

SPI_prepare_cursor################################SPI_prepare##### ## #DeclareCursorStmt#options######nodes/parsenodes.h################ SPI_prepare######################

#####SPI_prepare_extended#############

##

const char * *command*

#######

int *nargs*

########$1#$2#####

Oid * *argtypes*

###########OID###########

int *cursorOptions*

################## ##################

###

SPI_prepare_cursor#SPI_prepare##############

##

cursorOptions#########
##CURSOR_OPT_SCROLL#CURSOR_OPT_NO_SCROLL#CURSOR_OPT_FAST_PLAN#CURSOR_OPT_GENERIC
##CURSOR_OPT_HOLD################

SPI_prepare_extended

SPI_prepare_extended — ################

##

```
SPIPlanPtr SPI_prepare_extended(const char * command,
                                const SPIPrepareOptions * options)
```

##

SPI_prepare_extended##SPI_prepare#### ##

##

const char * *command*

#######

const SPIPrepareOptions * *options*

##############

#########*options*################################## ##*options*##############

ParserSetupHook *parserSetup*

##########

void * *parserSetupArg*

parserSetup######

RawParseMode *parseMode*

raw####### RAW_PARSE_DEFAULT(##)##############

int *cursorOptions*

##################################

###

SPI_prepare_extended#SPI_prepare##############

SPI_prepare_params

SPI_prepare_params — ###############

##

```
SPIPlanPtr SPI_prepare_params(const char * command,
                              ParserSetupHook parserSetup,
                              void * parserSetupArg,
                              int cursorOptions)
```

##

SPI_prepare_params#################### ################ ####
#SPI_prepare_cursor###

#####SPI_prepare_extended#############

##

const char * *command*

#######

ParserSetupHook *parserSetup*

##########

void * *parserSetupArg*

parserSetup######

int *cursorOptions*

################## ##################

###

SPI_prepare_params#SPI_prepare##############

SPI_getargcount

SPI_getargcount — SPI_prepare####################

##

```
int SPI_getargcount(SPIPlanPtr plan)
```

##

SPI_getargcount##SPI_prepare##############################

##

SPIPlanPtr *plan*

#SPI_prepare##########

###

plan############ *plan*#NULL########SPI_result#SPI_ERROR_ARGUMENT######-1 #######

SPI_getargtypeid

SPI_getargtypeid — SPI_prepare####################OID###

##

```
Oid SPI_getargtypeid(SPIPlanPtr plan, int argIndex)
```

##

SPI_getargtypeid##SPI_prepare##########*argIndex*#########OID###### ##### #0###########

##

SPIPlanPtr *plan*

#SPI_prepare##########

int *argIndex*

0##############

###

################OID### *plan*#NULL#########*argIndex*#0###*plan*############## ##SPI_result#SPI_ERROR_ARGUMENT#####InvalidOid######

SPI_is_cursor_plan

SPI_is_cursor_plan — SPI_prepare########SPI_cursor_open########true###

##

```
bool SPI_is_cursor_plan(SPIPlanPtr plan)
```

##

SPI_prepare#######SPI_cursor_open###############
##SPI_is_cursor_plan#true###### ###########false###### ######*plan*#########
############################ ####INTO########SELECT###### ####RETURNING#####
###UPDATE######

##

SPIPlanPtr *plan*

#SPI_prepare##########

###

plan######################true####false### ###SPI_result######### #########
##########*plan*#NULL##############SPI##############SPI_result#############
##false#######

SPI_execute_plan

SPI_execute_plan — SPI_prepare###########

##

```
int SPI_execute_plan(SPIPlanPtr plan, Datum * values, const char * nulls,
                     bool read_only, long count)
```

##

SPI_execute_plan##SPI_prepare#####################
read_only#*count*#SPI_execute############

##

SPIPlanPtr *plan*

#SPI_prepare##########

Datum * *values*

########### ####################

const char * *nulls*

########NULL######### #####################

nulls#NULL####SPI_execute_plan##########NULL########## #######*nulls*#
#################NULL###' '##########NULL###'n'### #######*values*######
########### *nulls*############################## '\0'###########

bool *read_only*

############true

long *count*

###############0#

###

#####SPI_execute##############################

SPI_ERROR_ARGUMENT

plan#NULL##########*count*#0#####

SPI_ERROR_PARAM

values#NULL####*plan*###############

####SPI_processed#SPI_tuptable#SPI_execute#########

SPI_execute_plan_extended

SPI_execute_plan_extended — SPI_prepare###########

##

```
int SPI_execute_plan_extended(SPIPlanPtr plan,
                              const SPIExecuteOptions * options)
```

##

SPI_execute_plan_extended#SPI_prepare#################### ############ ##SPI_execute_plan######

###########ParamListInfo################################## ParamListInfo################################

##########SPI_tuptable## #DestReceiver################ ## ########

##

SPIPlanPtr *plan*

#SPI_prepare##########

const SPIExecuteOptions * *options*

#############

########*options*################################ ########################## ### #######*options*#############

ParamListInfo *params*

##########################NULL

bool *read_only*

###########true

bool *allow_nonatomic*

true#CALL#DO##################SPI_OPT_NONATOMIC### #SPI_connect_ext#######################

bool *must_return_tuples*

true####################################(#######0################)

uint64 *tcount*

##################0

DestReceiver * *dest*

###############DestReceiver###### NULL###SPI_execute_plan######### #SPI_tuptable##########

ResourceOwner *owner*

########################## NULL##CurrentResourceOwner####### SPI##############
################################

###

####SPI_execute_plan######

options->dest#NULL#####SPI_processed#SPI_tuptable#SPI_execute_plan## ######### *options->dest*#NULL######SPI_processed#########SPI_tuptable# NULL######## ##################DestReceiver###################

SPI_execute_plan_with_paramlist

SPI_execute_plan_with_paramlist — SPI_prepare###########

##

```
int SPI_execute_plan_with_paramlist(SPIPlanPtr plan,
                                    ParamListInfo params,
                                    bool read_only,
                                    long count)
```

##

SPI_execute_plan_with_paramlist#SPI_prepare############# ####
#SPI_execute_plan##
ParamListInfo######################### ##ParamListInfo##################
###############

#####SPI_execute_plan_extended#############

##

SPIPlanPtr *plan*

#SPI_prepare##########

ParamListInfo *params*

################## ####NULL#

bool *read_only*

###########true

long *count*

###############0#

###

####SPI_execute_plan######

####SPI_processed#SPI_tuptable#SPI_execute_plan#########

SPI_execp

SPI_execp — #############

##

```
int SPI_execp(SPIPlanPtr plan, Datum * values, const char * nulls,
 long count)
```

##

SPI_execp####*read_only*######false###SPI_execute_plan######

##

SPIPlanPtr *plan*

#SPI_prepare##########

Datum * *values*

########## ####################

const char * *nulls*

#######NULL######### ####################

nulls#NULL####SPI_execp##########NULL########## #######*nulls*######### #########NULL###' '##########NULL###'n'### #######*values*############# ### *nulls*############################## '\0'##########

long *count*

###############0#

###

SPI_execute_plan#########

####SPI_execute###SPI_processed#SPI_tuptable#######

SPI_cursor_open

SPI_cursor_open — SPI_prepare###################

##

```
Portal SPI_cursor_open(const char * name, SPIPlanPtr plan,
                       Datum * values, const char * nulls,
                       bool read_only)
```

##

SPI_cursor_open##SPI_prepare################################## ###### #SPI_execute_plan###################

#########################2######### 1### ########## 2###########C## C#### ####################################

############################## ##################################

##

const char * *name*

###########################NULL

SPIPlanPtr *plan*

#SPI_prepare##########

Datum * *values*

########## ####################

const char * *nulls*

#######NULL######### ####################

nulls#NULL####SPI_cursor_open#########NULL########## ######*nulls*### ###############NULL###' '##########NULL###'n'### ######*values*####### ######## *nulls*############################## '\0'##########

bool *read_only*

###########true

###

################# ############################## #######elog#########

SPI_cursor_open_with_args

SPI_cursor_open_with_args — ######################

##

```
Portal SPI_cursor_open_with_args(const char *name,
                                 const char *command,
                                 int nargs, Oid *argtypes,
                                 Datum *values, const char *nulls,
                                 bool read_only, int cursorOptions)
```

##

SPI_cursor_open_with_args################################ #########
#SPI_prepare_cursor#SPI_cursor_open####################

##################SPI_cursor_open####SPI_prepare_cursor###############
###

###

#########################API#############SPI_cursor_parse_open##########
###

##

const char * *name*

#######################NULL

const char * *command*

#######

int *nargs*

#######$1#$2#####

Oid * *argtypes*

##########OID####*nargs*####

Datum * *values*

##########*nargs*####

const char * *nulls*

#######null######*nargs*####

nulls#NULL#####SPI_cursor_open_with_args########null######### #####
##*nulls*################NULL###' '##########NULL###'n'### #####
##*values*############### *nulls*############################ '\0'######
####

bool *read_only*

###########true

int *cursorOptions*

################################

###

################## ###################### ########elog########

SPI_cursor_open_with_paramlist

SPI_cursor_open_with_paramlist — ################

##

```
Portal SPI_cursor_open_with_paramlist(const char *name,
                                      SPIPlanPtr plan,
                                      ParamListInfo params,
                                      bool read_only)
```

##

SPI_cursor_open_with_paramlist#SPI_prepare###############(########)#### ### #####SPI_cursor_open## ParamListInfo######################## ##ParamListInfo################## ###############

############################## ##################################

##

const char * *name*

###########################NULL

SPIPlanPtr *plan*

#SPI_prepare##########

ParamListInfo *params*

################## ####NULL#

bool *read_only*

###########true

###

################# ############################## #######elog##########

SPI_cursor_parse_open

SPI_cursor_parse_open — ##########################

##

```
Portal SPI_cursor_parse_open(const char *name,
                             const char *command,
                             const SPIParseOpenOptions * options)
```

##

SPI_cursor_parse_open#################################### ###############
######ParamListInfo####################SPI_prepare_cursor###
#SPI_cursor_open_with_paramlist############

######################SPI_prepare_cursor###
#SPI_cursor_open_with_paramlist############## ##########################
####################################

#############################*options->params*##############
#PARAM_FLAG_CONST###########

###

##

const char * *name*

##########################NULL

const char * *command*

#######

const SPIParseOpenOptions * *options*

#############

#########*options*################################ ##########################
#######*options*#############

ParamListInfo *params*

##########################NULL

int *cursorOptions*

##############################

bool *read_only*

###########true

###

################## ##################### ########elog########

SPI_cursor_find

SPI_cursor_find — ##############

##

```
Portal SPI_cursor_find(const char * name)
```

##

SPI_cursor_find################# ##
##

##

const char * *name*

#######

###

##########################NULL###

##

#######################Portal##
Portal######SPI################################Portal#################

SPI_cursor_fetch

SPI_cursor_fetch — ############

##

```
void SPI_cursor_fetch(Portal portal, bool forward, long count)
```

##

SPI_cursor_fetch################ ####FETCH SQL############# ######## #SPI_scroll_cursor_fetch##########

##

Portal *portal*

##########

bool *forward*

########################

long *count*

########

###

####SPI_processed#SPI_tuptable#SPI_execute#########

##

######## CURSOR_OPT_SCROLL##

SPI_cursor_move

SPI_cursor_move — #########

##

```
void SPI_cursor_move(Portal portal, bool forward, long count)
```

##

SPI_cursor_move################## ###MOVE SQL############## #########
#SPI_scroll_cursor_move###########

##

Portal *portal*

###########

bool *forward*

######################

long *count*

#########

##

########CURSOR_OPT_SCROLL##

SPI_scroll_cursor_fetch

SPI_scroll_cursor_fetch — ###############

##

```
void SPI_scroll_cursor_fetch(Portal portal, FetchDirection direction,
                             long count)
```

##

SPI_scroll_cursor_fetch################## ###SQL####FETCH######

##

Portal *portal*

###########

FetchDirection *direction*

FETCH_FORWARD#FETCH_BACKWARD#FETCH_ABSOLUTE#FETCH_RELATIVE#####

long *count*

FETCH_FORWARD###FETCH_BACKWARD######## FETCH_ABSOLUTE############ FETCH_RELATIVE#############

###

####SPI_execute###SPI_processed#SPI_tuptable#######

##

direction#######*count*###############SQL FETCH##############

########CURSOR_OPT_SCROLL##################FETCH_FORWARD################ ###

SPI_scroll_cursor_move

SPI_scroll_cursor_move — ########

##

```
void SPI_scroll_cursor_move(Portal portal, FetchDirection direction,
                            long count)
```

##

SPI_scroll_cursor_move################### ###SQL####MOVE######

##

Portal *portal*

###########

FetchDirection *direction*

FETCH_FORWARD#FETCH_BACKWARD#FETCH_ABSOLUTE#FETCH_RELATIVE#####

long *count*

FETCH_FORWARD###FETCH_BACKWARD######## FETCH_ABSOLUTE############# FETCH_RELATIVE#############

###

####SPI_execute###SPI_processed######## ###############SPI_tuptable#NULL# #######

##

direction########*count*################SQL FETCH##############

########CURSOR_OPT_SCROLL###################FETCH_FORWARD################# ###

SPI_cursor_close

SPI_cursor_close — ########

##

```
void SPI_cursor_close(Portal portal)
```

##

SPI_cursor_close################################

#################################### SPI_cursor_close#######################
############

##

Portal *portal*

###########

SPI_keepplan

SPI_keepplan — ##########

##

```
int SPI_keepplan(SPIPlanPtr plan)
```

##

SPI_keepplan######SPI_prepare########SPI_finish########################
####################C###########################

##

SPIPlanPtr *plan*

#########

###

####0# *plan*#NULL########SPI_ERROR_ARGUMENT

##

##############################(###########)# ###########SPI_freeplan##########

SPI_saveplan

SPI_saveplan — #########

##

```
SPIPlanPtr SPI_saveplan(SPIPlanPtr plan)
```

##

SPI_saveplan#####SPI_prepare########SPI_finish#######################
###################C###########################

##

SPIPlanPtr *plan*

########

###

############ ######NULL### #####SPI_result#############

SPI_ERROR_ARGUMENT

plan#NULL########

SPI_ERROR_UNCONNECTED

####C###########

##

############### #####SPI_finish##################SPI_freeplan#########

###SPI_keepplan######
###

SPI_register_relation

SPI_register_relation — ##############SPI#################

##

```
int SPI_register_relation(EphemeralNamedRelation enr)
```

##

SPI_register_relation##################SPI################################
############

##

EphemeralNamedRelation *enr*

###################

###

##############################

SPI_OK_REL_REGISTER

################

###########################

SPI_ERROR_ARGUMENT

enr#NULL####name######NULL###

SPI_ERROR_UNCONNECTED

####C###########

SPI_ERROR_REL_DUPLICATE

enr#name############################

SPI_unregister_relation

SPI_unregister_relation — ##############SPI############

##

```
int SPI_unregister_relation(const char * name)
```

##

SPI_unregister_relation################################

##

const char * *name*

name##################

###

################################

SPI_OK_REL_UNREGISTER

name####################

############################

SPI_ERROR_ARGUMENT

name#NULL###

SPI_ERROR_UNCONNECTED

name####C############

SPI_ERROR_REL_NOT_FOUND

name####################

SPI_register_trigger_data

SPI_register_trigger_data — ##########SPI##############

##

```
int SPI_register_trigger_data(TriggerData *tdata)
```

##

SPI_register_trigger_data##########################SPI################ ############# #########REFERENCING OLD/NEW TABLE AS#######AFTER######## ################ ########PL#######################

##

TriggerData *tdata

#########fcinfo->context######TriggerData######

###

##############################

SPI_OK_TD_REGISTER

#######################

##########################

SPI_ERROR_ARGUMENT

tdata#NULL###

SPI_ERROR_UNCONNECTED

####C###########

SPI_ERROR_REL_DUPLICATE

#####################################

45.2.

##########SPI_execute###SPI####################################

##################C####################

SPI_fname

SPI_fname — ################

##

```
char * SPI_fname(TupleDesc rowdesc, int colnumber)
```

##

SPI_fname ################ ###############pfree#####################

##

TupleDesc *rowdesc*

######

int *colnumber*

#1########

###

####### *colnumber*#######NULL### #####SPI_result#SPI_ERROR_NOATTRIBUTE### #####

SPI_fnumber

SPI_fnumber — ###############

##

```
int SPI_fnumber(TupleDesc rowdesc, const char * colname)
```

##

SPI_fnumber##################

colname##ctid############################## ####################
#SPI_ERROR_NOATTRIBUTE############################## ###################0##
##0######################

##

TupleDesc *rowdesc*

######

const char * *colname*

##

###

###########1######### ##################SPI_ERROR_NOATTRIBUTE###

SPI_getvalue

SPI_getvalue — #############

##

```
char * SPI_getvalue(HeapTuple row, TupleDesc rowdesc, int colnumber)
```

##

SPI_getvalue###################

####palloc################### ###########pfree####################

##

HeapTuple *row*

########

TupleDesc *rowdesc*

######

int *colnumber*

#1########

###

##NULL#######*colnumber*#######NULL## #SPI_result#SPI_ERROR_NOATTRIBUTE######## ##################NULL## #SPI_result#SPI_ERROR_NOOUTFUNC########

SPI_getbinval

SPI_getbinval — ##############

##

```
Datum SPI_getbinval(HeapTuple row, TupleDesc rowdesc, int colnumber,
                    bool * isnull)
```

##

SPI_getbinval ###############Datum########

#################### ####################################

##

HeapTuple *row*

########

TupleDesc *rowdesc*

######

int *colnumber*

#1########

bool * *isnull*

##NULL########

###

############## *isnull*############NULL####################

#####SPI_result#SPI_ERROR_NOATTRIBUTE#######

SPI_gettype

SPI_gettype — ##############

##

```
char * SPI_gettype(TupleDesc rowdesc, int colnumber)
```

##

SPI_gettype################### ##########pfree#######################

##

TupleDesc *rowdesc*

######

int *colnumber*

#1########

###

############ #####NULL### #####SPI_result#SPI_ERROR_NOATTRIBUTE#######

SPI_gettypeid

SPI_gettypeid — ###########OID###

##

```
Oid SPI_gettypeid(TupleDesc rowdesc, int colnumber)
```

##

SPI_gettypeid############OID######

##

TupleDesc *rowdesc*

######

int *colnumber*

#1########

###

###########OID### #####InvalidOid### ### ##SPI_result#SPI_ERROR_NOATTRIBUTE#######

SPI_getrelname

SPI_getrelname — ################

##

```
char * SPI_getrelname(Relation rel)
```

##

SPI_getrelname################### ###########pfree#######################

##

Relation *rel*

########

###

################

SPI_getnspname

SPI_getnspname — ##################

##

```
char * SPI_getnspname(Relation rel)
```

##

SPI_getnspname######Relation################## ################### ######## ##########pfree############

##

Relation *rel*

########

###

####################

SPI_result_code_string

SPI_result_code_string — ##############

##

```
const char * SPI_result_code_string(int code);
```

##

SPI_result_code_string#####SPI##########SPI_result######################

##

int *code*

#####

###

##########

45.3.

PostgreSQL###################### ### #### ################################### ################################### ###### ############################## palloc##############################

SPI_connect################################ SPI_finish################ ##SPI_connect################## ##########C####################C########### ####################

###################C#################################SPI######### ##palloc#################### ################SPI_finish#########C####### ######## ###########SPI_palloc############################## SPI_palloc############################## ############SPI_connect########## ##################C########################## ############################## ################################

SPI_connect########SPI_connect########C########################## palloc#repalloc#SPI### ##### C###SPI#######SPI_finish#####################################C###### ######################################

SPI_palloc

SPI_palloc — ##########################

##

```
void * SPI_palloc(Size size)
```

##

SPI_palloc###########################

#####SPI#################### ##################

##

Size *size*

   ###########

###

##################

SPI_repalloc

SPI_repalloc — ##########################

##

```
void * SPI_repalloc(void * pointer, Size size)
```

##

SPI_repalloc#####SPI_palloc##############################

##########repalloc############## ########################

##

```
void * pointer
```

##############

```
Size size
```

############

###

####################################

SPI_pfree

SPI_pfree — ########################

##

```
void SPI_pfree(void * pointer)
```

##

SPI_pfree#####SPI_palloc#SPI_repalloc####################

###########pfree############## ########################

##

```
void * pointer
```

##############

SPI_copytuple

SPI_copytuple — ####################

##

```
HeapTuple SPI_copytuple(HeapTuple row)
```

##

SPI_copytuple############################ ########################### ###### ################SPI_returntuple#########

#####SPI#################### ########NULL## ##SPI_result#SPI_ERROR_UNCONNECTED#######

##

HeapTuple *row*

#######

###

################NULL ############SPI_result#########

SPI_returntuple

SPI_returntuple — Datum##############

##

```
HeapTupleHeader SPI_returntuple(HeapTuple row, TupleDesc rowdesc)
```

##

SPI_returntuple##############################Datum####### ############
#PointerGetDatum#####Datum##############

#####SPI##################### ########NULL##
##SPI_result#SPI_ERROR_UNCONNECTED#######

#################################### ############ ##############
#SPI_copytuple#########

##

HeapTuple *row*

#######

TupleDesc *rowdesc*

##

###

###########HeapTupleHeader#########NULL### ###########SPI_result#######
##

SPI_modifytuple

SPI_modifytuple — ########################

##

```
HeapTuple SPI_modifytuple(Relation rel, HeapTuple row, int ncols,
                          int * colnum, Datum * values, const char
 * nulls)
```

##

SPI_modifytuple## ############################

#####SPI#################### ########NULL#####SPI_result#SPI_ERROR_UNCONNECTED#######

##

Relation *rel*

   ##################### ############################

HeapTuple *row*

   ######

int *ncols*

   #######

int * *colnum*

   ############*ncols*########1#######

Datum * *values*

   ###############*ncols*####

const char * *nulls*

   ########NULL#######*ncols*####

   nulls#NULL#####SPI_modifytuple########null######### #######*nulls*###### #############NULL###' '###########NULL###'n'### #######*values*########### ###### *nulls*############################## '\0'###########

###

################################## #####NULL### ############SPI_result######## ##

#####SPI_result#############

SPI_ERROR_ARGUMENT

   rel#NULL####*row*#NULL####*ncols*#0######*colnum*#NULL####*values*#NULL####

SPI_ERROR_NOATTRIBUTE

colnum########### #0###*row*#######

SPI_ERROR_UNCONNECTED

SPI##########

SPI_freetuple

SPI_freetuple — ##############################

##

```
void SPI_freetuple(HeapTuple row)
```

##

SPI_freetuple##################################

###########heap_freetuple############### #########################

##

HeapTuple *row*

#####

SPI_freetuptable

SPI_freetuptable — `SPI_execute`#######################

##

```
void SPI_freetuptable(SPITupleTable * tuptable)
```

##

`SPI_freetuptable`#####`SPI_execute`###SPI##
###`SPI_tuptable`####################

#####SPI### ##########
##`SPI_finish`#################### ################SPI####################SPI##
##################################

PostgreSQL 9.3##`SPI_freetuptable`#################################### #######
############################

##

```
SPITupleTable * tuptable
```

  ############################NULL#

SPI_freeplan

SPI_freeplan — ################

##

```
int SPI_freeplan(SPIPlanPtr plan)
```

##

SPI_freeplan#####SPI_prepare##########SPI_keepplan#SPI_saveplan###### ##########

##

SPIPlanPtr *plan*

##########

###

####0# *plan*#NULL########SPI_ERROR_ARGUMENT###

45.4.

COMMIT#ROLLBACK##################SPI_execute###SPI################## ##### ##SPI################################

##################################SQL### ############ ####SQL##########SQL## ###### #################CALL################CALL############################ #### SPI####C##

SPI_commit

SPI_commit, SPI_commit_and_chain — ##################

##

```
void SPI_commit(void)

void SPI_commit_and_chain(void)
```

##

SPI_commit################### ###SQL#####COMMIT############## ############ ##SPI############# # ##

SPI_commit_and_chain##################SQL####COMMIT AND CHAIN######### ##########################

#######SPI###SPI_connect_ext############################

SPI_rollback

SPI_rollback, SPI_rollback_and_chain — ################

##

```
void SPI_rollback(void)
```

```
void SPI_rollback_and_chain(void)
```

##

SPI_rollback###################### ###SQL#####ROLLBACK############## ##SPI#######

SPI_rollback_and_chain##################SQL####ROLLBACK AND CHAIN################################

#######SPI###SPI_connect_ext############################

SPI_start_transaction

SPI_start_transaction — #####

##

```
void SPI_start_transaction(void)
```

##

SPI_start_transaction##########PostgreSQL#######################
SPI_commit###SPI_rollback##

45.5.

SPI#######C##############################

- SQL## ####

  ```
  INSERT INTO a SELECT * FROM a;
  ```

  #########SELECT#########

- ####C########C###################### C#############C##################

- SQL################################SPI###############SPI################### ########### ### ######################################2###########

- #######################SPI############## STABLE###IMMUTABLE################ ######VOLATILE#################### C#################################### #####

##########################

45.6.

#####SPI################# C##execq#1#######SQL#####2############ ##SPI_exec############################ SPI###############src/test/regress/regress.c#spi##########

```
#include "postgres.h"

#include "executor/spi.h"
#include "utils/builtins.h"

PG_MODULE_MAGIC;

PG_FUNCTION_INFO_V1(execq);

Datum
execq(PG_FUNCTION_ARGS)
{
    char *command;
    int cnt;
    int ret;
    uint64 proc;
```

```
    /* ################C###### */
    command = text_to_cstring(PG_GETARG_TEXT_PP(0));
    cnt = PG_GETARG_INT32(1);

    SPI_connect();

    ret = SPI_exec(command, cnt);

    proc = SPI_processed;

    /*

     * ##################elog(INFO)#######
     */
    if (ret > 0 && SPI_tuptable != NULL)
    {
        SPITupleTable *tuptable = SPI_tuptable;
        TupleDesc tupdesc = tuptable->tupdesc;
        char buf[8192];
        uint64 j;

        for (j = 0; j < tuptable->numvals; j++)
        {
            HeapTuple tuple = tuptable->vals[j];
            int i;

            for (i = 1, buf[0] = 0; i <= tupdesc->natts; i++)
                snprintf(buf + strlen(buf), sizeof(buf) - strlen(buf), "
 %s%s",
                         SPI_getvalue(tuple, tupdesc, i),
                         (i == tupdesc->natts) ? " " : " |");
            elog(INFO, "EXECQ: %s", buf);
        }
    }

    SPI_finish();
    pfree(command);

    PG_RETURN_INT64(proc);
}
```

##################36.10.5#######################

```
CREATE FUNCTION execq(text, integer) RETURNS int8
    AS 'filename'
    LANGUAGE C STRICT;
```

#############

```
=> SELECT execq('CREATE TABLE a (x integer)', 0);
 execq
-------
     0
(1 row)

=> INSERT INTO a VALUES (execq('INSERT INTO a VALUES (0)', 0));
INSERT 0 1
=> SELECT execq('SELECT * FROM a', 0);

INFO:  EXECQ:  0    -- execq#########
INFO:  EXECQ:  1    -- execq###########INSERT#########
```

```
 execq
-------
     2
(1 row)

=> SELECT execq('INSERT INTO a SELECT x + 2 FROM a RETURNING *', 1);

INFO:  EXECQ:  2    -- 0 + 2#####################
 execq
-------
     1
(1 row)

=> SELECT execq('SELECT * FROM a', 10);
INFO:  EXECQ:  0
INFO:  EXECQ:  1
INFO:  EXECQ:  2

 execq
-------

     3              -- 10###########3######
(1 row)

=> SELECT execq('INSERT INTO a SELECT x + 10 FROM a', 1);
 execq
-------

     3              -- ##################################
(1 row)

=> SELECT * FROM a;
 x
----
  0
  1
  2
 10
 11
 12
(6 rows)

=> DELETE FROM a;
DELETE 6
=> INSERT INTO a VALUES (execq('SELECT * FROM a', 0) + 1);
INSERT 0 1
=> SELECT * FROM a;
 x
---

 1                  -- 0 (a######) + 1
(1 row)

=> INSERT INTO a VALUES (execq('SELECT * FROM a', 0) + 1);
INFO:  EXECQ:  1
INSERT 0 1
=> SELECT * FROM a;
 x
---
 1

 2                  -- 1 (a##1###) + 1
(2 rows)
```

```
-- #######################
-- execq#2#################

=> INSERT INTO a SELECT execq('SELECT * FROM a', 0) * x FROM a;

INFO:  EXECQ:  1    -- 1###execq###
INFO:  EXECQ:  2

INFO:  EXECQ:  1    -- 2###execq###
INFO:  EXECQ:  2
INFO:  EXECQ:  2
INSERT 0 2
=> SELECT * FROM a;
 x
---
 1
 2

 2                  -- 2 # * 1 (#####x)
 6                  -- 3 # (2 + #########1) * 2 (2###x)
(4 rows)
```

#46#

PostgreSQL################################ #######postgres################## ###################### ########PostgreSQL################################## #### ## ########### #libpq##

##
##C################## ############## ## ## ##

###################shared_preload_libraries#########PostgreSQL########### ## ##########################_PG_init()### #RegisterBackgroundWorker(BackgroundWorker *worker)############## #### ################RegisterDynamicBackgroundWorker(BackgroundWorker *worker, BackgroundWorkerHandle **handle)####### ############## postmaster#################RegisterBackgroundWorker#### ##RegisterDynamicBackgroundWorker#####################################

BackgroundWorker#################

```
typedef void (*bgworker_main_type)(Datum main_arg);
typedef struct BackgroundWorker
{
    char        bgw_name[BGW_MAXLEN];
    char        bgw_type[BGW_MAXLEN];
    int         bgw_flags;
    BgWorkerStartTime bgw_start_time;
    int         bgw_restart_time;       /* in seconds, or
 BGW_NEVER_RESTART */
    char        bgw_library_name[MAXPGPATH];
    char        bgw_function_name[BGW_MAXLEN];
    Datum       bgw_main_arg;
    char        bgw_extra[BGW_EXTRALEN];
    pid_t       bgw_notify_pid;
} BackgroundWorker;
```

bgw_name#bgw_type################################### bgw_type############ ## ###bgw_name############## ############ ####bgw_name##

bgw_flags##############OR#########################

BGWORKER_SHMEM_ACCESS

#################### ###########

BGWORKER_BACKEND_DATABASE_CONNECTION

################################## BGWORKER_BACKEND_DATABASE_CONNECTION######################### #BGWORKER_SHMEM_ACCESS######################################

bgw_start_time##postgres######################## ################### BgWorkerStart_PostmasterStart#postgres##################################

################### BgWorkerStart_ConsistentState#################### ################################### BgWorkerStart_RecoveryFinished##### #####/####################### ## ######## ############################ ################################

bgw_restart_time##################postgres############################## ##### ##############################BGW_NEVER_RESTART#######

bgw_library_name#################################### ################## ################################bgw_function_name####### ############## ##"postgres"############

bgw_function_name## ############### #########PGDLLEXPORT#static################

bgw_main_arg###################Datum##### ########Datum######void##### bgw_main_arg########### ###########MyBgworkerEntry######## #BackgroundWorker############## ########################

Windows#########EXEC_BACKEND############################Datum########## ################## ##########int32## ######## cstring#text##

bgw_extra############################## bgw_main_arg#################### ######################MyBgworkerEntry################

bgw_notify_pid##############postmaster#SIGUSR1#####PostgreSQL##########PID# ## ###postmaster###0######### ## ########MyProcPid#############

Once running, the process can connect to a database by calling `BackgroundWorkerInitializeConnection(*char *dbname, char *username, uint32 flags*)` or `BackgroundWorkerInitializeConnectionByOid(*Oid dboid, Oid useroid, uint32 flags*)`. This allows the process to run transactions and queries using the `SPI` interface. If `dbname` is NULL or `dboid` is `InvalidOid`, the session is not connected to any particular database, but shared catalogs can be accessed. If `username` is NULL or `useroid` is `InvalidOid`, the process will run as the superuser created during `initdb`. If `BGWORKER_BYPASS_ALLOWCONN` is specified as `flags` it is possible to bypass the restriction to connect to databases not allowing user connections. If `BGWORKER_BYPASS_ROLELOGINCHECK` is specified as `flags` it is possible to bypass the login check for the role used to connect to databases. A background worker can only call one of these two functions, and only once. It is not possible to switch databases.

#BackgroundWorkerUnblockSignals############ ##BackgroundWorkerBlockSignals################

#############bgw_restart_time#BGW_NEVER_RESTART##################0###### #####TerminateBackgroundWorker##########postmaster################## ###### ##bgw_restart_time##################################postmaster############# ########### ## ## #WaitLatch()################## ############WL_POSTMASTER_DEATH########## #####postgres##################################

############RegisterDynamicBackgroundWorker########################### ######################## ########RegisterDynamicBackgroundWorker#2###### ##BackgroundWorkerHandle *############### ################### #GetBackgroundWorkerPid(*BackgroundWorkerHandle *,pid_t *)# ##TerminateBackgroundWorker(*BackgroundWorkerHandle *)########

opaque(###)############# GetBackgroundWorkerPid#########################
BGWH_NOT_YET_STARTED######postmaster########### BGWH_STOPPED#
################ BGWH_STARTED###### ##########PID##2##############
TerminateBackgroundWorker#############postmaster######SIGTERM############
##########

###
bgw_notify_pid#MyProcPid#############BackgroundWorkerHandle *####
#WaitForBackgroundWorkerStartup(*BackgroundWorkerHandle *handle,pid_t **)############### postmaster#######################postmaster###
############### ##########################BGWH_STARTED#############PID#####
###########BGWH_STOPPED###BGWH_POSTMASTER_DIED######

#######BackgroundWorkerHandle *####
#WaitForBackgroundWorkerShutdown(*BackgroundWorkerHandle *handle*)####
##################postmaster###########
####### #######################BGWH_STOPPED#postmaster#########
#BGWH_POSTMASTER_DIED######

#############SPI###NOTIFY#############Async_Notify()#####################
#############LISTEN#######################
######### ############################

##################src/test/modules/worker_spi############# #############
#######

######################max_worker_processes############

#47#

PostgreSQL##SQL## ################## ####################

####################(logical replication slots)##################

##########PostgreSQL############ ############PostgreSQL## ######### #####INSERT################UPDATE########################## UPDATE#DELETE##(replica identity)######### ####REPLICA IDENTITY####

################################53.4#47.3########SQL##############47.4##### ##47.7####

47.1.

##############SQL###########

################wal_level#logical####max_replication_slots#####1############## #################(#####postgres)########

```
postgres=# -- Create a slot named 'regression_slot' using the output
 plugin 'test_decoding'
postgres=# -- #######'test_decoding'#####'regression_slot'###############
##
postgres=# SELECT * FROM
 pg_create_logical_replication_slot('regression_slot', 'test_decoding',
 false, true);
    slot_name    |    lsn
-----------------+-----------
 regression_slot | 0/16B1970
(1 row)

postgres=# SELECT slot_name, plugin, slot_type, database, active,
 restart_lsn, confirmed_flush_lsn FROM pg_replication_slots;
    slot_name    |    plugin     | slot_type | database | active |
 restart_lsn | confirmed_flush_lsn
-----------------+---------------+-----------+----------+--------
+-------------+-----------------
 regression_slot | test_decoding | logical   | postgres | f      |
 0/16A4408   | 0/16A4440
(1 row)

postgres=# -- There are no changes to see yet
postgres=# -- ##########
postgres=# SELECT * FROM pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
 lsn | xid | data
-----+-----+------
(0 rows)

postgres=# CREATE TABLE data(id serial primary key, data text);
CREATE TABLE

postgres=# -- DDL isn't replicated, so all you'll see is the transaction
postgres=# -- DDL##################################
postgres=# SELECT * FROM pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
    lsn    |  xid  |     data
```

```
-----------+-------+--------------
 0/BA2DA58 | 10297 | BEGIN 10297
 0/BA5A5A0 | 10297 | COMMIT 10297
(2 rows)

postgres=# -- Once changes are read, they're consumed and not emitted
postgres=# -- in a subsequent call:
postgres=# -- ##################################:
postgres=# SELECT * FROM pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
 lsn | xid | data
-----+-----+------
(0 rows)

postgres=# BEGIN;
postgres=*# INSERT INTO data(data) VALUES('1');
postgres=*# INSERT INTO data(data) VALUES('2');
postgres=*# COMMIT;

postgres=# SELECT * FROM pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
    lsn    |  xid  |                          data
-----------+-------
+-------------------------------------------------------
 0/BA5A688 | 10298 | BEGIN 10298
 0/BA5A6F0 | 10298 | table public.data: INSERT: id[integer]:1
 data[text]:'1'
 0/BA5A7F8 | 10298 | table public.data: INSERT: id[integer]:2
 data[text]:'2'
 0/BA5A8A8 | 10298 | COMMIT 10298
(4 rows)

postgres=# INSERT INTO data(data) VALUES('3');

postgres=# -- You can also peek ahead in the change stream without
 consuming changes
postgres=# -- ############################
postgres=# SELECT * FROM pg_logical_slot_peek_changes('regression_slot',
 NULL, NULL);
    lsn    |  xid  |                          data
-----------+-------
+-------------------------------------------------------
 0/BA5A8E0 | 10299 | BEGIN 10299
 0/BA5A8E0 | 10299 | table public.data: INSERT: id[integer]:3
 data[text]:'3'
 0/BA5A990 | 10299 | COMMIT 10299
(3 rows)

postgres=# -- The next call to pg_logical_slot_peek_changes() returns the
 same changes again
postgres=# -- pg_logical_slot_peek_changes()###################
postgres=# SELECT * FROM pg_logical_slot_peek_changes('regression_slot',
 NULL, NULL);
    lsn    |  xid  |                          data
-----------+-------
+-------------------------------------------------------
 0/BA5A8E0 | 10299 | BEGIN 10299
 0/BA5A8E0 | 10299 | table public.data: INSERT: id[integer]:3
 data[text]:'3'
 0/BA5A990 | 10299 | COMMIT 10299
(3 rows)

postgres=# -- options can be passed to output plugin, to influence the
 formatting
```

```
postgres=# -- ########################################
postgres=# SELECT * FROM pg_logical_slot_peek_changes('regression_slot',
 NULL, NULL, 'include-timestamp', 'on');
    lsn    |  xid  |                          data
-----------+-------
+-------------------------------------------------------
 0/BA5A8E0 | 10299 | BEGIN 10299
 0/BA5A8E0 | 10299 | table public.data: INSERT: id[integer]:3
 data[text]:'3'
 0/BA5A990 | 10299 | COMMIT 10299 (at 2017-05-10 12:07:21.272494-04)
(3 rows)

postgres=# -- Remember to destroy a slot you no longer need to stop it
 consuming
postgres=# -- server resources:
postgres=# -- ##############################################
postgres=# SELECT pg_drop_replication_slot('regression_slot');
 pg_drop_replication_slot
------------------------

(1 row)
```

#####[92.914654]####PostgreSQL##########pg_recvlogical#############################
################## ####################################(26.2.5.1##)#max_wal_senders################################## 2######2################
2############max_prepared_transactions######1############

```
Example 1:
$ pg_recvlogical -d postgres --slot=test --create-slot
$ pg_recvlogical -d postgres --slot=test --start -f -
Control+Z
$ psql -d postgres -c "INSERT INTO data(data) VALUES('4');"
$ fg
BEGIN 693
table public.data: INSERT: id[integer]:4 data[text]:'4'
COMMIT 693
Control+C
$ pg_recvlogical -d postgres --slot=test --drop-slot

Example 2:
$ pg_recvlogical -d postgres --slot=test --create-slot --two-phase
$ pg_recvlogical -d postgres --slot=test --start -f -
Control+Z
$ psql -d postgres -c "BEGIN;INSERT INTO data(data) VALUES('5');PREPARE
 TRANSACTION 'test';"
$ fg
BEGIN 694
table public.data: INSERT: id[integer]:5 data[text]:'5'
PREPARE TRANSACTION 'test', txid 694
Control+Z
$ psql -d postgres -c "COMMIT PREPARED 'test';"
$ fg
COMMIT PREPARED 'test', txid 694
Control+C
$ pg_recvlogical -d postgres --slot=test --drop-slot
```

##############################SQL############ 2###############
##max_prepared_transactions######1############
pg_create_logical_replication_slot################2#######'true'###########
##

```
postgres=# BEGIN;
```

```
postgres=*# INSERT INTO data(data) VALUES('5');
postgres=*# PREPARE TRANSACTION 'test_prepared1';

postgres=# SELECT * FROM pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
    lsn    | xid |                          data
-----------+-----
+-------------------------------------------------------
 0/1689DC0 | 529 | BEGIN 529
 0/1689DC0 | 529 | table public.data: INSERT: id[integer]:3 data[text]:'5'
 0/1689FC0 | 529 | PREPARE TRANSACTION 'test_prepared1', txid 529
(3 rows)

postgres=# COMMIT PREPARED 'test_prepared1';
postgres=# select * from pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
    lsn    | xid |                     data
-----------+-----+---------------------------------------------
 0/168A060 | 529 | COMMIT PREPARED 'test_prepared1', txid 529
(4 row)

postgres=#-- you can also rollback a prepared transaction
postgres=#-- ##############################
postgres=# BEGIN;
postgres=*# INSERT INTO data(data) VALUES('6');
postgres=*# PREPARE TRANSACTION 'test_prepared2';
postgres=# select * from pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
    lsn    | xid |                          data
-----------+-----
+-------------------------------------------------------
 0/168A180 | 530 | BEGIN 530
 0/168A1E8 | 530 | table public.data: INSERT: id[integer]:4 data[text]:'6'
 0/168A430 | 530 | PREPARE TRANSACTION 'test_prepared2', txid 530
(3 rows)

postgres=# ROLLBACK PREPARED 'test_prepared2';
postgres=# select * from pg_logical_slot_get_changes('regression_slot',
 NULL, NULL);
    lsn    | xid |                      data
-----------+-----+---------------------------------------------
 0/168A4B8 | 530 | ROLLBACK PREPARED 'test_prepared2', txid 530
(1 row)
```

47.2.

47.2.1.

##
##########

PostgreSQL##SQL###############
############################

47.2.2.

#########
################################

> **##**
>
> ##PostgreSQL########################## (26.2.5##)########################
> ##

##################PostgreSQL#################### ##############################
########

#################################### ####################################
####################LSN###
####################
###LSN###
LSN########################## ############################## replication origins#
########

#################################### ####################################
########################## ##

################################## ##################################### ####
######1##################1##########

#####[90.770609]################################### ###############VACUUM####
################hot_standby_feedback############ ######################
#######hot_standby_feedback#
##################################
#########################catalog_xmin############ ########################
#wal_level#logical################ ########WAL##########################
########walsender##################wal_level#############WAL###########
#########

##################################
################ ## ########
########
#pg_log_standby_snapshot################

> **##**
>
> ## ##########################
> ################## ##########################WAL####################
> ##########VACUUM############## ##############ID####24.1.5##############
> ################ ########################

47.2.3. Replication Slot Synchronization

###
`pg_create_logical_replication_slot`, or by using the `failover` option of `CREATE SUBSCRIPTION` during slot creation, and then calling `pg_sync_replication_slots` on the standby. By setting `sync_replication_slots` on the standby, the failover slots can be synchronized periodically in the slotsync worker. For the synchronization to work, it is mandatory to have a physical replication slot between the primary and the standby (i.e., `primary_slot_name` should be configured on the standby), and `hot_standby_feedback` must be enabled on the standby. It is also necessary to specify a valid `dbname` in the `primary_conninfo`. It's highly recommended that the said physical replication slot is named in `synchronized_standby_slots` list on the primary, to prevent the subscriber from

consuming changes faster than the hot standby. Even when correctly configured, some latency is expected when sending changes to logical subscribers due to the waiting on slots named in `synchronized_standby_slots`. When `synchronized_standby_slots` is utilized, the primary server will not completely shut down until the corresponding standbys, associated with the physical replication slots specified in `synchronized_standby_slots`, have confirmed receiving the WAL up to the latest flushed position on the primary server.

##pg_##### _slots.`synced`######### ## ######### ## ######## ##

##'conninfo'########## ##################### ###`ALTER`#########.################## ################## ##################################

> ##
>
> ##################################### ################################## ## ############### ##

47.2.4.

###

47.2.5.

#######################################CREATE_REPLICATION_SLOT############# ###########9.28.5#### ## ############# ############################SET TRANSACTION SNAPSHOT###### ######################### ## ###### ##

############################ ########################## ######################## #######NOEXPORT_SNAPSHOT##############################

47.3.

####

- `CREATE_REPLICATION_SLOT` *`slot_name`* `LOGICAL` *`output_plugin`*
- `DROP_REPLICATION_SLOT` *`slot_name`* `[WAIT]`
- `START_REPLICATION SLOT` *`slot_name`* `LOGICAL ...`

SQL### ####### ################53.4##########

####pg_recvlogical#####################################(#######################)#

47.4. ############SQL#######

############SQL####API#########9.28.6##########

#########(26.2.8##)## ###############
##################################

47.5.

pg_replication_slots####pg_stat_replication########################
################################## pg_stat_replication_slots#############
####################

47.6.

PostgreSQL############### contrib/test_decoding #################

47.6.1.

###############
################################_PG_output_plugin_init################
##

```
typedef struct OutputPluginCallbacks
{
    LogicalDecodeStartupCB startup_cb;
    LogicalDecodeBeginCB begin_cb;
    LogicalDecodeChangeCB change_cb;
    LogicalDecodeTruncateCB truncate_cb;
    LogicalDecodeCommitCB commit_cb;
    LogicalDecodeMessageCB message_cb;
    LogicalDecodeFilterByOriginCB filter_by_origin_cb;
    LogicalDecodeShutdownCB shutdown_cb;
    LogicalDecodeFilterPrepareCB filter_prepare_cb;
    LogicalDecodeBeginPrepareCB begin_prepare_cb;
    LogicalDecodePrepareCB prepare_cb;
    LogicalDecodeCommitPreparedCB commit_prepared_cb;
    LogicalDecodeRollbackPreparedCB rollback_prepared_cb;
    LogicalDecodeStreamStartCB stream_start_cb;
    LogicalDecodeStreamStopCB stream_stop_cb;
    LogicalDecodeStreamAbortCB stream_abort_cb;
    LogicalDecodeStreamPrepareCB stream_prepare_cb;
    LogicalDecodeStreamCommitCB stream_commit_cb;
    LogicalDecodeStreamChangeCB stream_change_cb;
    LogicalDecodeStreamMessageCB stream_message_cb;
    LogicalDecodeStreamTruncateCB stream_truncate_cb;
} OutputPluginCallbacks;

typedef void (*LogicalOutputPluginInit) (struct OutputPluginCallbacks
 *cb);
```

```
#########begin_cb#change_cb#####commit_cb#####
##startup_cb#truncate_cb#message_cb#filter_by_origin_cb###
##shutdown_cb########## truncate_cb#############TRUNCATE################
############
```

```
###############(in-progress)#####################################
stream_start_cb#stream_stop_cb#stream_abort_cb#stream_commit_cb#stream_change_cb
######stream_message_cb#stream_truncate_cb########## #######2###########
#####stream_prepare_cb######
```

```
#########PREPARE   TRANSACTION##################2##########################
begin_prepare_cb#prepare_cb#commit_prepared_cb#rollback_prepared_cb##
```

##########filter_prepare_cb########## ##################################
######stream_prepare_cb#####

47.6.2.

##
####initdb#####pg_catalog##
#################

```
ALTER TABLE user_catalog_table SET (user_catalog_table = true);
CREATE TABLE another_catalog_table(data text) WITH (user_catalog_table =
 true);
```

##systable_*####API##############
########## heap_*####API################# ###########ID#####################
###############DDL######pg_current_xact_id()#############

47.6.3.

SQL############################
########(####bytea)######### ##
###OutputPluginOptions.output_type#OUTPUT_PLUGIN_BINARY_OUTPUT###
##OUTPUT_PLUGIN_TEXTUAL_OUTPUT################# #####textdatum###########
##

47.6.4.

##

################################## ######################begin#commit#######
############# ### ####################
############################### PREPARE TRANSACTION#####2##################
ROLLBACK PREPARED#################
########################### ################################### #############
##################prepare_cb##################### ########################
##ROLLBACK PREPARED############################

> ##
>
> ################################## #####synchronous_commit#off######
> ########pg_logical_slot_get_changes()###COMMIT###############

47.6.4.1.

###startup_cb##
###########

```
typedef void (*LogicalDecodeStartupCB) (struct LogicalDecodingContext
 *ctx,
                                        OutputPluginOptions *options,
                                        bool is_init);
```

is_init ##########################true#######false###### *options*###############
##########

```
typedef struct OutputPluginOptions
{
    OutputPluginOutputType output_type;
    bool        receive_rewrites;
} OutputPluginOptions;
```

output_type#OUTPUT_PLUGIN_TEXTUAL_OUTPUT#OUTPUT_PLUGIN_BINARY_OUTPUT# ####### 47.6.3######### receive_rewrites#######DDL######################### ######### ###DDL##

##########ctx->output_plugin_options################## ################# #####ctx->output_plugin_private#######

47.6.4.2.

##################################shutdown_cb############################ ######## ################################

```
typedef void (*LogicalDecodeShutdownCB) (struct LogicalDecodingContext
 *ctx);
```

47.6.4.3.

#####begin_cb### ################### ############

```
typedef void (*LogicalDecodeBeginCB) (struct LogicalDecodingContext *ctx,
                                      ReorderBufferTXN *txn);
```

txn#########################ID######################

47.6.4.4.

#####commit_cb## ################### #change_cb#######commit_cb##########

```
typedef void (*LogicalDecodeCommitCB) (struct LogicalDecodingContext *ctx,
                                       ReorderBufferTXN *txn,
                                       XLogRecPtr commit_lsn);
```

47.6.4.5.

#########INSERT#UPDATE#DELETE##################change_cb######## ####### ## change_cb######################### ################################## #####(###########)######################### ## ##

```
typedef void (*LogicalDecodeChangeCB) (struct LogicalDecodingContext *ctx,
                                       ReorderBufferTXN *txn,
                                       Relation relation,
                                       ReorderBufferChange *change);
```

ctx#txn##begin_cb#commit_cb################ ######*relation*############## ##########*change*###########

##
######################UNLOGGED##############TEMPORARY###TEMP####### ################################

47.6.4.6. TRUNCATE######

######truncate_cb#######TRUNCATE#############

```
typedef void (*LogicalDecodeTruncateCB) (struct LogicalDecodingContext
 *ctx,
                                          ReorderBufferTXN *txn,
                                          int nrelations,
                                          Relation relations[],
                                          ReorderBufferChange *change);
```

######change_cb############ ######################TRUNCATE################ ###################################### ####TRUNCATE#############

47.6.4.7.

######filter_by_origin_cb#######*origin_id*################################ ############

```
typedef bool (*LogicalDecodeFilterByOriginCB) (struct
 LogicalDecodingContext *ctx,
                                                RepOriginId origin_id);
```

ctx######################### ############### ##############################true# ##### ######################### ##################################

##

47.6.4.8.

######message_cb##

```
typedef void (*LogicalDecodeMessageCB) (struct LogicalDecodingContext
 *ctx,
                                         ReorderBufferTXN *txn,
                                         XLogRecPtr message_lsn,
                                         bool transactional,
                                         const char *prefix,
                                         Size message_size,
                                         const char *message);
```

txn####################XID############################ ######################### ######################XID############NULL############## *lsn*###########WAL ###### *transactional*################################## #################(## #########)## ################################## ## *prefix*#null## ####*message*######## ##*message_size*################

##############################

47.6.4.9.

######filter_prepare_cb##########2#######################################
#COMMIT PREPARED#####1################################## #################
###true###### #######false###### #################false#######(##############
##2#########################2#########)#

```
typedef bool (*LogicalDecodeFilterPrepareCB) (struct
 LogicalDecodingContext *ctx,
                                              TransactionId xid,
                                              const char *gid);
```

ctx#################### #####*xid*#*gid*#################2############## #
#COMMIT PREPARED###ROLLBACK PREPARED##################################

##*xid*#*gid*######################
#########

47.6.4.10.

#####begin_prepare_cb## *txn*#########
##*gid*##########################PREPARE###################### ##############
####################

```
typedef void (*LogicalDecodeBeginPrepareCB) (struct LogicalDecodingContext
 *ctx,
                                             ReorderBufferTXN *txn);
```

47.6.4.11.

#####prepare_cb########2#################################### ##############
##########change_cb################### *txn*##########*gid*##################
####

```
typedef void (*LogicalDecodePrepareCB) (struct LogicalDecodingContext
 *ctx,
                                        ReorderBufferTXN *txn,
                                        XLogRecPtr prepare_lsn);
```

47.6.4.12.

#####commit_prepared_cb###############COMMIT PREPARED##################
txn##########*gid*#####################

```
typedef void (*LogicalDecodeCommitPreparedCB) (struct
 LogicalDecodingContext *ctx,
                                               ReorderBufferTXN *txn,
                                               XLogRecPtr commit_lsn);
```

47.6.4.13.

#####rollback_prepared_cb###############ROLLBACK PREPARED##############
txn##########*gid*#####################
#*prepare_end_lsn*#*prepare_time*#########PREPARE TRANSACTION############
############ ######################## ######################### *gid*##########
##

```
typedef void (*LogicalDecodeRollbackPreparedCB) (struct
 LogicalDecodingContext *ctx,
                                                 ReorderBufferTXN *txn,
                                                 XLogRecPtr
 prepare_end_lsn,
                                                 TimestampTz
 prepare_time);
```

47.6.4.14.

```
#####stream_start_cb##################################################
```

```
typedef void (*LogicalDecodeStreamStartCB) (struct LogicalDecodingContext
 *ctx,
                                             ReorderBufferTXN *txn);
```

47.6.4.15.

```
#####stream_stop_cb##################################################
```

```
typedef void (*LogicalDecodeStreamStopCB) (struct LogicalDecodingContext
 *ctx,
                                            ReorderBufferTXN *txn);
```

47.6.4.16.

```
#####stream_abort_cb##########################################
```

```
typedef void (*LogicalDecodeStreamAbortCB) (struct LogicalDecodingContext
 *ctx,
                                             ReorderBufferTXN *txn,
                                             XLogRecPtr abort_lsn);
```

47.6.4.17.

```
stream_prepare_cb#######2#############################################  ##
################################2#########################
```

```
typedef void (*LogicalDecodeStreamPrepareCB) (struct
 LogicalDecodingContext *ctx,
                                               ReorderBufferTXN *txn,
                                               XLogRecPtr prepare_lsn);
```

47.6.4.18.

```
#####stream_commit_cb##############################################
```

```
typedef void (*LogicalDecodeStreamCommitCB) (struct LogicalDecodingContext
 *ctx,
                                              ReorderBufferTXN *txn,
                                              XLogRecPtr commit_lsn);
```

47.6.4.19.

```
#####stream_change_cb######################
#stream_start_cb#stream_stop_cb############################## ############
## ##############################################################
```

```
typedef void (*LogicalDecodeStreamChangeCB) (struct LogicalDecodingContext
 *ctx,
                                              ReorderBufferTXN *txn,
                                              Relation relation,
                                              ReorderBufferChange *change);
```

47.6.4.20.

```
######stream_message_cb######################
#stream_start_cb#stream_stop_cb################################  ########
#################### ##########################################################
######
```

```
typedef void (*LogicalDecodeStreamMessageCB) (struct
 LogicalDecodingContext *ctx,
                                               ReorderBufferTXN *txn,
                                               XLogRecPtr message_lsn,
                                               bool transactional,
                                               const char *prefix,
                                               Size message_size,
                                               const char *message);
```

47.6.4.21. #####TRUNCATE######

```
######stream_truncate_cb######################
#stream_start_cb#stream_stop_cb#############TRUNCATE###############
```

```
typedef void (*LogicalDecodeStreamTruncateCB) (struct
 LogicalDecodingContext *ctx,
                                                ReorderBufferTXN *txn,
                                                int nrelations,
                                                Relation relations[],
                                                ReorderBufferChange
 *change);
```

```
######stream_change_cb############## ######################TRUNCATE########
################################################## ###TRUNCATE#############
```

47.6.5.

```
begin_cb#commit_cb#change_cb##############################ctx-
>out#StringInfo#############                                  ###########
##OutputPluginPrepareWrite(ctx, last_write)####### ###################
##OutputPluginWrite(ctx, last_write)################### last_write####
#####################################
```

```
########################################
```

```
OutputPluginPrepareWrite(ctx, true);
appendStringInfo(ctx->out, "BEGIN %u", txn->xid);
OutputPluginWrite(ctx, true);
```

47.7.

```
##############################         src/backend/replication/logical/
logicalfuncs.c######### #####3################ WAL#######################
################ (47.6.5##)#
```

47.8.

47.8.1.

##
######## ############################47.3###################### ###############
############################(F)#################53.4####

> ##
>
> *synchronous_standby_names*##
> ############################ ##
> ######################

47.8.2.

#######################[user]## #########
###########47.6.2######### ## ####
##########[user]############################## ##############

- ########pg_class#######LOCK####
- #########pg_class####CLUSTER####
- pg_class##LOCK######PREPARE TRANSACTION#####2#########################
  ##
- pg_trigger##CLUSTER######PREPARE TRANSACTION#####2###################
  ####### #####################################
- ########[user]###########TRUNCATE####

Note that these commands can cause deadlocks not only for the system catalog tables listed above but for other catalog tables.

47.9.

#################(begin_cb#change_cb#commit_cb#message_cb##)##############
################## ##(############
###########)#

##
###########(################################)############## ##################
####################################

##
######## ##################
(stream_start_cb#stream_stop_cb#stream_abort_cb#stream_commit_cb##
#stream_change_cb)#2#############(stream_message_cb##
#stream_truncate_cb)##### 2##
#47.10##########

##########################(########)#stream_start_cb###stream_stop_cb#####
####################### ############################stream_commit_cb######
############(###stream_abort_cb##################)# 2######################

####stream_prepare_cb################ COMMIT PREPARED#commit_prepared_cb################ ###rollback_prepared_cb### ##########

1##

```
stream_start_cb(...);   <-- start of first block of changes
  stream_change_cb(...);
  stream_change_cb(...);
  stream_message_cb(...);
  stream_change_cb(...);
  ...
  stream_change_cb(...);
stream_stop_cb(...);    <-- end of first block of changes

stream_start_cb(...);   <-- start of second block of changes
  stream_change_cb(...);
  stream_change_cb(...);
  stream_change_cb(...);
  ...
  stream_message_cb(...);
  stream_change_cb(...);
stream_stop_cb(...);    <-- end of second block of changes


[a. when using normal commit]
stream_commit_cb(...);    <-- commit of the streamed transaction

[b. when using two-phase commit]
stream_prepare_cb(...);   <-- prepare the streamed transaction
commit_prepared_cb(...);  <-- commit of the prepared transaction
```

#################################### ##
###############

#######################WAL##############(##################)
#logical_decoding_work_mem####################### ######################
(############################)################ ############################
############################(##)#####
####################

##

47.10. ################2#####

################(####begin_cb#change_cb#commit_cb#message_cb)###PREPARE TRANSACTION#COMMIT PREPARED#ROLLBACK PREPARED####2################## PREPARE TRANSACTION########COMMIT PREPARED#COMMIT#########ROLLBACK PREPARED#ROLLBACK###########

2### #####2##########
(begin_prepare_cb#prepare_cb#commit_prepared_cb#rollback_prepared_cb#stream_prepa
############(filter_prepare_cb)######

2###PREPARE TRANSACTION#############
####################prepare_cb############## ##############################
######################## #################begin_prepare_cb###############

#############ROLLBACK PREPARED###############rollback_prepared_cb######
######## #############COMMIT PREPARED############commit_prepared_cb###
###########

##############filter_prepare_cb##############################2############ ###### ###*gid*##############*xid*################

##

- ##############[user]###
  ##
- ###########2##[user]################
  ######################## ##(######
  #LOCK####)################ #######47.8.2##########

#48#

##########(replication origins)##
###2#######################

- #######################
- ##

##############ID######## ########################text### ##################
###
ID##

##########pg_replication_origin_create()###
##pg_replication_origin_drop()#####pg_replication_origin#############
####

###(##: ###)###########
#######################
##

###
(pg_replication_origin_session_setup()#####) ###
pg_replication_origin_xact_setup()################################LSN##
######################### ## ########
##############pg_replication_origin_status########## ###################
###########pg_replication_origin_progress()####### ###################
##pg_replication_origin_session_progress()######

###
########################### ## #######
################################### ##########################(47.6##)########
##############################
################################### ##
#filter_by_origin_cb################ ######################################
#########

#49#

PostgreSQL##25.3##### ##########
#archive_command###

####archive_library##########PostgreSQL#####WAL################ #############
#######################WAL################# #WAL######################
##25.3.1##################

#######################49.1###############49.2################### ##############
########GUC##############################

contrib/basic_archive####################################

49.1.

###########archive_library## #####
####################### ##
##_PG_archive_module_init################### ############################
#############ArchiveModuleCallbacks#################### ################
#########################static const####################

```
typedef struct ArchiveModuleCallbacks
{
    ArchiveStartupCB startup_cb;
    ArchiveCheckConfiguredCB check_configured_cb;
    ArchiveFileCB archive_file_cb;
    ArchiveShutdownCB shutdown_cb;
} ArchiveModuleCallbacks;
typedef const ArchiveModuleCallbacks *(*ArchiveModuleInit) (void);
```

archive_file_cb############# ###########

49.2.

################################## ########WAL###########################

49.2.1.

startup_cb############################## ################################ ##
#################state->private_data###########

```
typedef void (*ArchiveStartupCB) (ArchiveModuleState *state);
```

49.2.2.

check_configured_cb#######################WAL##########################
check_configured_cb#########
##########################

```
typedef bool (*ArchiveCheckConfiguredCB) (ArchiveModuleState *state);
```

true###########archive_file_cb########################### false#########
##################################:

```
WARNING:  archive_mode enabled, yet archiving is not configured
```

```
#######################true####################
```

```
##

##############################################   ######arch_module_
####_errdetail#################### errdetail()####################
######################## ##############DETAIL######### ###########
```

49.2.3.

archive_file_cb##########WAL#####################

```
typedef bool (*ArchiveFileCB) (ArchiveModuleState *state, const char
 *file, const char *path);
```

#####[89.655172]#true#################################### ######WAL#######
################### false#############WAL######################## *file*######
###WAL################*path*##WAL######################### ################
##########################include######WAL######################### #########
##########################WAL###################### ################WAL####
#########################WAL######### ####################

```
##

###########_####_cb############################################ ######
##########################_cb######################
```

49.2.4.

shutdown_cb#####################################archive_library##############
shutdown_cb################################
############################

```
typedef void (*ArchiveShutdownCB) (ArchiveModuleState *state);
```

VI.

PostgreSQL############ ###################################

##############man###############

##

I. SQL#### 1347
ABORT 1351
ALTER AGGREGATE 1352
ALTER COLLATION 1354
ALTER CONVERSION 1356
ALTER DATABASE 1357
ALTER DEFAULT PRIVILEGES 1359
ALTER DOMAIN 1362
ALTER EVENT TRIGGER 1365
ALTER EXTENSION 1366
ALTER FOREIGN DATA WRAPPER 1369
ALTER FOREIGN TABLE 1371
ALTER FUNCTION 1376
ALTER GROUP 1379
ALTER INDEX 1381
ALTER LANGUAGE 1384
ALTER LARGE OBJECT 1385
ALTER MATERIALIZED VIEW 1386
ALTER OPERATOR 1388
ALTER OPERATOR CLASS 1390
ALTER OPERATOR FAMILY 1391
ALTER POLICY 1394
ALTER PROCEDURE 1395
ALTER PUBLICATION 1398
ALTER ROLE 1401
ALTER ROUTINE 1405
ALTER RULE 1406
ALTER SCHEMA 1407
ALTER SEQUENCE 1408
ALTER SERVER 1411
ALTER STATISTICS 1413
ALTER SUBSCRIPTION 1414
ALTER SYSTEM 1417
ALTER TABLE 1419
ALTER TABLESPACE 1436
ALTER TEXT SEARCH CONFIGURATION 1438
ALTER TEXT SEARCH DICTIONARY 1440
ALTER TEXT SEARCH PARSER 1442
ALTER TEXT SEARCH TEMPLATE 1443
ALTER TRIGGER 1444
ALTER TYPE 1446
ALTER USER 1450
ALTER USER MAPPING 1451
ALTER VIEW 1452
ANALYZE 1454
BEGIN 1457
CALL 1459
CHECKPOINT 1460
CLOSE 1461
CLUSTER 1462
COMMENT 1465
COMMIT 1469
COMMIT PREPARED 1470
COPY 1471
CREATE ACCESS METHOD 1480

CREATE AGGREGATE 1481
CREATE CAST 1487
CREATE COLLATION 1491
CREATE CONVERSION 1494
CREATE DATABASE 1496
CREATE DOMAIN 1500
CREATE EVENT TRIGGER 1503
CREATE EXTENSION 1505
CREATE FOREIGN DATA WRAPPER 1507
CREATE FOREIGN TABLE 1509
CREATE FUNCTION 1513
CREATE GROUP 1521
CREATE INDEX 1522
CREATE LANGUAGE 1530
CREATE MATERIALIZED VIEW 1532
CREATE OPERATOR 1534
CREATE OPERATOR CLASS 1537
CREATE OPERATOR FAMILY 1540
CREATE POLICY 1541
CREATE PROCEDURE 1546
CREATE PUBLICATION 1549
CREATE ROLE 1553
CREATE RULE 1557
CREATE SCHEMA 1560
CREATE SEQUENCE 1562
CREATE SERVER 1566
CREATE STATISTICS 1568
CREATE SUBSCRIPTION 1572
CREATE TABLE 1577
CREATE TABLE AS 1596
CREATE TABLESPACE 1599
CREATE TEXT SEARCH CONFIGURATION 1601
CREATE TEXT SEARCH DICTIONARY 1602
CREATE TEXT SEARCH PARSER 1604
CREATE TEXT SEARCH TEMPLATE 1606
CREATE TRANSFORM 1607
CREATE TRIGGER 1609
CREATE TYPE 1615
CREATE USER 1623
CREATE USER MAPPING 1624
CREATE VIEW 1625
DEALLOCATE 1630
DECLARE 1631
DELETE 1634
DISCARD 1637
DO 1638
DROP ACCESS METHOD 1640
DROP AGGREGATE 1641
DROP CAST 1643
DROP COLLATION 1644
DROP CONVERSION 1645
DROP DATABASE 1646
DROP DOMAIN 1647
DROP EVENT TRIGGER 1648
DROP EXTENSION 1649
DROP FOREIGN DATA WRAPPER 1650
DROP FOREIGN TABLE 1651
DROP FUNCTION 1652

DROP GROUP 1654
DROP INDEX 1655
DROP LANGUAGE 1657
DROP MATERIALIZED VIEW 1658
DROP OPERATOR 1659
DROP OPERATOR CLASS 1661
DROP OPERATOR FAMILY 1663
DROP OWNED 1664
DROP POLICY 1665
DROP PROCEDURE 1666
DROP PUBLICATION 1668
DROP ROLE 1669
DROP ROUTINE 1670
DROP RULE 1671
DROP SCHEMA 1672
DROP SEQUENCE 1673
DROP SERVER 1674
DROP STATISTICS 1675
DROP SUBSCRIPTION 1676
DROP TABLE 1678
DROP TABLESPACE 1679
DROP TEXT SEARCH CONFIGURATION 1680
DROP TEXT SEARCH DICTIONARY 1681
DROP TEXT SEARCH PARSER 1682
DROP TEXT SEARCH TEMPLATE 1683
DROP TRANSFORM 1684
DROP TRIGGER 1685
DROP TYPE 1686
DROP USER 1687
DROP USER MAPPING 1688
DROP VIEW 1689
END 1690
EXECUTE 1691
EXPLAIN 1692
FETCH 1698
GRANT 1702
IMPORT FOREIGN SCHEMA 1707
INSERT 1709
LISTEN 1715
LOAD 1717
LOCK 1718
MERGE 1721
MOVE 1727
NOTIFY 1729
PREPARE 1731
PREPARE TRANSACTION 1733
REASSIGN OWNED 1735
REFRESH MATERIALIZED VIEW 1736
REINDEX 1738
RELEASE SAVEPOINT 1743
RESET 1745
REVOKE 1746
ROLLBACK 1750
ROLLBACK PREPARED 1751
ROLLBACK TO SAVEPOINT 1752
SAVEPOINT 1754
SECURITY LABEL 1756
SELECT 1759

SELECT INTO 1777
SET 1779
SET CONSTRAINTS 1782
SET ROLE 1783
SET SESSION AUTHORIZATION 1785
SET TRANSACTION 1787
SHOW 1790
START TRANSACTION 1792
TRUNCATE 1793
UNLISTEN 1795
UPDATE 1797
VACUUM 1802
VALUES 1806
II. PostgreSQL############# 1808
clusterdb 1809
createdb 1812
createuser 1815
dropdb 1819
dropuser 1822
ecpg 1824
pg_amcheck 1826
pg_basebackup 1831
pgbench 1839
pg_combinebackup 1861
pg_config 1864
pg_dump 1867
pg_dumpall 1880
pg_isready 1886
pg_receivewal 1888
pg_recvlogical 1892
pg_restore 1896
pg_verifybackup 1905
psql 1908
reindexdb 1945
vacuumdb 1948
III. PostgreSQL########### 1953
initdb 1954
pg_archivecleanup 1959
pg_checksums 1961
pg_controldata 1963
pg_createsubscriber 1964
pg_ctl 1968
pg_resetwal 1973
pg_rewind 1977
pg_test_fsync 1981
pg_test_timing 1982
pg_upgrade 1985
pg_waldump 1994
pg_walsummary 1997
postgres 1998

SQL####

####PostgreSQL########SQL################## #SQL###

##

ABORT 1351
ALTER AGGREGATE 1352
ALTER COLLATION 1354
ALTER CONVERSION 1356
ALTER DATABASE 1357
ALTER DEFAULT PRIVILEGES 1359
ALTER DOMAIN 1362
ALTER EVENT TRIGGER 1365
ALTER EXTENSION 1366
ALTER FOREIGN DATA WRAPPER 1369
ALTER FOREIGN TABLE 1371
ALTER FUNCTION 1376
ALTER GROUP 1379
ALTER INDEX 1381
ALTER LANGUAGE 1384
ALTER LARGE OBJECT 1385
ALTER MATERIALIZED VIEW 1386
ALTER OPERATOR 1388
ALTER OPERATOR CLASS 1390
ALTER OPERATOR FAMILY 1391
ALTER POLICY 1394
ALTER PROCEDURE 1395
ALTER PUBLICATION 1398
ALTER ROLE 1401
ALTER ROUTINE 1405
ALTER RULE 1406
ALTER SCHEMA 1407
ALTER SEQUENCE 1408
ALTER SERVER 1411
ALTER STATISTICS 1413
ALTER SUBSCRIPTION 1414
ALTER SYSTEM 1417
ALTER TABLE 1419
ALTER TABLESPACE 1436
ALTER TEXT SEARCH CONFIGURATION 1438
ALTER TEXT SEARCH DICTIONARY 1440
ALTER TEXT SEARCH PARSER 1442
ALTER TEXT SEARCH TEMPLATE 1443
ALTER TRIGGER 1444
ALTER TYPE 1446
ALTER USER 1450
ALTER USER MAPPING 1451
ALTER VIEW 1452
ANALYZE 1454
BEGIN 1457
CALL 1459
CHECKPOINT 1460
CLOSE 1461
CLUSTER 1462

COMMENT 1465
COMMIT 1469
COMMIT PREPARED 1470
COPY 1471
CREATE ACCESS METHOD 1480
CREATE AGGREGATE 1481
CREATE CAST 1487
CREATE COLLATION 1491
CREATE CONVERSION 1494
CREATE DATABASE 1496
CREATE DOMAIN 1500
CREATE EVENT TRIGGER 1503
CREATE EXTENSION 1505
CREATE FOREIGN DATA WRAPPER 1507
CREATE FOREIGN TABLE 1509
CREATE FUNCTION 1513
CREATE GROUP 1521
CREATE INDEX 1522
CREATE LANGUAGE 1530
CREATE MATERIALIZED VIEW 1532
CREATE OPERATOR 1534
CREATE OPERATOR CLASS 1537
CREATE OPERATOR FAMILY 1540
CREATE POLICY 1541
CREATE PROCEDURE 1546
CREATE PUBLICATION 1549
CREATE ROLE 1553
CREATE RULE 1557
CREATE SCHEMA 1560
CREATE SEQUENCE 1562
CREATE SERVER 1566
CREATE STATISTICS 1568
CREATE SUBSCRIPTION 1572
CREATE TABLE 1577
CREATE TABLE AS 1596
CREATE TABLESPACE 1599
CREATE TEXT SEARCH CONFIGURATION 1601
CREATE TEXT SEARCH DICTIONARY 1602
CREATE TEXT SEARCH PARSER 1604
CREATE TEXT SEARCH TEMPLATE 1606
CREATE TRANSFORM 1607
CREATE TRIGGER 1609
CREATE TYPE 1615
CREATE USER 1623
CREATE USER MAPPING 1624
CREATE VIEW 1625
DEALLOCATE 1630
DECLARE 1631
DELETE 1634
DISCARD 1637
DO 1638
DROP ACCESS METHOD 1640
DROP AGGREGATE 1641
DROP CAST 1643
DROP COLLATION 1644
DROP CONVERSION 1645
DROP DATABASE 1646
DROP DOMAIN 1647

DROP EVENT TRIGGER 1648
DROP EXTENSION 1649
DROP FOREIGN DATA WRAPPER 1650
DROP FOREIGN TABLE 1651
DROP FUNCTION 1652
DROP GROUP 1654
DROP INDEX 1655
DROP LANGUAGE 1657
DROP MATERIALIZED VIEW 1658
DROP OPERATOR 1659
DROP OPERATOR CLASS 1661
DROP OPERATOR FAMILY 1663
DROP OWNED 1664
DROP POLICY 1665
DROP PROCEDURE 1666
DROP PUBLICATION 1668
DROP ROLE 1669
DROP ROUTINE 1670
DROP RULE 1671
DROP SCHEMA 1672
DROP SEQUENCE 1673
DROP SERVER 1674
DROP STATISTICS 1675
DROP SUBSCRIPTION 1676
DROP TABLE 1678
DROP TABLESPACE 1679
DROP TEXT SEARCH CONFIGURATION 1680
DROP TEXT SEARCH DICTIONARY 1681
DROP TEXT SEARCH PARSER 1682
DROP TEXT SEARCH TEMPLATE 1683
DROP TRANSFORM 1684
DROP TRIGGER 1685
DROP TYPE 1686
DROP USER 1687
DROP USER MAPPING 1688
DROP VIEW 1689
END 1690
EXECUTE 1691
EXPLAIN 1692
FETCH 1698
GRANT 1702
IMPORT FOREIGN SCHEMA 1707
INSERT 1709
LISTEN 1715
LOAD 1717
LOCK 1718
MERGE 1721
MOVE 1727
NOTIFY 1729
PREPARE 1731
PREPARE TRANSACTION 1733
REASSIGN OWNED 1735
REFRESH MATERIALIZED VIEW 1736
REINDEX 1738
RELEASE SAVEPOINT 1743
RESET 1745
REVOKE 1746
ROLLBACK 1750

ROLLBACK PREPARED 1751
ROLLBACK TO SAVEPOINT 1752
SAVEPOINT 1754
SECURITY LABEL 1756
SELECT 1759
SELECT INTO 1777
SET 1779
SET CONSTRAINTS 1782
SET ROLE 1783
SET SESSION AUTHORIZATION 1785
SET TRANSACTION 1787
SHOW 1790
START TRANSACTION 1792
TRUNCATE 1793
UNLISTEN 1795
UPDATE 1797
VACUUM 1802
VALUES 1806

ABORT

ABORT — ################

##

```
ABORT [ WORK | TRANSACTION ] [ AND [ NO ] CHAIN ]
```

##

ABORT## ############SQL #ROLLBACK#############################

#####

WORK
TRANSACTION

  #######################

AND CHAIN

  AND CHAIN##(SET TRANSACTION# #######)########### ###########################

##

###################COMMIT#########

################ABORT##############################

#

#############

```
ABORT;
```

###

#######PostgreSQL################# ROLLBACK##############SQL#######

####

BEGIN, COMMIT, ROLLBACK

ALTER AGGREGATE

ALTER AGGREGATE — ##########

##

```
ALTER AGGREGATE name ( aggregate_signature ) RENAME TO new_name
ALTER AGGREGATE name ( aggregate_signature )
                OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER AGGREGATE name ( aggregate_signature ) SET SCHEMA new_schema
```

###aggregate_signature#########

```
* |
[ argmode ] [ argname ] argtype [ , ... ] |
[ [ argmode ] [ argname ] argtype [ , ... ] ] ORDER BY [ argmode ]
 [ argname ] argtype [ , ... ]
```

##

ALTER AGGREGATE##############

ALTER AGGREGATE######################### ###########################CREATE# ####### ########################SET ROLE################################# #CREATE############## ## ################################

#####

name

########################

argmode

########IN####VARIADIC### #############IN###

argname

######## ALTER AGGREGATE########################### ###################### *##################*

argtype

################## ###################################### ##################* *#############ORDER BY#########*

new_name

#############

new_owner

##############

new_schema

##############

##

#####################CREATE AGGREGATE#######################ORDER BY#### #### ####ORDER BY################1################ VARIADIC "any"######## ##############################VARIADIC "any"#1##########

#

integer###myavg#######my_average######

```
ALTER AGGREGATE myavg(integer) RENAME TO my_average;
```

integer###myavg########joe######

```
ALTER AGGREGATE myavg(integer) OWNER TO joe;
```

#####float8######integer#######mypercentile#myschema##########

```
ALTER AGGREGATE mypercentile(float8 ORDER BY integer) SET SCHEMA myschema;
```

########

```
ALTER AGGREGATE mypercentile(float8, integer) SET SCHEMA myschema;
```

###

##SQL##ALTER AGGREGATE#######

####

CREATE AGGREGATE, DROP AGGREGATE

ALTER COLLATION

ALTER COLLATION — ###########

##

```
ALTER COLLATION name REFRESH VERSION

ALTER COLLATION name RENAME TO new_name
ALTER COLLATION name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER COLLATION name SET SCHEMA new_schema
```

##

ALTER COLLATION##############

ALTER COLLATION########################### ##########################SET ROLE##################################CREATE############# ################## ############################## ####################################

#####

name

###################

new_name

############

new_owner

#############

new_schema

##############

REFRESH VERSION

################ ##############

##

##################################

```
WARNING:  collation "xx-x-icu" has version mismatch
DETAIL:  The collation in the database was created using version 1.2.3.4,
 but the operating system provides version 2.3.4.5.
HINT:  Rebuild all objects affected by this collation and run ALTER
 COLLATION pg_catalog."xx-x-icu" REFRESH VERSION, or build PostgreSQL with
 the right library version.
```

##pg_upgrade#########

####ICU## ###################
##############REINDEX########## #######################ALTER COLLATION ...
REFRESH VERSION########## ## #
###

libc##############GNU C#####(#####Linux####)#FreeBSD#Windows################
####### ICU#######################ICU################################

> ##
>
> #####GNU C##########C########################### ###Linux##############C
> ##C####################
> ##################
>
> #####Windows###############en-US####BCP 47#######################

############################ALTER DATABASE ... REFRESH COLLATION VERSION######

###

```
SELECT pg_describe_object(refclassid, refobjid, refobjsubid) AS
 "Collation",
       pg_describe_object(classid, objid, objsubid) AS "Object"
  FROM pg_depend d JOIN pg_collation c
       ON refclassid = 'pg_collation'::regclass AND refobjid = c.oid
  WHERE c.collversion <> pg_collation_actual_version(c.oid)
  ORDER BY 1, 2;
```

# #

```
####de_DE####german#######
```

```
ALTER COLLATION "de_DE" RENAME TO german;
```

```
####en_US#####joe#######
```

```
ALTER COLLATION "en_US" OWNER TO joe;
```

###

```
##SQL##ALTER COLLATION########
```

####

CREATE COLLATION, DROP COLLATION

ALTER CONVERSION

ALTER CONVERSION — #########

##

```
ALTER CONVERSION name RENAME TO new_name
ALTER CONVERSION name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER CONVERSION name SET SCHEMA new_schema
```

##

ALTER CONVERSION############

ALTER CONVERSION######################## ########################SET ROLE### ##############################CREATE############## ########################### ######################### ##############################

#####

name

###################

new_name

##########

new_owner

###########

new_schema

############

#

##iso_8859_1_to_utf8####latin1_to_unicode######

```
ALTER CONVERSION iso_8859_1_to_utf8 RENAME TO latin1_to_unicode;
```

##iso_8859_1_to_utf8#####joe######

```
ALTER CONVERSION iso_8859_1_to_utf8 OWNER TO joe;
```

###

##SQL##ALTER CONVERSION#######

####

CREATE CONVERSION, DROP CONVERSION

ALTER DATABASE

ALTER DATABASE — ##########

##

```
ALTER DATABASE name [ [ WITH ] option [ ... ] ]
```

```
###option#########

    ALLOW_CONNECTIONS allowconn
    CONNECTION LIMIT connlimit
    IS_TEMPLATE istemplate

ALTER DATABASE name RENAME TO new_name

ALTER DATABASE name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }

ALTER DATABASE name SET TABLESPACE new_tablespace

ALTER DATABASE name REFRESH COLLATION VERSION

ALTER DATABASE name SET configuration_parameter { TO | = } { value |
 DEFAULT }
ALTER DATABASE name SET configuration_parameter FROM CURRENT
ALTER DATABASE name RESET configuration_parameter
ALTER DATABASE name RESET ALL
```

##

ALTER DATABASE################

###################### ########## ###############################

2###################### ### ############
#######CREATEDB################# ##
######

3####################### #######################SET ROLE##############
##CREATEDB############## ####################################

4################################ ################################## #########
##################### ##
########## ## ######
################################

#######PostgreSQL################################## ######################
########################## ################postgresql.conf#############
#postgres######################## ##
########### ######################################

#####

name

    ####################

allowconn

false#######################

connlimit

################## -1##########

istemplate

true####CREATEDB############################ false##########################
###################

new_name

############

new_owner

###############

new_tablespace

########################

#######################################

REFRESH COLLATION VERSION

######################## ##################

configuration_parameter
value

value#DEFAULT##########
#RESET## ########
##############RESET ALL######### SET FROM CURRENT####################
##################

#########################SET####19###########

##

ALTER ROLE##########
##########################

# #

######test#########################

```
ALTER DATABASE test SET enable_indexscan TO off;
```

###

ALTER DATABASE##PostgreSQL######

####

CREATE DATABASE, DROP DATABASE, SET, CREATE TABLESPACE

ALTER DEFAULT PRIVILEGES

ALTER DEFAULT PRIVILEGES — ################

##

```
ALTER DEFAULT PRIVILEGES
    [ FOR { ROLE | USER } target_role [, ...] ]
    [ IN SCHEMA schema_name [, ...] ]
    abbreviated_grant_or_revoke


###abbreviated_grant_or_revoke###########

GRANT { { SELECT | INSERT | UPDATE | DELETE | TRUNCATE | REFERENCES |
 TRIGGER | MAINTAIN }
    [, ...] | ALL [ PRIVILEGES ] }
    ON TABLES
    TO { [ GROUP ] role_name | PUBLIC } [, ...] [ WITH GRANT OPTION ]

GRANT { { USAGE | SELECT | UPDATE }
    [, ...] | ALL [ PRIVILEGES ] }
    ON SEQUENCES
    TO { [ GROUP ] role_name | PUBLIC } [, ...] [ WITH GRANT OPTION ]

GRANT { EXECUTE | ALL [ PRIVILEGES ] }
    ON { FUNCTIONS | ROUTINES }
    TO { [ GROUP ] role_name | PUBLIC } [, ...] [ WITH GRANT OPTION ]

GRANT { USAGE | ALL [ PRIVILEGES ] }
    ON TYPES
    TO { [ GROUP ] role_name | PUBLIC } [, ...] [ WITH GRANT OPTION ]

GRANT { { USAGE | CREATE }
    [, ...] | ALL [ PRIVILEGES ] }
    ON SCHEMAS
    TO { [ GROUP ] role_name | PUBLIC } [, ...] [ WITH GRANT OPTION ]

REVOKE [ GRANT OPTION FOR ]
    { { SELECT | INSERT | UPDATE | DELETE | TRUNCATE | REFERENCES |
 TRIGGER | MAINTAIN }
    [, ...] | ALL [ PRIVILEGES ] }
    ON TABLES
    FROM { [ GROUP ] role_name | PUBLIC } [, ...]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { USAGE | SELECT | UPDATE }
    [, ...] | ALL [ PRIVILEGES ] }
    ON SEQUENCES
    FROM { [ GROUP ] role_name | PUBLIC } [, ...]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { EXECUTE | ALL [ PRIVILEGES ] }
    ON { FUNCTIONS | ROUTINES }
    FROM { [ GROUP ] role_name | PUBLIC } [, ...]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
```

```
    { USAGE | ALL [ PRIVILEGES ] }
    ON TYPES
    FROM { [ GROUP ] role_name | PUBLIC } [, ...]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { USAGE | CREATE }
    [, ...] | ALL [ PRIVILEGES ] }
    ON SCHEMAS
    FROM { [ GROUP ] role_name | PUBLIC } [, ...]
    [ CASCADE | RESTRICT ]
```

##

ALTER DEFAULT PRIVILEGES################################### ###############
################ ##
#########################

###
##

5.8### ##
##PUBLIC#################### ###ALTER DEFAULT PRIVILEGES################
################

#################
######### FUNCTIONS#ROUTINES############## (ROUTINES#####################
########### PostgreSQL#########FUNCTIONS########### ########################
###########)

##############(###########
###########ALTER DEFAULT PRIVILEGES#########)############################
#######REVOKE###########GRANT################

#####

target_role

target_role###

schema_name

############ ## IN SCHEMA####
###################### ################################IN SCHEMA#########

role_name

######################## ##########*abbreviated_grant_or_revoke*########
GRANT#REVOKE###########
###

##

##################################psql#\ddp############## #########5.8#\dp#######
##

###DROP OWNED BY####
#########

# #

####myschema################################SELECT######## ######webuser##
###############

```
ALTER DEFAULT PRIVILEGES IN SCHEMA myschema GRANT SELECT ON TABLES TO
 PUBLIC;
ALTER DEFAULT PRIVILEGES IN SCHEMA myschema GRANT INSERT ON TABLES TO
 webuser;
```

####################################

```
ALTER DEFAULT PRIVILEGES IN SCHEMA myschema REVOKE SELECT ON TABLES FROM
 PUBLIC;
ALTER DEFAULT PRIVILEGES IN SCHEMA myschema REVOKE INSERT ON TABLES FROM
 webuser;
```

##admin#####################################EXECUTE##########

```
ALTER DEFAULT PRIVILEGES FOR ROLE admin REVOKE EXECUTE ON FUNCTIONS FROM
 PUBLIC;
```

###########1### ####GRANT###############
###########

```
ALTER DEFAULT PRIVILEGES IN SCHEMA public REVOKE EXECUTE ON FUNCTIONS FROM
 PUBLIC;
```

##

###

##SQL##ALTER DEFAULT PRIVILEGES########

####

GRANT, REVOKE

ALTER DOMAIN

ALTER DOMAIN — ##########

##

```
ALTER DOMAIN name
    { SET DEFAULT expression | DROP DEFAULT }
ALTER DOMAIN name
    { SET | DROP } NOT NULL
ALTER DOMAIN name
    ADD domain_constraint [ NOT VALID ]
ALTER DOMAIN name
    DROP CONSTRAINT [ IF EXISTS ] constraint_name [ RESTRICT | CASCADE ]
ALTER DOMAIN name
     RENAME CONSTRAINT constraint_name TO new_constraint_name
ALTER DOMAIN name
    VALIDATE CONSTRAINT constraint_name
ALTER DOMAIN name
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
ALTER DOMAIN name
    RENAME TO new_name
ALTER DOMAIN name
    SET SCHEMA new_schema
```

##

ALTER DOMAIN################ #############

SET/DROP DEFAULT

  ############################## ###############INSERT#####################
  ## ##############################

SET/DROP NOT NULL

  ##########NULL################### SET NOT NULL###################NULL
  ###############

ADD *domain_constraint* [NOT VALID]

  #####CREATE DOMAIN########################## ##########################
  ########################## NOT VALID##################################
  #### ALTER DOMAIN ... VALIDATE CONSTRAINT######################## ##
  #################NOT VALID########################### NOT VALID#CHECK#
  ############

DROP CONSTRAINT [IF EXISTS]

  ################### IF EXISTS########################### ###############
  #######

RENAME CONSTRAINT

  ####################

VALIDATE CONSTRAINT

  #########NOT VALID################ ######################################
  #####

OWNER

###########################

RENAME

##################

SET SCHEMA

#################### ############################

ALTER DOMAIN######################### ###########################CREATE### ########### ########################SET ROLE################################ ##CREATE############## ## ## ##################################

#####

name

##############################

domain_constraint

#################

constraint_name

####################

NOT VALID

############################

CASCADE

##5.15####

RESTRICT

############################## ##############

new_name

############

new_constraint_name

##########

new_owner

######################

new_schema

##############

##

ALTER DOMAIN ADD CONSTRAINT## ## ####################

###########NOT VALID###
##################################ALTER DOMAIN VALIDATE CONSTRAINT#######
##

######ALTER DOMAIN ADD CONSTRAINT#ALTER DOMAIN VALIDATE CONSTRAINT##
#ALTER DOMAIN SET NOT NULL###
############################## ##

# #

#####NOT NULL########

```
ALTER DOMAIN zipcode SET NOT NULL;
```

######NOT NULL########

```
ALTER DOMAIN zipcode DROP NOT NULL;
```

#####CHECK########

```
ALTER DOMAIN zipcode ADD CONSTRAINT zipchk CHECK (char_length(VALUE) = 5);
```

######CHECK########

```
ALTER DOMAIN zipcode DROP CONSTRAINT zipchk;
```

###################

```
ALTER DOMAIN zipcode RENAME CONSTRAINT zipchk TO zip_check;
```

##################

```
ALTER DOMAIN zipcode SET SCHEMA customers;
```

###

ALTER DOMAIN####SQL######## ####OWNER#RENAME#SET SCHEMA#VALIDATE CONSTRAINT#######PostgreSQL###### ADD CONSTRAINT###NOT VALID##PostgreSQL### ###

####

CREATE DOMAIN, DROP DOMAIN

ALTER EVENT TRIGGER

ALTER EVENT TRIGGER — ###############

##

```
ALTER EVENT TRIGGER name DISABLE
ALTER EVENT TRIGGER name ENABLE [ REPLICA | ALWAYS ]
ALTER EVENT TRIGGER name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
ALTER EVENT TRIGGER name RENAME TO new_name
```

##

ALTER EVENT TRIGGER###################

################################

#####

name

###############

new_owner

#########################

new_name

###############

DISABLE/ENABLE [REPLICA | ALWAYS]

######################## ##
session_replication_role##########

###

##SQL##ALTER EVENT TRIGGER#######

####

CREATE EVENT TRIGGER, DROP EVENT TRIGGER

ALTER EXTENSION

ALTER EXTENSION — #########

##

```
ALTER EXTENSION name UPDATE [ TO new_version ]
ALTER EXTENSION name SET SCHEMA new_schema
ALTER EXTENSION name ADD member_object
ALTER EXTENSION name DROP member_object
```

###member_object#########

```
  ACCESS METHOD object_name |
  AGGREGATE aggregate_name ( aggregate_signature ) |
  CAST (source_type AS target_type) |
  COLLATION object_name |
  CONVERSION object_name |
  DOMAIN object_name |
  EVENT TRIGGER object_name |
  FOREIGN DATA WRAPPER object_name |
  FOREIGN TABLE object_name |
  FUNCTION function_name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] |
  MATERIALIZED VIEW object_name |
  OPERATOR operator_name (left_type, right_type) |
  OPERATOR CLASS object_name USING index_method |
  OPERATOR FAMILY object_name USING index_method |
  [ PROCEDURAL ] LANGUAGE object_name |
  PROCEDURE procedure_name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] |
  ROUTINE routine_name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
 |
  SCHEMA object_name |
  SEQUENCE object_name |
  SERVER object_name |
  TABLE object_name |
  TEXT SEARCH CONFIGURATION object_name |
  TEXT SEARCH DICTIONARY object_name |
  TEXT SEARCH PARSER object_name |
  TEXT SEARCH TEMPLATE object_name |
  TRANSFORM FOR type_name LANGUAGE lang_name |
  TYPE object_name |
  VIEW object_name
```

##aggregate_signature#########

```
* |
[ argmode ] [ argname ] argtype [ , ... ] |
[ [ argmode ] [ argname ] argtype [ , ... ] ] ORDER BY [ argmode ]
 [ argname ] argtype [ , ... ]
```

##

ALTER EXTENSION#################### ############

UPDATE

###################### ##
#########################

SET SCHEMA

########################### ##################################

ADD *member_object*

######################## ################### ######################## #####
######################

DROP *member_object*

######################## ################### ############################
##

#############36.17#########

ALTER EXTENSION######################## ADD/DROP#########################
#################

#####

name

################

new_version

################## ################################ #######ALTER
EXTENSION UPDATE##

new_schema

############

object_name
aggregate_name
function_name
operator_name
procedure_name
routine_name

############################## ##
##

source_type

#################

target_type

#################

argmode

######################IN#OUT#INOUT#VARIADIC####### #########IN### ###
##########################ALTER EXTENSION#OUT################## ###
#IN#INOUT###VARIADIC##############

argname

###################### ##############################ALTER EXTENSION### ##################

argtype

#########################

left_type
right_type

##################### #################NONE#########

PROCEDURAL

###########

type_name

############

lang_name

##########

# #

hstore#######2.0#######

```
ALTER EXTENSION hstore UPDATE TO '2.0';
```

hstore#######utils######

```
ALTER EXTENSION hstore SET SCHEMA utils;
```

hstore##############

```
ALTER EXTENSION hstore ADD FUNCTION populate_record(anyelement, hstore);
```

###

ALTER EXTENSION#PostgreSQL######

####

CREATE EXTENSION, DROP EXTENSION

ALTER FOREIGN DATA WRAPPER

ALTER FOREIGN DATA WRAPPER — ################

##

```
ALTER FOREIGN DATA WRAPPER name
    [ HANDLER handler_function | NO HANDLER ]
    [ VALIDATOR validator_function | NO VALIDATOR ]
    [ OPTIONS ( [ ADD | SET | DROP ] option ['value'] [, ... ]) ]
ALTER FOREIGN DATA WRAPPER name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
ALTER FOREIGN DATA WRAPPER name RENAME TO new_name
```

##

ALTER FOREIGN DATA WRAPPER################### ########1#################### ################### ######1########## #2#######################

################################ ##################################

#####

name

################

HANDLER *handler_function*

##########################

NO HANDLER

####################################

##

VALIDATOR *validator_function*

########################

###
PostgreSQL########### ##
#####ALTER FOREIGN DATA WRAPPER################################

NO VALIDATOR

#######################################

OPTIONS ([ADD | SET | DROP] *option* ['*value*'] [, ...])

###################### ADD#SET#DROP############# ##############ADD######
################################

new_owner

#######################

new_name

##################

# #

#########dbi#####foo#########bar############

```
ALTER FOREIGN DATA WRAPPER dbi OPTIONS (ADD foo '1', DROP bar);
```

#########dbi######bob.myvalidator#######

```
ALTER FOREIGN DATA WRAPPER dbi VALIDATOR bob.myvalidator;
```

###

ALTER FOREIGN DATA WRAPPER#ISO/IEC 9075-9 (SQL/MED)###### #### HANDLER#VALIDATOR#OWNER TO#RENAME#######

####

CREATE FOREIGN DATA WRAPPER, DROP FOREIGN DATA WRAPPER

ALTER FOREIGN TABLE

ALTER FOREIGN TABLE — ##############

##

```
ALTER FOREIGN TABLE [ IF EXISTS ] [ ONLY ] name [ * ]
    action [, ... ]
ALTER FOREIGN TABLE [ IF EXISTS ] [ ONLY ] name [ * ]
    RENAME [ COLUMN ] column_name TO new_column_name
ALTER FOREIGN TABLE [ IF EXISTS ] name
    RENAME TO new_name
ALTER FOREIGN TABLE [ IF EXISTS ] name
    SET SCHEMA new_schema


###action###########

    ADD [ COLUMN ] column_name data_type [ COLLATE collation ]
 [ column_constraint [ ... ] ]
    DROP [ COLUMN ] [ IF EXISTS ] column_name [ RESTRICT | CASCADE ]
    ALTER [ COLUMN ] column_name [ SET DATA ] TYPE data_type
 [ COLLATE collation ]
    ALTER [ COLUMN ] column_name SET DEFAULT expression
    ALTER [ COLUMN ] column_name DROP DEFAULT
    ALTER [ COLUMN ] column_name { SET | DROP } NOT NULL
    ALTER [ COLUMN ] column_name SET STATISTICS integer
    ALTER [ COLUMN ] column_name SET ( attribute_option = value [, ... ] )
    ALTER [ COLUMN ] column_name RESET ( attribute_option [, ... ] )
    ALTER [ COLUMN ] column_name SET STORAGE { PLAIN | EXTERNAL | EXTENDED
 | MAIN | DEFAULT }
    ALTER [ COLUMN ] column_name OPTIONS ( [ ADD | SET | DROP ] option
 ['value'] [, ... ])
    ADD table_constraint [ NOT VALID ]
    VALIDATE CONSTRAINT constraint_name
    DROP CONSTRAINT [ IF EXISTS ]  constraint_name [ RESTRICT | CASCADE ]
    DISABLE TRIGGER [ trigger_name | ALL | USER ]
    ENABLE TRIGGER [ trigger_name | ALL | USER ]
    ENABLE REPLICA TRIGGER trigger_name
    ENABLE ALWAYS TRIGGER trigger_name
    SET WITHOUT OIDS
    INHERIT parent_table
    NO INHERIT parent_table
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
    OPTIONS ( [ ADD | SET | DROP ] option ['value'] [, ... ])
```

##

ALTER FOREIGN TABLE################### ##################

ADD COLUMN

######CREATE FOREIGN TABLE############################ ################ ######################## ##

DROP COLUMN [IF EXISTS]

#################### ############################CASCADE############ IF EXISTS########################### ######################

SET DATA TYPE

###################### ######################### ########PostgreSQL#########
############

SET/DROP DEFAULT

############################ ##############INSERT###UPDATE#############
########################

SET/DROP NOT NULL

##NULL##################

SET STATISTICS

##########ANALYZE######################## #######ALTER TABLE############
###

SET (*attribute_option* = *value* [, ...])
RESET (*attribute_option* [, ...])

############################ #######ALTER TABLE##################

SET STORAGE

###################### ####ALTER TABLE############## ##################
##

ADD *table_constraint* [NOT VALID]

######CREATE FOREIGN TABLE########################### #######CHECK####
############

########################
################################### #CREATE FOREIGN TABLE############ #
##NOT VALID###

VALIDATE CONSTRAINT

##########NOT VALID############valid####### ##############################
###############

DROP CONSTRAINT [IF EXISTS]

########################### IF EXISTS############################## ######
############

DISABLE/ENABLE [REPLICA | ALWAYS] TRIGGER

############################### #######ALTER TABLE##################

SET WITHOUT OIDS

#####oid################## #####oid########################

INHERIT *parent_table*

####ALTER TABLE##############

NO INHERIT *parent_table*

##

OWNER

###########################

OPTIONS ([ADD | SET | DROP] *option* ['*value*'] [, ...])

################################ ADD#SET#DROP############# ###########
#ADD####### ################# ## ##
####################################

RENAME

RENAME##################################

SET SCHEMA

########################

RENAME###SET SCHEMA#################################### ##################
##########################

#####ALTER FOREIGN TABLE IF EXISTS ...################################ #
#############

ALTER FOREIGN TABLE########################## ########################
########CREATE################ #######################SET ROLE##########
#######################CREATE############## ################################
################# ############################## ##############################
#USAGE#######

#####

name

########################## #######ONLY######################### ONLY##
################################
#*######################

column_name

################

new_column_name

###############

new_name

############

data_type

##############################

table_constraint

##################

constraint_name

##############

CASCADE

###
####5.15####

RESTRICT

################################## ##############

trigger_name

#######################

ALL

################################ (################################## ########
##)

USER

##

parent_table

##############################

new_owner

##################

new_schema

####################

##

COLUMN###################

ADD COLUMN###DROP COLUMN##############NOT NULL###CHECK#########SET DATA TYPE################################ ##############################

#####################CREATE FOREIGN TABLE#########

#

###NULL#######

```
ALTER FOREIGN TABLE distributors ALTER COLUMN street SET NOT NULL;
```

##################

```
ALTER FOREIGN TABLE myschema.distributors OPTIONS (ADD opt1 'value', SET
 opt2 'value2', DROP opt3);
```

###

ADD#DROP#SET DATA TYPE#####SQL###### ######SQL####PostgreSQL###### ## #ALTER FOREIGN TABLE######################

ALTER FOREIGN TABLE DROP COLUMN##############1########################### ########SQL###################

####

CREATE FOREIGN TABLE, DROP FOREIGN TABLE

ALTER FUNCTION

ALTER FUNCTION — ########

##

```
ALTER FUNCTION name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    action [ ... ] [ RESTRICT ]
ALTER FUNCTION name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    RENAME TO new_name
ALTER FUNCTION name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
ALTER FUNCTION name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    SET SCHEMA new_schema
ALTER FUNCTION name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    [ NO ] DEPENDS ON EXTENSION extension_name


####action###########

    CALLED ON NULL INPUT | RETURNS NULL ON NULL INPUT | STRICT
    IMMUTABLE | STABLE | VOLATILE
    [ NOT ] LEAKPROOF
    [ EXTERNAL ] SECURITY INVOKER | [ EXTERNAL ] SECURITY DEFINER
    PARALLEL { UNSAFE | RESTRICTED | SAFE }
    COST execution_cost
    ROWS result_rows
    SUPPORT support_function
    SET configuration_parameter { TO | = } { value | DEFAULT }
    SET configuration_parameter FROM CURRENT
    RESET configuration_parameter
    RESET ALL
```

##

ALTER FUNCTION###########

ALTER FUNCTION######################## ##########################CREATE###### ## ########################SET ROLE#################################CREATE#### ########### ## ################## ##################

#####

name

###################### ###################################

argmode

########IN#OUT#INOUT#VARIADIC####### #############IN### ############ ###############ALTER FUNCTION#OUT################# #####IN#INOUT## #VARIADIC##############

argname

######## ##############################ALTER FUNCTION#################### ##

argtype

###############################

new_name

#########

new_owner

############ ###SECURITY DEFINER###
###

new_schema

#############

DEPENDS ON EXTENSION *extension_name*
NO DEPENDS ON EXTENSION *extension_name*

#######################NO############################## ##################
##CASCADE############################## ##################################
################

CALLED ON NULL INPUT
RETURNS NULL ON NULL INPUT
STRICT

CALLED ON NULL INPUT#############NULL#################### RETURNS
NULL ON NULL INPUT####STRICT#######NULL#################### #######
#NULL########## ###CREATE FUNCTION#########

IMMUTABLE
STABLE
VOLATILE

################### #######CREATE FUNCTION#########

[EXTERNAL] SECURITY INVOKER
[EXTERNAL] SECURITY DEFINER

########################## EXTERNAL######SQL#################### #########
###CREATE FUNCTION#########

PARALLEL

################################## ####CREATE FUNCTION#########

LEAKPROOF

###################### ##############CREATE FUNCTION#########

COST *execution_cost*

################ #######CREATE FUNCTION#########

ROWS *result_rows*

###################### #######CREATE FUNCTION#########

SUPPORT *support_function*

################################## ###36.11######### ########################
#####

##########
##CREATE OR REPLACE FUNCTION########

configuration_parameter
value

#################################### *value*#DEFAULT##########RESET#######
################## ########################## ####################RESET
ALL######### SET FROM CURRENT##ALTER FUNCTION######################
#################

##########################SET####19###########

RESTRICT

##SQL#####################

# #

integer###sqrt######square_root#######

ALTER FUNCTION sqrt(integer) RENAME TO square_root;

integer###sqrt#######joe#######

ALTER FUNCTION sqrt(integer) OWNER TO joe;

integer###sqrt########maths#######

ALTER FUNCTION sqrt(integer) SET SCHEMA maths;

integer#######sqrt####mathlib#######################

ALTER FUNCTION sqrt(integer) DEPENDS ON EXTENSION mathlib;

#########################

ALTER FUNCTION check_password(text) SET search_path = admin, pg_temp;

####search_path############

ALTER FUNCTION check_password(text) RESET search_path;

################################

###

######SQL#ALTER FUNCTION############ ######################################
#######RESTRICT####
###########PostgreSQL########

####

CREATE FUNCTION, DROP FUNCTION, ALTER PROCEDURE, ALTER ROUTINE

ALTER GROUP

ALTER GROUP — ##################

##

```
ALTER GROUP role_specification ADD USER user_name [, ... ]
ALTER GROUP role_specification DROP USER user_name [, ... ]
```

```
###role_specification#########

    role_name
  | CURRENT_ROLE
  | CURRENT_USER
  | SESSION_USER
```

```
ALTER GROUP group_name RENAME TO new_name
```

##

ALTER GROUP################# ################################## ##############
############################

###2################################ ################################ ########
#####GRANT#REVOKE######
####### GRANT#REVOKE###ADMIN OPTION#######################################
######################

3################### ####ALTER ROLE#####################

#####

group_name

##################

user_name

######################### ######################### ALTER GROUP############
######

new_name

###########

#

###############

```
ALTER GROUP staff ADD USER karl, john;
```

################

```
ALTER GROUP workers DROP USER beth;
```

###

##SQL##ALTER GROUP########

####

GRANT, REVOKE, ALTER ROLE

ALTER INDEX

ALTER INDEX — ############

##

```
ALTER INDEX [ IF EXISTS ] name RENAME TO new_name
ALTER INDEX [ IF EXISTS ] name SET TABLESPACE tablespace_name
ALTER INDEX name ATTACH PARTITION index_name
ALTER INDEX name [ NO ] DEPENDS ON EXTENSION extension_name
ALTER INDEX [ IF EXISTS ] name SET ( storage_parameter [= value]
 [, ... ] )
ALTER INDEX [ IF EXISTS ] name RESET ( storage_parameter [, ... ] )
ALTER INDEX [ IF EXISTS ] name ALTER [ COLUMN ] column_number
    SET STATISTICS integer
ALTER INDEX ALL IN TABLESPACE name [ OWNED BY role_name [, ... ] ]
    SET TABLESPACE new_tablespace [ NOWAIT ]
```

##

ALTER INDEX################### ################ ############################## ######### #########ACCESS EXCLUSIVE######### ############################## ########################

RENAME

##RENAME################### #############UNIQUE#PRIMARY KEY#EXCLUDE## ############################ ################

#############SHARE UPDATE EXCLUSIVE########

SET TABLESPACE

##############################CREATE############### ALL IN TABLESPACE# ## ########################### ####1######### #####OWNED BY## NOWAIT## ################## ################ #######ALTER DATABASE########ALTER INDEX############## CREATE TABLESPACE#########

ATTACH PARTITION

############################ ## ################### ##

DEPENDS ON EXTENSION *extension_name*
NO DEPENDS ON EXTENSION *extension_name*

##########################NO############################## ################ ############################

SET (*storage_parameter* [= *value*] [, ...])

############################1################## ###############CREATE INDEX######### ## ############## ##########REINDEX################################

RESET (*storage_parameter* [, ...])

######1##################################### SET##################
#REINDEX#############

ALTER [COLUMN] *column_number* SET STATISTICS *integer*

#########ANALYZE## #######
################################ ###0##10000##############-1#################
####default_statistics_target###### PostgreSQL#########################14.2#######
###

#####

IF EXISTS

######################## #################

column_number

#########################

name

##############################

new_name

###############

tablespace_name

###################

extension_name

#######################

storage_parameter

#########################

value

########################### ################################

##

#######ALTER TABLE############## #####ALTER INDEX##ALTER TABLE##########
########

###ALTER INDEX OWNER################################## ####################
################# ##################################

##############################

# #

##################

```
ALTER INDEX distributors RENAME TO suppliers;

######################


ALTER INDEX distributors SET TABLESPACE fasttablespace;

#############################################################


ALTER INDEX distributors SET (fillfactor = 75);
REINDEX INDEX distributors;

########################


CREATE INDEX coord_idx ON measured (x, y, (z + t));
ALTER INDEX coord_idx ALTER COLUMN 3 SET STATISTICS 1000;
```

###

ALTER INDEX#PostgreSQL######

####

CREATE INDEX, REINDEX

ALTER LANGUAGE

ALTER LANGUAGE — ############

##

```
ALTER [ PROCEDURAL ] LANGUAGE name RENAME TO new_name
ALTER [ PROCEDURAL ] LANGUAGE name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
```

##

ALTER LANGUAGE################ ######################### ALTER LANGUAGE### ################################

#####

name

########

new_name

###########

new_owner

############

###

##SQL##ALTER LANGUAGE########

####

CREATE LANGUAGE, DROP LANGUAGE

ALTER LARGE OBJECT

ALTER LARGE OBJECT — ################

##

```
ALTER LARGE OBJECT large_object_oid OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
```

##

ALTER LARGE OBJECT##################

ALTER LARGE OBJECT################################ ######################
#SET ROLE############ (############################) ########################
################

#####

large_object_oid

###############OID###

new_owner

##################

###

##SQL##ALTER LARGE OBJECT#######

####

#33#

ALTER MATERIALIZED VIEW

ALTER MATERIALIZED VIEW — ##################

##

```
ALTER MATERIALIZED VIEW [ IF EXISTS ] name
    action [, ... ]
ALTER MATERIALIZED VIEW name
    [ NO ] DEPENDS ON EXTENSION extension_name
ALTER MATERIALIZED VIEW [ IF EXISTS ] name
    RENAME [ COLUMN ] column_name TO new_column_name
ALTER MATERIALIZED VIEW [ IF EXISTS ] name
    RENAME TO new_name
ALTER MATERIALIZED VIEW [ IF EXISTS ] name
    SET SCHEMA new_schema
ALTER MATERIALIZED VIEW ALL IN TABLESPACE name [ OWNED BY role_name
 [, ... ] ]
    SET TABLESPACE new_tablespace [ NOWAIT ]


###action###########

    ALTER [ COLUMN ] column_name SET STATISTICS integer
    ALTER [ COLUMN ] column_name SET ( attribute_option = value [, ... ] )
    ALTER [ COLUMN ] column_name RESET ( attribute_option [, ... ] )
    ALTER [ COLUMN ] column_name SET STORAGE { PLAIN | EXTERNAL | EXTENDED
 | MAIN | DEFAULT }
    ALTER [ COLUMN ] column_name SET COMPRESSION compression_method
    CLUSTER ON index_name
    SET WITHOUT CLUSTER
    SET ACCESS METHOD new_access_method
    SET TABLESPACE new_tablespace
    SET ( storage_parameter [= value] [, ... ] )
    RESET ( storage_parameter [, ... ] )
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
```

##

ALTER MATERIALIZED VIEW############################

ALTER MATERIALIZED VIEW#################################### ###############
#####################CREATE############## #######################SET ROLE##
##CREATE############## ################
################################
#######

ALTER MATERIALIZED VIEW##############ALTER TABLE#######################
####################### #######ALTER TABLE############

#####

name

   ############################

column_name

   #########

extension_name

###############(####NO##################)######## ######################### ##########################

new_column_name

###############

new_owner

############################

new_name

###################

new_schema

#####################

# #

##########foo####bar######

```
ALTER MATERIALIZED VIEW foo RENAME TO bar;
```

###

ALTER MATERIALIZED VIEW#PostgreSQL######

####

CREATE MATERIALIZED VIEW, DROP MATERIALIZED VIEW, REFRESH MATERIALIZED VIEW

ALTER OPERATOR

ALTER OPERATOR — ##########

##

```
ALTER OPERATOR name ( { left_type | NONE } , right_type )
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }

ALTER OPERATOR name ( { left_type | NONE } , right_type )
    SET SCHEMA new_schema

ALTER OPERATOR name ( { left_type | NONE } , right_type )
    SET ( {  RESTRICT = { res_proc | NONE }
           | JOIN = { join_proc | NONE }
           | COMMUTATOR = com_op
           | NEGATOR = neg_op
           | HASHES
           | MERGES
          } [, ... ] )
```

##

ALTER OPERATOR#############

ALTER OPERATOR######################## ########################SET ROLE##### ############################CREATE############## ############################ ##################### ####################################

#####

name

##########################

left_type

################## ################NONE#######

right_type

##################

new_owner

#############

new_schema

##############

res_proc

################# ##################NONE#######

join_proc

################# ##################NONE#######

com_op

################ ########################

neg_op

################ ########################

HASHES

################################## ####################

MERGES

################################# ####################

##

#########36.14#36.15##########

######################ALTER OPERATOR SET COMMUTATOR#*com_op*############## ### ####ALTER OPERATOR SET NEGATOR#*neg_op*################ ################ #######################

#

text########a @@ b##########

```
ALTER OPERATOR @@ (text, text) OWNER TO joe;
```

#####[97.278912]#int[]########a && b####################

```
ALTER OPERATOR && (int[], int[]) SET (RESTRICT = _int_contsel, JOIN =
 _int_contjoinsel);
```

######&&####################

```
ALTER OPERATOR && (int[], int[]) SET (COMMUTATOR = &&);
```

###

##SQL##ALTER OPERATOR#######

####

CREATE OPERATOR, DROP OPERATOR

ALTER OPERATOR CLASS

ALTER OPERATOR CLASS — ##############

##

```
ALTER OPERATOR CLASS name USING index_method
    RENAME TO new_name

ALTER OPERATOR CLASS name USING index_method
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }

ALTER OPERATOR CLASS name USING index_method
    SET SCHEMA new_schema
```

##

ALTER OPERATOR CLASS################

ALTER OPERATOR CLASS########################## ########################SET ROLE####################################CREATE############## ################ ##################################### ######################################

#####

name

##########################

index_method

#######################

new_name

############

new_owner

###############

new_schema

################

###

##SQL##ALTER OPERATOR CLASS########

####

CREATE OPERATOR CLASS, DROP OPERATOR CLASS, ALTER OPERATOR FAMILY

ALTER OPERATOR FAMILY

ALTER OPERATOR FAMILY — ###########

##

```
ALTER OPERATOR FAMILY name USING index_method ADD
  {  OPERATOR strategy_number operator_name ( op_type, op_type )
             [ FOR SEARCH | FOR ORDER BY sort_family_name ]
   | FUNCTION support_number [ ( op_type [ , op_type ] ) ]
             function_name [ ( argument_type [, ...] ) ]
  } [, ... ]

ALTER OPERATOR FAMILY name USING index_method DROP
  {  OPERATOR strategy_number ( op_type [ , op_type ] )
   | FUNCTION support_number ( op_type [ , op_type ] )
  } [, ... ]

ALTER OPERATOR FAMILY name USING index_method
    RENAME TO new_name

ALTER OPERATOR FAMILY name USING index_method
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }

ALTER OPERATOR FAMILY name USING index_method
    SET SCHEMA new_schema
```

##

ALTER OPERATOR FAMILY############## ######################################
#######################

ALTER OPERATOR FAMILY###
################### ##
CREATE OPERATOR CLASS###########
PostgreSQL################################## ################################
##################### ##
######### ##################################

ALTER OPERATOR FAMILY######################## ##############################
#####################

#####ALTER OPERATOR FAMILY##
########################

###36.16##########

#####

name

###################

index_method

#########################

strategy_number

##################################

operator_name

##########################

op_type

OPERATOR#######################NONE### CREATE OPERATOR CLASS######## #######################

ADD FUNCTION#######################(###############)# B-tree#######Hash##### ############################*op_type*############# B-tree#########B-tree##### ####GiST#SP-GiST#GIN###

DROP FUNCTION##################################

sort_family_name

#########################btree##################

FOR SEARCH#FOR ORDER BY#########FOR SEARCH########

support_number

##################################

function_name

######

argument_type

###############

new_name

############

new_owner

#############

new_schema

##############

OPERATOR#FUNCTION##############

##

DROP## ############### ################### ###DROP FUNCTION############################## GiST#SP-GiST###GIN##################################

################################ ## ######## ####################

####SQL############# SQL############################## ###################### ####################

PostgreSQL 8.4#######OPERATOR##RECHECK################# ##################### ## ## #################

# #

###########################int4#int2######B-Tree######################

```
ALTER OPERATOR FAMILY integer_ops USING btree ADD

  -- int4 vs int2
  OPERATOR 1 < (int4, int2) ,
  OPERATOR 2 <= (int4, int2) ,
  OPERATOR 3 = (int4, int2) ,
  OPERATOR 4 >= (int4, int2) ,
  OPERATOR 5 > (int4, int2) ,
  FUNCTION 1 btint42cmp(int4, int2) ,

  -- int2 vs int4
  OPERATOR 1 < (int2, int4) ,
  OPERATOR 2 <= (int2, int4) ,
  OPERATOR 3 = (int2, int4) ,
  OPERATOR 4 >= (int2, int4) ,
  OPERATOR 5 > (int2, int4) ,
  FUNCTION 1 btint24cmp(int2, int4) ;
```

###############

```
ALTER OPERATOR FAMILY integer_ops USING btree DROP

  -- int4 vs int2
  OPERATOR 1 (int4, int2) ,
  OPERATOR 2 (int4, int2) ,
  OPERATOR 3 (int4, int2) ,
  OPERATOR 4 (int4, int2) ,
  OPERATOR 5 (int4, int2) ,
  FUNCTION 1 (int4, int2) ,

  -- int2 vs int4
  OPERATOR 1 (int2, int4) ,
  OPERATOR 2 (int2, int4) ,
  OPERATOR 3 (int2, int4) ,
  OPERATOR 4 (int2, int4) ,
  OPERATOR 5 (int2, int4) ,
  FUNCTION 1 (int2, int4) ;
```

###

##SQL##ALTER OPERATOR FAMILY########

####

CREATE OPERATOR FAMILY, DROP OPERATOR FAMILY, CREATE OPERATOR CLASS, ALTER OPERATOR CLASS, DROP OPERATOR CLASS

ALTER POLICY

ALTER POLICY — ####################

##

```
ALTER POLICY name ON table_name RENAME TO new_name

ALTER POLICY name ON table_name
    [ TO { role_name | PUBLIC | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER } [, ...] ]
    [ USING ( using_expression ) ]
    [ WITH CHECK ( check_expression ) ]
```

##

ALTER POLICY######################### ALTER POLICY################## #USING#WITH CHECK####################### ################################ ############################

ALTER POLICY################################

ALTER POLICY#2##############*using_expression*#*check_expression*######### ############# ####1##############################

#####

name

#################

table_name

##############################

new_name

############

role_name

############### ################### ##################PUBLIC######

using_expression

#####USING#### ####CREATE POLICY########

check_expression

#####WITH CHECK#### ####CREATE POLICY########

###

ALTER POLICY#PostgreSQL######

####

CREATE POLICY, DROP POLICY

ALTER PROCEDURE

ALTER PROCEDURE — ##############

##

```
ALTER PROCEDURE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    action [ ... ] [ RESTRICT ]
ALTER PROCEDURE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    RENAME TO new_name
ALTER PROCEDURE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
ALTER PROCEDURE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    SET SCHEMA new_schema
ALTER PROCEDURE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    [ NO ] DEPENDS ON EXTENSION extension_name


###action###########

    [ EXTERNAL ] SECURITY INVOKER | [ EXTERNAL ] SECURITY DEFINER
    SET configuration_parameter { TO | = } { value | DEFAULT }
    SET configuration_parameter FROM CURRENT
    RESET configuration_parameter
    RESET ALL
```

##

ALTER PROCEDURE###############

ALTER PROCEDURE########################## ############################ #CREATE####### #######################SET ROLE############################ #######CREATE############## ## ### ##################################

#####

name

###################### ##################################

argmode

#######IN#OUT#INOUT#VARIADIC####### ############IN###

argname

######## ##################################ALTER PROCEDURE############## #########

argtype

################################## ##############################DROP PROCEDURE#########

new_name

###########

new_owner

###############SECURITY DEFINER#################################
##############

new_schema

##############

extension_name

#######################NO########################### ###################
##############################
###################

[EXTERNAL] SECURITY INVOKER
[EXTERNAL] SECURITY DEFINER

############################EXTERNAL#####SQL#################### ######
#####CREATE PROCEDURE#########

configuration_parameter
value

#####################################*value*#DEFAULT#########RESET####
######################### ############################## ##################
#####RESET ALL#########SET FROM CURRENT##ALTER PROCEDURE#########
##################################

#######################SET##19##########

RESTRICT

##SQL####################

# #

integer###############insert_data#insert_record########

```
ALTER PROCEDURE insert_data(integer, integer) RENAME TO insert_record;
```

integer###############insert_data#####joe######

```
ALTER PROCEDURE insert_data(integer, integer) OWNER TO joe;
```

integer###############insert_data#####accounting######

```
ALTER PROCEDURE insert_data(integer, integer) SET SCHEMA accounting;
```

#####insert_data(integer, integer)###myext##############

```
ALTER PROCEDURE insert_data(integer, integer) DEPENDS ON EXTENSION myext;
```

################################

```
ALTER PROCEDURE check_password(text) SET search_path = admin, pg_temp;
```

#########search_path##############

```
ALTER PROCEDURE check_password(text) RESET search_path;
```

##

###

####SQL###ALTER PROCEDURE############## ############################## ## ###### #RESTRICT###########PostgreSQL#########

####

CREATE PROCEDURE, DROP PROCEDURE, ALTER FUNCTION, ALTER ROUTINE

ALTER PUBLICATION

ALTER PUBLICATION — ###############

##

```
ALTER PUBLICATION name ADD publication_object [, ...]
ALTER PUBLICATION name SET publication_object [, ...]
ALTER PUBLICATION name DROP publication_object [, ...]
ALTER PUBLICATION name SET ( publication_parameter [= value] [, ... ] )
ALTER PUBLICATION name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER
 | SESSION_USER }
ALTER PUBLICATION name RENAME TO new_name
```

###publication_object###########

```
    TABLE [ ONLY ] table_name [ * ] [ ( column_name [, ... ] ) ] [ WHERE
 ( expression ) ] [, ... ]
    TABLES IN SCHEMA { schema_name | CURRENT_SCHEMA } [, ... ]
```

##

####ALTER PUBLICATION###################

#####[88.387716]####3#####################/################ SET###############/ ##/########### ADD##DROP###########1#### ####/############### ##########################/############################## ###ALTER SUBSCRIPTION ... REFRESH PUBLICATION################### ## DROP TABLES IN SCHEMA#FOR TABLE/ADD TABLE################### ##WHERE#####DROP########################### ########3#################### #/########### SET################/######################## ################### #/############ ADD#DROP###########1########/############### ################# #########/###########ALTER SUBSCRIPTION ... REFRESH PUBLICATION##### ## ############################## DROP TABLES IN SCHEMA#FOR TABLE/ADD TABLE## #################DROP#WHERE#########################

################4########CREATE PUBLICATION########################### ### ##################################

################################

#####[93.675418]#ALTER PUBLICATION################################ ######## ################################### ##########ADD TABLES IN SCHEMA#SET TABLES IN SCHEMA############################ ######################## #SET ROLE###########################CREATE################ ###FOR ALL TABLES#FOR TABLES IN SCHEMA################################## ############# ############################

##/###############

#####

name

    ###########################

table_name

############ ########ONLY########################## ########ONLY#########
################################## #############*#########################
#######

################## ###CREATE PUBLICATION######### ####################
##########################
###: ############################

######WHERE###########*expression*#####NULL################### #########
############# *expression*###########################

schema_name

###########

SET (*publication_parameter* [= *value*] [, ...])

########CREATE PUBLICATION########################### ################
##

new_owner

######################

new_name

################

# #

delete#update############################

```
ALTER PUBLICATION noinsert SET (publish = 'update, delete');
```

#########################

```
ALTER PUBLICATION mypublication ADD TABLE users (user_id, firstname),
 departments;
```

#########################

```
ALTER PUBLICATION mypublication SET TABLE users (user_id, firstname,
 lastname), TABLE departments;
```

########sales_publication#####marketing#sales#######

```
ALTER PUBLICATION sales_publication ADD TABLES IN SCHEMA marketing, sales;
```

########production_publication#####users#departments#####production#####
###

```
ALTER PUBLICATION production_publication ADD TABLE users, departments,
 TABLES IN SCHEMA production;
```

###

ALTER PUBLICATION#PostgreSQL######

####

CREATE PUBLICATION, DROP PUBLICATION, CREATE SUBSCRIPTION, ALTER SUBSCRIPTION

ALTER ROLE

ALTER ROLE — ##############

##

```
ALTER ROLE role_specification [ WITH ] option [ ... ]


###option#########

      SUPERUSER | NOSUPERUSER
    | CREATEDB | NOCREATEDB
    | CREATEROLE | NOCREATEROLE
    | INHERIT | NOINHERIT
    | LOGIN | NOLOGIN
    | REPLICATION | NOREPLICATION
    | BYPASSRLS | NOBYPASSRLS
    | CONNECTION LIMIT connlimit
    | [ ENCRYPTED ] PASSWORD 'password' | PASSWORD NULL
    | VALID UNTIL 'timestamp'

ALTER ROLE name RENAME TO new_name

ALTER ROLE { role_specification | ALL } [ IN DATABASE database_name ]
 SET configuration_parameter { TO | = } { value | DEFAULT }
ALTER ROLE { role_specification | ALL } [ IN DATABASE database_name ]
 SET configuration_parameter FROM CURRENT
ALTER ROLE { role_specification | ALL } [ IN DATABASE database_name ]
 RESET configuration_parameter
ALTER ROLE { role_specification | ALL } [ IN DATABASE database_name ]
 RESET ALL


###role_specification#########

    role_name
  | CURRENT_ROLE
  | CURRENT_USER
  | SESSION_USER
```

##

ALTER ROLE#PostgreSQL##############

#####[83.876092]###############CREATE ROLE#################### ## ## #########GRANT#REVOKE####### #### ############################## ##################################### CREATEROLE##################################ADMIN OPTION########### ########################### ########SUPERUSER############################ #CREATEDB#REPLICATION#BYPASSRLS######### ######################## ###### ##################CREATE ROLE#################### ################################## ########################### ######GRANT#REVOKE########### ################## ########### ##############################SUPERUSER########################### #### CREATEROLE###ADMIN OPTION#### ##################################### ##############SUPERUSER############## # ##CREATEDB#REPLICATION#BYPASSRLS################################### ##### ####################

2#################### ############################## CREATEROLE#######
##ADMIN OPTION############################## ########################## ##
################## MD5#####################################MD5#############
####################

##################IN DATABASE##
########### ########ALL######################## ALL#IN DATABASE###########
#ALTER DATABASE ... SET ...############

#######################postgresql.conf#####postgres#####################
######################### ################ SET ROLE###SET SESSION
AUTHORIZATION################### ##
##

################################## CREATEROLE########ADMIN OPTION########
#################### ####################### ########################## #####
################################## ##

#####

name

###############

CURRENT_ROLE
CURRENT_USER

############################

SESSION_USER

################################

SUPERUSER
NOSUPERUSER
CREATEDB
NOCREATEDB
CREATEROLE
NOCREATEROLE
INHERIT
NOINHERIT
LOGIN
NOLOGIN
REPLICATION
NOREPLICATION
BYPASSRLS
NOBYPASSRLS
CONNECTION LIMIT *connlimit*
[ENCRYPTED] PASSWORD '*password*'
PASSWORD NULL
VALID UNTIL '*timestamp*'

########CREATE ROLE############## ###CREATE ROLE##################

new_name

###########

database_name

######################

configuration_parameter
value

value#DEFAULT####RESET#####
#################
#RESET ALL######### SET FROM CURRENT########################### IN
DATABASE##

########################## SET ROLE###SET SESSION AUTHORIZATION#####
############

####################SET####19##########

##

############CREATE ROLE######### ############DROP ROLE#########

ALTER ROLE################### #########GRANT###REVOKE#########

###################################### ################### ###################
################### psql##\password######### ################################
###########

###################################### ALTER DATABASE######### ############
###

# #

###############

ALTER ROLE davide WITH PASSWORD 'hu8jmn3';

###############

ALTER ROLE davide WITH PASSWORD NULL;

###############UTC#1################2015#5#4#######################

ALTER ROLE chris VALID UNTIL 'May 4 12:00:00 2015 +1';

#################

ALTER ROLE fred VALID UNTIL 'infinity';

##################################

ALTER ROLE miriam CREATEROLE CREATEDB;

####maintenance_work_mem#######################

ALTER ROLE worker_bee SET maintenance_work_mem = 100000;

#############client_min_messages#######################

ALTER ROLE fred IN DATABASE devel SET client_min_messages = DEBUG;

###

`ALTER ROLE`##PostgreSQL######

####

CREATE ROLE, DROP ROLE, ALTER DATABASE, SET

ALTER ROUTINE

ALTER ROUTINE — ###########

##

```
ALTER ROUTINE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    action [ ... ] [ RESTRICT ]
ALTER ROUTINE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    RENAME TO new_name
ALTER ROUTINE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
ALTER ROUTINE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    SET SCHEMA new_schema
ALTER ROUTINE name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
    [ NO ] DEPENDS ON EXTENSION extension_name
```

```
###action###########

    IMMUTABLE | STABLE | VOLATILE
    [ NOT ] LEAKPROOF
    [ EXTERNAL ] SECURITY INVOKER | [ EXTERNAL ] SECURITY DEFINER
    PARALLEL { UNSAFE | RESTRICTED | SAFE }
    COST execution_cost
    ROWS result_rows
    SET configuration_parameter { TO | = } { value | DEFAULT }
    SET configuration_parameter FROM CURRENT
    RESET configuration_parameter
    RESET ALL
```

##

ALTER ROUTINE#################################### ####################
##ALTER AGGREGATE#ALTER FUNCTION#ALTER PROCEDURE#########

#

integer########foo#foobar########

```
ALTER ROUTINE foo(integer) RENAME TO foobar;
```

#######foo#############################

###

####SQL###ALTER ROUTINE############## ######ALTER FUNCTION#ALTER PROCEDURE######### ##################PostgreSQL######

####

ALTER AGGREGATE, ALTER FUNCTION, ALTER PROCEDURE, DROP ROUTINE

CREATE ROUTINE##################

ALTER RULE

ALTER RULE — ##########

##

```
ALTER RULE name ON table_name RENAME TO new_name
```

##

ALTER RULE################ ########################

ALTER RULE##

#####

name

################

table_name

################################

new_name

###########

#

###############

```
ALTER RULE notify_all ON emp RENAME TO notify_me;
```

###

ALTER RULE#PostgreSQL############################

####

CREATE RULE, DROP RULE

ALTER SCHEMA

ALTER SCHEMA — ##########

##

```
ALTER SCHEMA name RENAME TO new_name
ALTER SCHEMA name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
```

##

ALTER SCHEMA#############

ALTER SCHEMA######################### ######################CREATE##########
########################SET ROLE################################CREATE##
############ ######################################

#####

name

#############

new_name

############ pg_##

new_owner

##############

###

##SQL##ALTER SCHEMA########

####

CREATE SCHEMA, DROP SCHEMA

ALTER SEQUENCE

ALTER SEQUENCE — ##################

##

```
ALTER SEQUENCE [ IF EXISTS ] name
    [ AS data_type ]
    [ INCREMENT [ BY ] increment ]
    [ MINVALUE minvalue | NO MINVALUE ] [ MAXVALUE maxvalue | NO
 MAXVALUE ]
    [ START [ WITH ] start ]
    [ RESTART [ [ WITH ] restart ] ]
    [ CACHE cache ] [ [ NO ] CYCLE ]
    [ OWNED BY { table_name.column_name | NONE } ]
ALTER SEQUENCE [ IF EXISTS ] name SET { LOGGED | UNLOGGED }
ALTER SEQUENCE [ IF EXISTS ] name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
ALTER SEQUENCE [ IF EXISTS ] name RENAME TO new_name
ALTER SEQUENCE [ IF EXISTS ] name SET SCHEMA new_schema
```

##

ALTER SEQUENCE############################ ALTER SEQUENCE################
###############

ALTER SEQUENCE########################## ##########################
#CREATE############## #######################SET ROLE#####################
#########CREATE############## ##
################### ################################

#####

name

##########################

IF EXISTS

###################### ##################

data_type

######AS *data_type*################ #####smallint#integer#bigint###

###
###NO MINVALUE###NO MAXVALUE################################### ####
#########################
##########

increment

INCREMENT BY *increment*######## ######################################
####################

minvalue
NO MINVALUE

MINVALUE *minvalue*########################## NO MINVALUE##############
1######################### ##################################

maxvalue
NO MAXVALUE

MAXVALUE *maxvalue*#####################NO MAXVALUE#####################
#######-1########### ##################################

start

START WITH *start*#(####)######################### ################### #####
####ALTER SEQUENCE RESTART###################

restart

RESTART WITH *restart*#(####)################## ###is_called = false##
#setval############### ########nextval######### *restart*#####RESTART###
######CREATE SEQUENCE#########ALTER SEQUENCE START WITH#############
#########

setval###################RESTART#######################################
############# ###############setval#########

cache

CACHE *cache*### ####1##
##########1######################### ########################

CYCLE

CYCLE###########################*maxvalue*#######*minvalue*###############
############ ##########################*minvalue*#######*maxvalue*######

NO CYCLE

NO CYCLE##########################nextval############### CYCLE#NO
CYCLE###########################

SET { LOGGED | UNLOGGED }

##(CREATE SEQUENCE#########)# ##
###############

OWNED BY *table_name.column_name*
OWNED BY NONE

OWNED BY###
############ ### #####################
################################ OWNED BY NONE##########################
############

new_owner

###################

new_name

##############

new_schema

################

##

ALTER SEQUENCE##########################nextval################## ##

ALTER SEQUENCE######currval########## #8.3####PostgreSQL################

ALTER SEQUENCE#########nextval#currval#lastval#setval############

#########ALTER TABLE################## ################ALTER TABLE##############

# #

serial########105#######

```
ALTER SEQUENCE serial RESTART WITH 105;
```

###

ALTER SEQUENCE##PostgreSQL######AS#START WITH#OWNED BY#OWNER TO#RENAME TO#SET SCHEMA########SQL#######

####

CREATE SEQUENCE, DROP SEQUENCE

ALTER SERVER

ALTER SERVER — ############

##

```
ALTER SERVER name [ VERSION 'new_version' ]
    [ OPTIONS ( [ ADD | SET | DROP ] option ['value'] [, ... ] ) ]
ALTER SERVER name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER SERVER name RENAME TO new_name
```

##

ALTER SERVER############### #1#################################### ######1 ######### #2################

############################## ############################SET ROLE########### ####################USAGE######## ##

#####

name

############

new_version

##############

OPTIONS ([ADD | SET | DROP] *option* ['*value*'] [, ...])

############### ADD#SET#DROP########### ##############ADD######## ######## ########### ####################################

new_owner

###################

new_name

#############

#

```
###foo##################

ALTER SERVER foo OPTIONS (host 'foo', dbname 'foodb');
```

```
###foo##########host###########

ALTER SERVER foo VERSION '8.4' OPTIONS (SET host 'baz');
```

###

ALTER SERVER#ISO/IEC 9075-9 (SQL/MED)###### OWNER TO#RENAME###PostgreSQL#### ##

####

CREATE SERVER, DROP SERVER

ALTER STATISTICS

ALTER STATISTICS — ################

##

```
ALTER STATISTICS name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER STATISTICS name RENAME TO new_name
ALTER STATISTICS name SET SCHEMA new_schema
ALTER STATISTICS name SET STATISTICS { new_target | DEFAULT }
```

##

ALTER STATISTICS######################### ALTER STATISTICS############## ##################

ALTER STATISTICS################################ ######################### #######CREATE################ ######################SET ROLE############## ########################CREATE################ ############################## ## ################################ ###

#####

name

##################################

new_owner

######################

new_name

################

new_schema

##################

new_target

#####[63.008850]####ANALYZE############################## ###0##10000######## ##(default_statistics_target)#### #######-1####### PostgreSQL############################14.2########## ####### #############ANALYZE######### #####0##10000########## ################## (default_statistics_target)###########DEFAULT####### ### -1 ###################### #### PostgreSQL########################14.2##########

###

##SQL##ALTER STATISTICS##########

####

CREATE STATISTICS, DROP STATISTICS

ALTER SUBSCRIPTION

ALTER SUBSCRIPTION — ################

##

```
ALTER SUBSCRIPTION name CONNECTION 'conninfo'
ALTER SUBSCRIPTION name SET PUBLICATION publication_name [, ...] [ WITH
 ( publication_option [= value] [, ... ] ) ]
ALTER SUBSCRIPTION name ADD PUBLICATION publication_name [, ...] [ WITH
 ( publication_option [= value] [, ... ] ) ]
ALTER SUBSCRIPTION name DROP PUBLICATION publication_name [, ...] [ WITH
 ( publication_option [= value] [, ... ] ) ]
ALTER SUBSCRIPTION name REFRESH PUBLICATION [ WITH ( refresh_option
 [= value] [, ... ] ) ]
ALTER SUBSCRIPTION name ENABLE
ALTER SUBSCRIPTION name DISABLE
ALTER SUBSCRIPTION name SET ( subscription_parameter [= value] [, ... ] )
ALTER SUBSCRIPTION name SKIP ( skip_option = value )
ALTER SUBSCRIPTION name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER
 | SESSION_USER }
ALTER SUBSCRIPTION name RENAME TO new_name
```

##

ALTER SUBSCRIPTION#CREATE SUBSCRIPTION##############################

ALTER SUBSCRIPTION################################## ################### ######################CREATE####### ###########################SET ROLE# ########### ############################CREATE############## ######## #password_required=false######################

##################PostgreSQL# ########################### ### ##DROP SUBSCRIPTION###########################

##########ALTER#########.################ALTER#########.{SET ADD DROP} ########.with######option as####ALTER#########.SET#########=############# #############

#####[68.451520]##########two_phase################copy_data#false######## ############# ###2########pg_subscription#subtwophasestate########## ### ###ALTER SUBSCRIPTION ... REFRESH PUBLICATION#ALTER SUBSCRIPTION ... {SET|ADD|DROP} PUBLICATION ...################################ #two_phase##################_################## ############pg_######## ####subtwophasestate#########

#####

name

#########################

CONNECTION '*conninfo*'

########CREATE SUBSCRIPTION################### ##################

SET PUBLICATION *publication_name*
ADD PUBLICATION *publication_name*
DROP PUBLICATION *publication_name*

################################ SET##########################ADD######
######################DROP############################ #################
###########ADD#SET############# ###CREATE SUBSCRIPTION######### ######
########REFRESH PUBLICATION##########

publication_option######################### ###################

refresh (boolean)

false########################## ####REFRESH PUBLICATION############ #
#####true###

##################REFRESH PUBLICATION#####################

REFRESH PUBLICATION

#####[78.839590]########################### CREATE SUBSCRIPTION######
##REFRESH PUBLICATION##
###CREATE SUBSCRIPTION#REFRESH
PUBLICATION###

refresh_option###(refresh)#################### ###################

copy_data (boolean)

######true###

###########WHERE#####################################

#####[79.475983]#copy_data = true#origin############################
######## ######copy_data = true#origin############################
#####

#####[77.830189]##########################CREATE
SUBSCRIPTION#binary############## ############################
##CREATE SUBSCRIPTION#binary##############

ENABLE

###

DISABLE

##

SET (*subscription_parameter* [= *value*] [, ...])

#####[80.380294]########CREATE SUBSCRIPTION################### #######
########### ########### slot_name# synchronous_commit# binary#
streaming# disable_on_error# password_required# run_as_owner#
origin ### #########password_required = false####### #######
####CREATE SUBSCRIPTION#################### #### ######### #########
##slot_name#synchronous_commit#binary#streaming#disable_on_error#password_requ
#########password_required = false#######

######slot_name################failover###two_phase##################
####failover###two_phase######################## ####################

##failover#two_phase################################## ###############
##############failover####
############failover######
################################

SKIP (*skip_option* = *value*)

#####[89.146341]############################## ############################
################ ALTER SUBSCRIPTION ... SKIP#############
################two_phase#####
########################## ##
##LSN(pg_subscription.subskiplsn########)######## ##################
#29.6######### ################################### ####################
################### ALTER SUBSCRIPTION ... SKIP####################
##################two_phase###########################
################################LSN#pg_subscription.subskiplsn#####
####### ######################29.6#########

skip_option################## #######################

lsn (pg_lsn)

##LSN###### ##LSN###########
##########LSN### ########################### NONE######LSN##########

new_owner

#######################

new_name

#################

boolean##############= *value*#############TRUE###########

# #

###########################insert_only######

ALTER SUBSCRIPTION mysub SET PUBLICATION insert_only;

####################

ALTER SUBSCRIPTION mysub DISABLE;

###

ALTER SUBSCRIPTION#PostgreSQL#####

####

CREATE SUBSCRIPTION, DROP SUBSCRIPTION, CREATE PUBLICATION, ALTER PUBLICATION

ALTER SYSTEM

ALTER SYSTEM — ###############

##

```
ALTER SYSTEM SET configuration_parameter { TO | = } { value [, ...] |
 DEFAULT }

ALTER SYSTEM RESET configuration_parameter
ALTER SYSTEM RESET ALL
```

##

ALTER SYSTEM###################################### ####postgresql.conf## ################################ ALTER SYSTEM############ #postgresql.auto.conf############postgresql.conf########### ##### #DEFAULT##################RESET#####postgresql.auto.conf################ ###### ####################RESET ALL#########

ALTER SYSTEM## ## ###########SQL##pg_reload_conf()######pg_ctl reload################## #SIGHUP##############

################ALTER SYSTEM###############ALTER SYSTEM################ # ##

#####

configuration_parameter

################ ###############19###########

value

############ ### ####### ######################## DEFAULT######################## #postgresql.auto.conf#######

##

##

#####[81.782946]#########data_directory###postgresql.conf###########(###preset options)############## ############data_directory#allow_alter_system################ ################################

####################19.1#########

######ALTER SYSTEM#allow_alter_system#off################################### ###########################

#

wal_level######

```
ALTER SYSTEM SET wal_level = replica;
```

```
#########postgresql.conf##############
```

```
ALTER SYSTEM RESET wal_level;
```

###

ALTER SYSTEM##PostgreSQL######

####

SET, SHOW

ALTER TABLE

ALTER TABLE — ###########

##

```
ALTER TABLE [ IF EXISTS ] [ ONLY ] name [ * ]
    action [, ... ]
ALTER TABLE [ IF EXISTS ] [ ONLY ] name [ * ]
    RENAME [ COLUMN ] column_name TO new_column_name
ALTER TABLE [ IF EXISTS ] [ ONLY ] name [ * ]
    RENAME CONSTRAINT constraint_name TO new_constraint_name
ALTER TABLE [ IF EXISTS ] name
    RENAME TO new_name
ALTER TABLE [ IF EXISTS ] name
    SET SCHEMA new_schema
ALTER TABLE ALL IN TABLESPACE name [ OWNED BY role_name [, ... ] ]
    SET TABLESPACE new_tablespace [ NOWAIT ]
ALTER TABLE [ IF EXISTS ] name
    ATTACH PARTITION partition_name { FOR VALUES partition_bound_spec |
 DEFAULT }
ALTER TABLE [ IF EXISTS ] name
    DETACH PARTITION partition_name [ CONCURRENTLY | FINALIZE ]
ALTER TABLE [ IF EXISTS ] name
    SPLIT PARTITION partition_name INTO
        (PARTITION partition_name1 { FOR VALUES partition_bound_spec |
 DEFAULT },
         PARTITION partition_name2 { FOR VALUES partition_bound_spec |
 DEFAULT } [, ...])
ALTER TABLE [ IF EXISTS ] name
    MERGE PARTITIONS (partition_name1, partition_name2 [, ...])
        INTO partition_name


####action###########

    ADD [ COLUMN ] [ IF NOT EXISTS ] column_name data_type
 [ COLLATE collation ] [ column_constraint [ ... ] ]
    DROP [ COLUMN ] [ IF EXISTS ] column_name [ RESTRICT | CASCADE ]
    ALTER [ COLUMN ] column_name [ SET DATA ] TYPE data_type
 [ COLLATE collation ] [ USING expression ]
    ALTER [ COLUMN ] column_name SET DEFAULT expression
    ALTER [ COLUMN ] column_name DROP DEFAULT
    ALTER [ COLUMN ] column_name { SET | DROP } NOT NULL
    ALTER [ COLUMN ] column_name SET EXPRESSION AS ( expression )
    ALTER [ COLUMN ] column_name DROP EXPRESSION [ IF EXISTS ]
    ALTER [ COLUMN ] column_name ADD GENERATED { ALWAYS | BY DEFAULT } AS
 IDENTITY [ ( sequence_options ) ]
    ALTER [ COLUMN ] column_name { SET GENERATED { ALWAYS | BY DEFAULT } |
 SET sequence_option | RESTART [ [ WITH ] restart ] } [...]
    ALTER [ COLUMN ] column_name DROP IDENTITY [ IF EXISTS ]
    ALTER [ COLUMN ] column_name SET STATISTICS { integer | DEFAULT }
    ALTER [ COLUMN ] column_name SET ( attribute_option = value [, ... ] )
    ALTER [ COLUMN ] column_name RESET ( attribute_option [, ... ] )
    ALTER [ COLUMN ] column_name SET STORAGE { PLAIN | EXTERNAL | EXTENDED
 | MAIN | DEFAULT }
    ALTER [ COLUMN ] column_name SET COMPRESSION compression_method
    ADD table_constraint [ NOT VALID ]
    ADD table_constraint_using_index
```

```
    ALTER CONSTRAINT constraint_name [ DEFERRABLE | NOT DEFERRABLE ]
 [ INITIALLY DEFERRED | INITIALLY IMMEDIATE ]
    VALIDATE CONSTRAINT constraint_name
    DROP CONSTRAINT [ IF EXISTS ]  constraint_name [ RESTRICT | CASCADE ]
    DISABLE TRIGGER [ trigger_name | ALL | USER ]
    ENABLE TRIGGER [ trigger_name | ALL | USER ]
    ENABLE REPLICA TRIGGER trigger_name
    ENABLE ALWAYS TRIGGER trigger_name
    DISABLE RULE rewrite_rule_name
    ENABLE RULE rewrite_rule_name
    ENABLE REPLICA RULE rewrite_rule_name
    ENABLE ALWAYS RULE rewrite_rule_name
    DISABLE ROW LEVEL SECURITY
    ENABLE ROW LEVEL SECURITY
    FORCE ROW LEVEL SECURITY
    NO FORCE ROW LEVEL SECURITY
    CLUSTER ON index_name
    SET WITHOUT CLUSTER
    SET WITHOUT OIDS
    SET ACCESS METHOD { new_access_method | DEFAULT }
    SET TABLESPACE new_tablespace
    SET { LOGGED | UNLOGGED }
    SET ( storage_parameter [= value] [, ... ] )
    RESET ( storage_parameter [, ... ] )
    INHERIT parent_table
    NO INHERIT parent_table
    OF type_name
    NOT OF
    OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
    REPLICA IDENTITY { DEFAULT | USING INDEX index_name | FULL | NOTHING }


###partition_bound_spec###########

IN ( partition_bound_expr [, ...] ) |
FROM ( { partition_bound_expr | MINVALUE | MAXVALUE } [, ...] )
  TO ( { partition_bound_expr | MINVALUE | MAXVALUE } [, ...] ) |
WITH ( MODULUS numeric_literal, REMAINDER numeric_literal )


###column_constraint#########

[ CONSTRAINT constraint_name ]
{ NOT NULL |
  NULL |
  CHECK ( expression ) [ NO INHERIT ] |
  DEFAULT default_expr |
  GENERATED ALWAYS AS ( generation_expr ) STORED |
  GENERATED { ALWAYS | BY DEFAULT } AS IDENTITY [ ( sequence_options ) ] |
  UNIQUE [ NULLS [ NOT ] DISTINCT ] index_parameters |
  PRIMARY KEY index_parameters |
  REFERENCES reftable [ ( refcolumn ) ] [ MATCH FULL | MATCH PARTIAL |
 MATCH SIMPLE ]
    [ ON DELETE referential_action ] [ ON UPDATE referential_action ] }
[ DEFERRABLE | NOT DEFERRABLE ] [ INITIALLY DEFERRED | INITIALLY
 IMMEDIATE ]


###table_constraint#########

[ CONSTRAINT constraint_name ]
{ CHECK ( expression ) [ NO INHERIT ] |
  UNIQUE [ NULLS [ NOT ] DISTINCT ] ( column_name
 [, ... ] ) index_parameters |
```

```
  PRIMARY KEY ( column_name [, ... ] ) index_parameters |
  EXCLUDE [ USING index_method ] ( exclude_element WITH operator
 [, ... ] ) index_parameters [ WHERE ( predicate ) ] |
  FOREIGN KEY ( column_name [, ... ] ) REFERENCES reftable [ ( refcolumn
 [, ... ] ) ]
     [ MATCH FULL | MATCH PARTIAL | MATCH SIMPLE ] [ ON
 DELETE referential_action ] [ ON UPDATE referential_action ] }
[ DEFERRABLE | NOT DEFERRABLE ] [ INITIALLY DEFERRED | INITIALLY
 IMMEDIATE ]


##table_constraint_using_index#########

    [ CONSTRAINT constraint_name ]
    { UNIQUE | PRIMARY KEY } USING INDEX index_name
    [ DEFERRABLE | NOT DEFERRABLE ] [ INITIALLY DEFERRED | INITIALLY
 IMMEDIATE ]


UNIQUE#PRIMARY KEY#####EXCLUDE####index_parameters#########

[ INCLUDE ( column_name [, ... ] ) ]
[ WITH ( storage_parameter [= value] [, ... ] ) ]
[ USING INDEX TABLESPACE tablespace_name ]


EXCLUDE####exclude_element#########

{ column_name | ( expression ) } [ COLLATE collation ] [ opclass
 [ ( opclass_parameter = value [, ... ] ) ] ] [ ASC | DESC ] [ NULLS
 { FIRST | LAST } ]

referential_action in a FOREIGN KEY/REFERENCES constraint is:

{ NO ACTION | RESTRICT | CASCADE | SET NULL [ ( column_name [, ... ] ) ] |
 SET DEFAULT [ ( column_name [, ... ] ) ] }
```

##

```
ALTER TABLE################## ################### ############################
########### #########ACCESS EXCLUSIVE########## ############################
##########################

ADD COLUMN [ IF NOT EXISTS ]

   ###########CREATE TABLE######################## IF NOT EXISTS#########
   ###########################

DROP COLUMN [ IF EXISTS ]

   ########################## ##################################### ###########
   #######################1############################## ####################
   ######################CASCADE############ IF EXISTS######################
   ############# ##################

SET DATA TYPE

   ########################### ##############################################
   ############ COLLATE##################### ######################### USING#
   ################################ ########################################
   ####### ######################################USING##############

   ####################################ANALYZE##############
```

SET/DROP DEFAULT

##################################(##############NULL###########)# #########
##########INSERT###UPDATE############# ##################

SET/DROP NOT NULL

##############NULL#################

SET NOT NULL#############NULL#######1#################### ####
#ALTER TABLE#############################NULL##############CHECK#######
#################

#######################NOT NULL##############DROP NOT NULL#########
############NOT NULL###############DROP NOT NULL#########
#NOT NULL### ##########NULL########
#####NULL#######################

SET EXPRESSION AS

############################## ##

DROP EXPRESSION [IF EXISTS]

########################### ####################################

DROP EXPRESSION IF EXISTS################################### #########
###########

ADD GENERATED { ALWAYS | BY DEFAULT } AS IDENTITY
SET GENERATED { ALWAYS | BY DEFAULT }
DROP IDENTITY [IF EXISTS]

#########IDENTITY##############IDENTITY################## ###CREATE
TABLE######### SET DEFAULT################INSERT###UPDATE###########
##################

DROP IDENTITY IF EXISTS#########IDENTITY################### #########
###########

SET *sequence_option*
RESTART

##########IDENTITY#################### *sequence_option*#INCREMENT BY#
#ALTER SEQUENCE##############

SET STATISTICS

#####[73.451327]##########ANALYZE########################## ######0##10000
############## ######-1#######################default_statistics_target########
PostgreSQL##########################14.2######### ##############ANALYZE#
######################### ######0##10000########## ################
(default_statistics_target)###########DEFAULT####### ### -1 ####################
PostgreSQL#######################14.2#########

SET STATISTICS#SHARE UPDATE EXCLUSIVE#########

SET (*attribute_option* = *value* [, ...])
RESET (*attribute_option* [, ...])

############################## #######################n_distinct##
#n_distinct_inherited##### ##############ANALYZE####################
n_distinct##################n_distinct_inherited################

################## #######ANALYZE##################NULL#############
##########-1#############ANALYZE#######NULL######################
################################### #####-1#########################-0.5##
###########2############ ##
################# 0########################## PostgreSQL################
#######14.2#########

#############SHARE UPDATE EXCLUSIVE#########

SET STORAGE { PLAIN | EXTERNAL | EXTENDED | MAIN | DEFAULT }

################## #############TOAST###############################
PLAIN##integer################################## MAIN##############
############## EXTERNAL################## EXTENDED################
DEFAULT################################## EXTENDED##PLAIN############
################## EXTERNAL###########text###bytea##################
############################ ALTER TABLE ... SET STORAGE##############
############# ########################## ###65.2#########

SET COMPRESSION *compression_method*

##(#################)# ######
#####pg_restore###########
#################### ######################(###INSERT ... SELECT####)#
################TOAST###
################ #############pglz#lz4##(lz4##PostgreSQL######--with-
lz4################)# ####*compression_method*#default################
###default_toast_compression################################

ADD *table_constraint* [NOT VALID]

######CREATE TABLE################################NOT VALID###########
#############

####NOT VALID#
##################################### ##################################
########
##VALIDATE CONSTRAINT## NOT
VALID################################

ADD *table_constraint*#########ACCESS EXCLUSIVE#########ADD FOREIGN
KEY##SHARE ROW EXCLUSIVE########## ADD FOREIGN KEY################
###########SHARE ROW EXCLUSIVE##################

CREATE TABLE#########
#########################NOT VALID########

ADD *table_constraint_using_index*

###########################PRIMARY KEY###UNIQUE########### #############
###########

##################B-Tree#########
####################ADD PRIMARY KEY###ADD UNIQUE#############
##################

PRIMARY KEY##############NOT NULL################################ALTER
COLUMN SET NOT NULL######## #####NULL################################
######################

#################################### ################################

##ADD PRIMARY KEY###ADD UNIQUE# ############## ##########################

################################

##

#####[80.733945]### ################### #######CREATE INDEX CONCURRENTLY####### ##################################### ############# ################ ### #######CREATE UNIQUE INDEX CONCURRENTLY################################ ###### #######

ALTER CONSTRAINT

######################## ##################

VALIDATE CONSTRAINT

#########NOT VALID## ############ ############################## (##########################)

#######SHARE UPDATE EXCLUSIVE#########

DROP CONSTRAINT [IF EXISTS]

IF EXISTS######################### ###

DISABLE/ENABLE [REPLICA | ALWAYS] TRIGGER

################################### ##################################### ######## ## ###### #1### ################## ########### ## ### ### ############################## ################################

############session_replication_role######### ############################## ##origin###########local########### ENABLE REPLICA################ ##replica################# ####ENABLE ALWAYS############################## #######

#################################### ################################### ## ########### ##ENABLE ALWAYS###############

############################ONLY#################################### ##

#######SHARE ROW EXCLUSIVE#########

DISABLE/ENABLE [REPLICA | ALWAYS] RULE

#####ON SELECT########### ################

######################session_replication_role#########

DISABLE/ENABLE ROW LEVEL SECURITY

################################ ######################################
################
CREATE POLICY#########

NO FORCE/FORCE ROW LEVEL SECURITY

################
####################### ##
CREATE POLICY#########

CLUSTER ON

#########CLUSTER#################### ####################

cluster########SHARE UPDATE EXCLUSIVE#########

SET WITHOUT CLUSTER

######################CLUSTER############## ############################
##

cluster########SHARE UPDATE EXCLUSIVE#########

SET WITHOUT OIDS

#####oid################## #####oid#########################

SET ACCESS METHOD

###
default_table_access_method######################### ####61##########

###USING###############
################## DEFAULT##############################
#default_table_access_method#####

SET TABLESPACE

###
######################## ###############SET TABLESPACE########## ######
###########################CREATE TABLE PARTITION OF###########
##TABLESPACE###############################

ALL IN TABLESPACE## #####
################################### #####OWNED BY######################
################## NOWAIT## ###
##################################ALTER DATABASE########ALTER TABLE#
############ information_schema################################ CREATE
TABLESPACE#########

SET { LOGGED | UNLOGGED }

###UNLOGGED#### #############
###########

####################(ID#########)########### ################################

SET (*storage_parameter* [= *value*] [, ...])

######1###################### #################CREATE TABLE##############
##############################

##############################VACUUM FULL#CLUSTER################ALTER TABLE#
##
#######

fillfactor#TOAST####autovacuum##########################parallel_workers#
####SHARE UPDATE EXCLUSIVE##########

RESET (*storage_parameter* [, ...])

######1#########################SET#####################################

INHERIT *parent_table*

###
###
###
#####################NOT NULL#################NOT NULL##############

#########CHECK##
######ALTER TABLE ... ADD CONSTRAINT ... NO INHERIT###############(##
########)###UNIQUE#PRIMARY KEY#FOREIGN
KEY#########################

NO INHERIT *parent_table*

###
###

OF *type_name*

######CREATE TABLE OF###
##
##################

NOT OF

###########################

OWNER TO

###

REPLICA IDENTITY

###
###
###########################

DEFAULT

###

USING INDEX *index_name*

###
################################NOT NULL#####
################################NOTHING######

FULL

#################

NOTHING

###

RENAME

RENAME## ################# ## ################ #####

SET SCHEMA

################################ ## ##

ATTACH PARTITION *partition_name* { FOR VALUES *partition_bound_spec* | DEFAULT }

#####FOR VALUES#######################DEFAULT######################## ######## ################################### ################################ALTER INDEX ATTACH PARTITION############################## ## ############################UNIQUE################################## ##################CREATE FOREIGN TABLE########## ###################

FOR VALUES###########*partition_bound_spec*#CREATE TABLE########## ### ################### ################################ ##############NOT NULL###CHECK######### ##### #####FOREIGN KEY########## #####UNIQUE###PRIMARY KEY########## ################# ##################CHECK###NO INHERIT################ ###### ########NO INHERIT################

##CHECK#################### ######## CHECK## ######### ####################NULL######################## NULL################### ###########################NOT NULL###########

###CREATE FOREIGN TABLE############

##################CHECK############ ##################################

##ACCESS EXCLUSIVE############ #SHARE UPDATE EXCLUSIVE#########

5.12.2.2#########################CHECK############

DETACH PARTITION *partition_name* [CONCURRENTLY | FINALIZE]

################################ ## ################ ################################### ################### ######### SHARE##

CONCURRENTLY## ######### #######2################### ############SHARE UPDATE EXCLUSIVE## ############## ## ####################2########

###############SHARE UPDATE EXCLUSIVE############ACCESS EXCLUSIVE## ################# #############CHECK################# CONCURRENTLY##### ##

FINALIZE#########################DETACH CONCURRENTLY########### #### ############1#######################

SPLIT PARTITION *partition_name* INTO (PARTITION *partition_name1* { FOR VALUES *partition_bound_spec* | DEFAULT }, PARTITION *partition_name2* { FOR VALUES *partition_bound_spec* | DEFAULT } [, ...])

################################## ################## ################# ##################*partition_name*############## #########DEFAULT#### ################1##DEFAULT########## ################################ DEFAULT#############*partition_name1*#*partition_name2*#...############# ############# ############DEFAULT############DEFAULT################# #1##########

################DEFAULT##########################DEFAULT########### ######################### ##############*partition_name1*, *partition_name2*, ...##########*partition_name*############### ######## #*partition_name1*#*partition_name2*#...######*partition_name*########### #######DEFAULT#################### ####DEFAULT####################### ########## ################## ###########

#################SQL####CREATE##########_*nameN*LIKE##INCLUDING ALL EXCLUDED INDEXES EXCLUDED IDENTITY EXCLUDED################### ####### #######ID############################## ###################### ###### #CREATE###########OF####### ################################ ######### ########################## ################################## ############# #################

##
############ACCESS EXCLUSIVE######### ########################## #########################

MERGE PARTITIONS (*partition_name1*, *partition_name2* [, ...]) INTO *partition_name*

################################1################ ######################## #######*partition_name1*#*partition_name2*#...#####DEFAULT###########:

- ##########################*partition_name1*#*partition_name2* [#...]###### ##########1################################ ################ #*partition_name*#########
- ##################*partition_name1*#*partition_name2* [#*partition_name*]#####################*partition_name*###########

If DEFAULT partition is in the list of partitions *partition_name1*, *partition_name2* [, ...]:

- #####[55.769231]#####*action*########## ############ *partition_name* ###### ##########
- ### ######################_################1################ ############## #########################

##################SQL##########################CREATE###########_#
##LIKE##INCLUDING ALL EXCLUDED INDEXES EXCLUDED IDENTITY EXCLUDED
#######ID############################## ########################
#CREATE###########OF####### ################################## ###########
########################### ################################## ###############
################

##
#############ACCESS EXCLUSIVE######### ########################### #########################

#####[82.014388]#RENAME#SET SCHEMA#ATTACH PARTITION#DETACH PARTITION##
#####################################ALTER TABLE########################## #
################################### ####################################1#####
######RENAME#SET SCHEMA#ATTACH PARTITION#DETACH PARTITION#SPLIT
PARTITION#MERGE PARTITIONS################ALTER TABLE###################
################## #####1 ################################### ########### 1 ####
############################

ALTER TABLE################################## ##############################
################CREATE################ ####################################
########## ## ################
#######SET ROLE##################################CREATE############### #####
###########################
########## ############OF###################USAGE###############

#####

IF EXISTS

######################## ####################

name

############################## ########ONLY######################### ONLY#
###############*#######
##############################

column_name

###############

new_column_name

#############

new_name

#############

data_type

##############################

table_constraint

#################

constraint_name

##################

CASCADE

##
############5.15####

RESTRICT

################################## ##############

trigger_name

#####################

ALL

########################### ##
###########################

USER

##
###

index_name

##############

storage_parameter

#################

value

################### ################################

parent_table

##

new_owner

##################

new_access_method

########################

new_tablespace

######################

new_schema

####################

partition_name

##
################

partition_bound_spec

######################### ###########CREATE TABLE#########

##

COLUMN##################

ADD COLUMN############DEFAULT###
################## DEFAULT############NULL###### #######################

####DEFAULT## #############
#####USING##
######### ## #
####################
##########text##varchar#(######)######################## ####################
################################## #######2################

CHECK####NOT NULL###
######

###

###ALTER TABLE##1##################
#########

##ALTER TABLE ADD CONSTRAINT#
################################### NOT VALID##############################
######### NOT VALID#####ADD CONSTRAINT######################### ###
##VALIDATE CONSTRAINT########################## #######################
###
################SHARE UPDATE EXCLUSIVE########## (##############
#######ROW SHARE########) ##NOT
VALID#VALIDATE CONSTRAINT############### ################################
######VALIDATE CONSTRAINT###########################

DROP COLUMN###############SQL############## ##############################
#######NULL####### ################### ####################################
#################### ############################

##ALTER TABLE############ #####
#########NULL#########

############ALTER TABLE#MVCC########### ################################
####13.6#########

SET DATA TYPE#USING################################## #######################
######### ###########SET DATA TYPE############ ########USING#############
################ ## ##################
####################USING#########SET DATA TYPE######################## ###
###DROP DEFAULT###########ALTER TYPE#######SET DEFAULT################
######## ############################

###
#
##################################### #############CHECK################
####################CHECK################## ############################
#########SELECT#############SELECT##################################
###################### ##############ALTER TABLE ONLY##########

####DROP COLUMN###
############ ######DROP COLUMN#####ALTER TABLE ONLY ... DROP COLUMN####

################ ## ######DROP COLUMN# ###################### ##

IDENTITY########ADD GENERATED#SET#DROP IDENTITY#####CLUSTER#OWNER## #TABLESPACE################### ######ONLY################## ############## ##(ONLY##########)## ##### ##NO INHERIT####CHECK################

####################################

##############CREATE TABLE######### #5######################

varchar##############

```
ALTER TABLE distributors ADD COLUMN address varchar(30);
```

##################NULL#############

#######NULL########

```
ALTER TABLE measurements
  ADD COLUMN mtime timestamp with time zone DEFAULT now();
```

##

#############################

```
ALTER TABLE transactions
  ADD COLUMN status varchar(30) DEFAULT 'old',
  ALTER COLUMN status SET default 'current';
```

#####old#########################current###### ###ALTER TABLE#####2######## ##############

#############

```
ALTER TABLE distributors DROP COLUMN address RESTRICT;
```

1########2############

```
ALTER TABLE distributors
    ALTER COLUMN address TYPE varchar(80),
    ALTER COLUMN name TYPE varchar(100);
```

USING#######Unix#########integer####timestamp with time zone#######

```
ALTER TABLE foo
    ALTER COLUMN foo_timestamp SET DATA TYPE timestamp with time zone
    USING
        timestamp with time zone 'epoch' + foo_timestamp * interval '1
 second';
```

########### ##

```
ALTER TABLE foo
```

```
    ALTER COLUMN foo_timestamp DROP DEFAULT,
    ALTER COLUMN foo_timestamp TYPE timestamp with time zone
    USING
        timestamp with time zone 'epoch' + foo_timestamp * interval '1
 second',
    ALTER COLUMN foo_timestamp SET DEFAULT now();

##############


ALTER TABLE distributors RENAME COLUMN address TO city;

################


ALTER TABLE distributors RENAME TO suppliers;

###############


ALTER TABLE distributors RENAME CONSTRAINT zipchk TO zip_check;

###NULL########


ALTER TABLE distributors ALTER COLUMN street SET NOT NULL;

####NULL########


ALTER TABLE distributors ALTER COLUMN street DROP NOT NULL;

################CHECK########


ALTER TABLE distributors ADD CONSTRAINT zipchk CHECK (char_length(zipcode)
 = 5);

#######################################


ALTER TABLE distributors ADD CONSTRAINT zipchk CHECK (char_length(zipcode)
 = 5) NO INHERIT;

################################

#################CHECK########


ALTER TABLE distributors DROP CONSTRAINT zipchk;

1####################


ALTER TABLE ONLY distributors DROP CONSTRAINT zipchk;

##########################

#################


ALTER TABLE distributors ADD CONSTRAINT distfk FOREIGN KEY (address)
 REFERENCES addresses (address);

##########################################
```

```
ALTER TABLE distributors ADD CONSTRAINT distfk FOREIGN KEY (address)
 REFERENCES addresses (address) NOT VALID;
ALTER TABLE distributors VALIDATE CONSTRAINT distfk;

#######################


ALTER TABLE distributors ADD CONSTRAINT dist_id_zipcode_key UNIQUE
 (dist_id, zipcode);

######################### 1##############1##################


ALTER TABLE distributors ADD PRIMARY KEY (dist_id);

####################


ALTER TABLE distributors SET TABLESPACE fasttablespace;

##################


ALTER TABLE myschema.distributors SET SCHEMA yourschema;

##########################################


CREATE UNIQUE INDEX CONCURRENTLY dist_id_temp_idx ON distributors
 (dist_id);
ALTER TABLE distributors DROP CONSTRAINT distributors_pkey,
    ADD CONSTRAINT distributors_pkey PRIMARY KEY USING INDEX
 dist_id_temp_idx;

###########################


ALTER TABLE measurement
    ATTACH PARTITION measurement_y2016m07 FOR VALUES FROM ('2016-07-01')
 TO ('2016-08-01');

############################


ALTER TABLE cities
    ATTACH PARTITION cities_ab FOR VALUES IN ('a', 'b');

##############################


ALTER TABLE orders
    ATTACH PARTITION orders_p4 FOR VALUES WITH (MODULUS 4, REMAINDER 3);

###############################


ALTER TABLE cities
    ATTACH PARTITION cities_partdef DEFAULT;

############################


ALTER TABLE measurement
    DETACH PARTITION measurement_y2015m12;
```

#####[67.796610]############################ ##################1##############
###########

```
ALTER TABLE sales_range SPLIT PARTITION sales_feb_mar_apr2023 INTO
   (PARTITION sales_feb2023 FOR VALUES FROM ('2023-02-01') TO
 ('2023-03-01'),
    PARTITION sales_mar2023 FOR VALUES FROM ('2023-03-01') TO
 ('2023-04-01'),
    PARTITION sales_apr2023 FOR VALUES FROM ('2023-04-01') TO
 ('2023-05-01'));
```

#####[67.241379]############################ ################################
###########

```
ALTER TABLE sales_list SPLIT PARTITION sales_all INTO
   (PARTITION sales_west FOR VALUES IN ('Lisbon', 'New York', 'Madrid'),
    PARTITION sales_east FOR VALUES IN ('Bejing', 'Delhi', 'Vladivostok'),
    PARTITION sales_central FOR VALUES IN ('Warsaw', 'Berlin', 'Kyiv'));
```

################1##########################

```
ALTER TABLE sales_list MERGE PARTITIONS (sales_west, sales_east,
 sales_central)
    INTO sales_all;
```

###

#USING INDEX####ADD#DROP [COLUMN]#DROP IDENTITY#RESTART#SET DEFAULT#
#USING####SET DATA TYPE#SET GENERATED#SET *sequence_option*#####SQL######
#######SQL####PostgreSQL###### ######ALTER TABLE##################PostgreSQL##
####

ALTER TABLE DROP COLUMN#####1### ###
PostgreSQL######SQL#######################

####

CREATE TABLE

ALTER TABLESPACE

ALTER TABLESPACE — ##############

##

```
ALTER TABLESPACE name RENAME TO new_name
ALTER TABLESPACE name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER TABLESPACE name SET ( tablespace_option = value [, ... ] )
ALTER TABLESPACE name RESET ( tablespace_option [, ... ] )
```

##

ALTER TABLESPACE##########################

################################### ########################SET ROLE#########
##################################

#####

name

##############

new_name

############## pg_##

new_owner

###############

tablespace_option

######################### ####
#seq_page_cost#random_page_cost#effective_io_concurrency#maintenance_io_concu
############## ####################################
(seq_page_cost#random_page_cost#effective_io_concurrency#maintenance_io_concurrency#
#)## ####
##########I/O##

#

######index_space#fast_raid###########

```
ALTER TABLESPACE index_space RENAME TO fast_raid;
```

######index_space##########

```
ALTER TABLESPACE index_space OWNER TO mary;
```

###

##SQL##ALTER TABLESPACE#######

####

CREATE TABLESPACE, DROP TABLESPACE

ALTER TEXT SEARCH CONFIGURATION

ALTER TEXT SEARCH CONFIGURATION — ################

##

```
ALTER TEXT SEARCH CONFIGURATION name
    ADD MAPPING FOR token_type [, ... ] WITH dictionary_name [, ... ]
ALTER TEXT SEARCH CONFIGURATION name
    ALTER MAPPING FOR token_type [, ... ] WITH dictionary_name [, ... ]
ALTER TEXT SEARCH CONFIGURATION name
    ALTER MAPPING REPLACE old_dictionary WITH new_dictionary
ALTER TEXT SEARCH CONFIGURATION name
    ALTER MAPPING FOR token_type [, ... ] REPLACE old_dictionary
 WITH new_dictionary
ALTER TEXT SEARCH CONFIGURATION name
    DROP MAPPING [ IF EXISTS ] FOR token_type [, ... ]
ALTER TEXT SEARCH CONFIGURATION name RENAME TO new_name
ALTER TEXT SEARCH CONFIGURATION name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
ALTER TEXT SEARCH CONFIGURATION name SET SCHEMA new_schema
```

##

ALTER TEXT SEARCH CONFIGURATION################## ######################
##########################

ALTER TEXT SEARCH CONFIGURATION#########################

#####

name

#########################

token_type

####################

dictionary_name

################################ ############################

old_dictionary

#######################

new_dictionary

old_dictionary##################

new_name

################

new_owner

#################

new_schema

##################

ADD MAPPING FOR################################ ########################## ###### ALTER MAPPING FOR##################################### ALTER MAPPING REPLACE#######*old_dictionary*#*new_dictionary*####### FOR################ ##################################### DROP MAPPING######################### ####### ## IF EXISTS################### ###############

# #

#####my_config##english############english###swedish########

```
ALTER TEXT SEARCH CONFIGURATION my_config
  ALTER MAPPING REPLACE english WITH swedish;
```

###

##SQL##ALTER TEXT SEARCH CONFIGURATION#######

####

CREATE TEXT SEARCH CONFIGURATION, DROP TEXT SEARCH CONFIGURATION

ALTER TEXT SEARCH DICTIONARY

ALTER TEXT SEARCH DICTIONARY — ################

##

```
ALTER TEXT SEARCH DICTIONARY name (
    option [ = value ] [, ... ]
)
ALTER TEXT SEARCH DICTIONARY name RENAME TO new_name
ALTER TEXT SEARCH DICTIONARY name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
ALTER TEXT SEARCH DICTIONARY name SET SCHEMA new_schema
```

##

ALTER TEXT SEARCH DICTIONARY################## ########################### ##################

ALTER TEXT SEARCH DICTIONARY#######################

#####

name

###########################

option

##############################

value

########################### ## ####

new_name

################

new_owner

#################

new_schema

##################

################################

#

################################ ###################

```
ALTER TEXT SEARCH DICTIONARY my_dict ( StopWords = newrussian );
```

```
#############dutch(#####)##########################

ALTER TEXT SEARCH DICTIONARY my_dict ( language = dutch, StopWords );

################################

ALTER TEXT SEARCH DICTIONARY my_dict ( dummy );

################################################## ##############################
##ALTER#################################### ################################
####
```

###

```
##SQL##ALTER TEXT SEARCH DICTIONARY#######
```

####

CREATE TEXT SEARCH DICTIONARY, DROP TEXT SEARCH DICTIONARY

ALTER TEXT SEARCH PARSER

ALTER TEXT SEARCH PARSER — ################

##

```
ALTER TEXT SEARCH PARSER name RENAME TO new_name
ALTER TEXT SEARCH PARSER name SET SCHEMA new_schema
```

##

ALTER TEXT SEARCH PARSER################### ##############################

ALTER TEXT SEARCH PARSER##########################

#####

name

##########################

new_name

#################

new_schema

#################

###

##SQL##ALTER TEXT SEARCH PARSER#######

####

CREATE TEXT SEARCH PARSER, DROP TEXT SEARCH PARSER

ALTER TEXT SEARCH TEMPLATE

ALTER TEXT SEARCH TEMPLATE — ##################

##

```
ALTER TEXT SEARCH TEMPLATE name RENAME TO new_name
ALTER TEXT SEARCH TEMPLATE name SET SCHEMA new_schema
```

##

ALTER TEXT SEARCH TEMPLATE###################### ########################
#######

ALTER TEXT SEARCH TEMPLATE#########################

#####

name

##############################

new_name

####################

new_schema

######################

###

##SQL##ALTER TEXT SEARCH TEMPLATE#######

####

CREATE TEXT SEARCH TEMPLATE, DROP TEXT SEARCH TEMPLATE

ALTER TRIGGER

ALTER TRIGGER — #########

##

```
ALTER TRIGGER name ON table_name RENAME TO new_name
ALTER TRIGGER name ON table_name [ NO ] DEPENDS ON
 EXTENSION extension_name
```

##

ALTER TRIGGER###################

RENAME################################ ######################################
##################

DEPENDS ON EXTENSION###
######

###

#####

name

###################

table_name

###################

new_name

###########

extension_name

########(####NO###################)######## ################################
#########

##

####################ALTER TABLE###### ALTER TRIGGER####### ALTER
TRIGGER##

#

###############

```
ALTER TRIGGER emp_stamp ON emp RENAME TO emp_track_chgs;
```

####################

```
ALTER TRIGGER emp_stamp ON emp DEPENDS ON EXTENSION emplib;
```

###

`ALTER TRIGGER`####SQL####PostgreSQL######

####

ALTER TABLE

ALTER TYPE

ALTER TYPE — ########

##

```
ALTER TYPE name OWNER TO { new_owner | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
ALTER TYPE name RENAME TO new_name
ALTER TYPE name SET SCHEMA new_schema
ALTER TYPE name RENAME ATTRIBUTE attribute_name TO new_attribute_name
 [ CASCADE | RESTRICT ]
ALTER TYPE name action [, ... ]
ALTER TYPE name ADD VALUE [ IF NOT EXISTS ] new_enum_value [ { BEFORE |
 AFTER } neighbor_enum_value ]
ALTER TYPE name RENAME VALUE existing_enum_value TO new_enum_value
ALTER TYPE name SET ( property = value [, ... ] )
```

###action###########

```
    ADD ATTRIBUTE attribute_name data_type [ COLLATE collation ] [ CASCADE
 | RESTRICT ]
    DROP ATTRIBUTE [ IF EXISTS ] attribute_name [ CASCADE | RESTRICT ]
    ALTER ATTRIBUTE attribute_name [ SET DATA ] TYPE data_type
 [ COLLATE collation ] [ CASCADE | RESTRICT ]
```

##

ALTER TYPE############## ###########

OWNER

#################

RENAME

###############

SET SCHEMA

###################

RENAME ATTRIBUTE

################### ################

ADD ATTRIBUTE

######CREATE TYPE#######################

DROP ATTRIBUTE [IF EXISTS]

################## IF EXISTS############################## ##############
####

ALTER ATTRIBUTE ... SET DATA TYPE

###################

ADD VALUE [IF NOT EXISTS] [BEFORE | AFTER]

################### ######################BEFORE###AFTER################ ## ###########################

IF NOT EXISTS## ################### #### ################################

RENAME VALUE

#################### ###################### ############################## ########

SET (*property* = *value* [, ...])

#################### CREATE TYPE######################## ###############

- RECEIVE###################NONE################### ################### ######
- SEND###################NONE################### ######################## ##
- TYPMOD_IN###################NONE################### ################### #######
- TYPMOD_OUT###################NONE################### ############### ##########
- ANALYZE#######################NONE################### ############### ##########
- SUBSCRIPT#######################NONE################### ############## ############
- STORAGE STORAGE#plain#extended#external#main######(############ ###65.2########)# #######plain############################(####C###### TOAST##################)#plain#########################(######TOAST# #######################)# ## ###########TOAST####################### ########TOAST########ALTER TABLE#########

############CREATE TYPE######### ##################################### ##########

ADD ATTRIBUTE#DROP ATTRIBUTE#ALTER ATTRIBUTE########################## ##### ##

ALTER TYPE###################### ########################CREATE####### #### ###################SET ROLE##############################CREATE########### ### ## ########################### ######## ##################################USAGE##############

#####

name

#########################

new_name

##########

new_owner

###############

new_schema

###########

attribute_name

###################

new_attribute_name

##############

data_type

#############################

new_enum_value

################################### ##############################

neighbor_enum_value

##############################

existing_enum_value

##################### ##############################

property

################################

CASCADE

##

RESTRICT

################################## ###########

##

ALTER TYPE ... ADD VALUE##

#BEFORE###AFTER################### ################################## ### ##OID################################ ###################

#

#############

```
ALTER TYPE electronic_mail RENAME TO email;
```

email#####joe######

```
ALTER TYPE email OWNER TO joe;
```

email######customers######

```
ALTER TYPE email SET SCHEMA customers;
```

###############

```
ALTER TYPE compfoo ADD ATTRIBUTE f3 int;
```

#######################

```
ALTER TYPE colors ADD VALUE 'orange' AFTER 'red';
```

##############

```
ALTER TYPE colors RENAME VALUE 'purple' TO 'mauve';
```

##############I/O########

```
CREATE FUNCTION mytypesend(mytype) RETURNS bytea ...;
CREATE FUNCTION mytyperecv(internal, oid, integer) RETURNS mytype ...;
ALTER TYPE mytype SET (
    SEND = mytypesend,
    RECEIVE = mytyperecv
);
```

###

##################SQL###### #####PostgreSQL######

####

CREATE TYPE, DROP TYPE

ALTER USER

ALTER USER — ##############

##

```
ALTER USER role_specification [ WITH ] option [ ... ]


option########

      SUPERUSER | NOSUPERUSER
    | CREATEDB | NOCREATEDB
    | CREATEROLE | NOCREATEROLE
    | INHERIT | NOINHERIT
    | LOGIN | NOLOGIN
    | REPLICATION | NOREPLICATION
    | BYPASSRLS | NOBYPASSRLS
    | CONNECTION LIMIT connlimit
    | [ ENCRYPTED ] PASSWORD 'password' | PASSWORD NULL
    | VALID UNTIL 'timestamp'

ALTER USER name RENAME TO new_name

ALTER USER { role_specification | ALL } [ IN DATABASE database_name ]
 SET configuration_parameter { TO | = } { value | DEFAULT }
ALTER USER { role_specification | ALL } [ IN DATABASE database_name ]
 SET configuration_parameter FROM CURRENT
ALTER USER { role_specification | ALL } [ IN DATABASE database_name ]
 RESET configuration_parameter
ALTER USER { role_specification | ALL } [ IN DATABASE database_name ]
 RESET ALL


###role_specification#########

    role_name
  | CURRENT_ROLE
  | CURRENT_USER
  | SESSION_USER
```

##

ALTER USER#ALTER ROLE##########

###

ALTER USER###PostgreSQL###### ##SQL####################

####

ALTER ROLE

ALTER USER MAPPING

ALTER USER MAPPING — ##############

##

```
ALTER USER MAPPING FOR { user_name | USER | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER | PUBLIC }
    SERVER server_name
    OPTIONS ( [ ADD | SET | DROP ] option ['value'] [, ... ] )
```

##

ALTER USER MAPPING################

########USAGE##################

#####

user_name

############ CURRENT_ROLE#CURRENT_USER#USER############## PUBLIC##### ##################################

server_name

##############

OPTIONS ([ADD | SET | DROP] *option* ['*value*'] [, ...])

################## ################################ ADD#SET#DROP######### #### ##############ADD######## ################## ######################### #####

#

###foo#######bob############

```
ALTER USER MAPPING FOR bob SERVER foo OPTIONS (SET password 'public');
```

###

ALTER USER MAPPING#ISO/IEC 9075-9(SQL/MED)###### ############### ####FOR###### ###### CREATE USER MAPPING#DROP USER MAPPING##FOR###############IBM DB2## ###SQL/MED############ALTER USER MAPPING############PostgreSQL############# ################

####

CREATE USER MAPPING, DROP USER MAPPING

ALTER VIEW

ALTER VIEW — ##########

##

```
ALTER VIEW [ IF EXISTS ] name ALTER [ COLUMN ] column_name SET
 DEFAULT expression
ALTER VIEW [ IF EXISTS ] name ALTER [ COLUMN ] column_name DROP DEFAULT
ALTER VIEW [ IF EXISTS ] name OWNER TO { new_owner | CURRENT_ROLE |
 CURRENT_USER | SESSION_USER }
ALTER VIEW [ IF EXISTS ] name RENAME [ COLUMN ] column_name
 TO new_column_name
ALTER VIEW [ IF EXISTS ] name RENAME TO new_name
ALTER VIEW [ IF EXISTS ] name SET SCHEMA new_schema
ALTER VIEW [ IF EXISTS ] name SET ( view_option_name [= view_option_value]
 [, ... ] )
ALTER VIEW [ IF EXISTS ] name RESET ( view_option_name [, ... ] )
```

##

ALTER VIEW################## ######################CREATE OR REPLACE VIEW# ##########

ALTER VIEW########################## ############################CREATE### ########### ########################SET ROLE################################ #CREATE############## ## ######### ####################################

#####

name

#####################

column_name

##########

new_column_name

################

IF EXISTS

#################### ####################

SET/DROP DEFAULT

######################### ## #INSERT###UPDATE############ ## ##

new_owner

##################

new_name

###########

new_schema

##############

SET (*view_option_name* [= *view_option_value*] [, ...])
RESET (*view_option_name* [, ...])

#################### #######################

check_option (enum)

####check option####### ##local###cascaded###############

security_barrier (boolean)

####security-barrier######### ##true#false#################

security_invoker (boolean)

####security-invoker######### ##true#false#################

##

##########ALTER TABLE################### ################################
#ALTER TABLE########

#

###foo####bar#######

```
ALTER VIEW foo RENAME TO bar;
```

#######################

```
CREATE TABLE base_table (id int, ts timestamptz);
CREATE VIEW a_view AS SELECT * FROM base_table;
ALTER VIEW a_view ALTER COLUMN ts SET DEFAULT now();
INSERT INTO base_table(id) VALUES(1);  -- ts will receive a NULL
INSERT INTO a_view(id) VALUES(2);  -- ts will receive the current time
```

###

ALTER VIEW###SQL####PostgreSQL######

####

CREATE VIEW, DROP VIEW

ANALYZE

ANALYZE — ################

##

```
ANALYZE [ ( option [, ...] ) ] [ table_and_columns [, ...] ]
```

option###############

```
    VERBOSE [ boolean ]
    SKIP_LOCKED [ boolean ]
    BUFFER_USAGE_LIMIT size
```

###table_and_columns#########

```
    table_name [ ( column_name [, ...] ) ]
```

##

ANALYZE################################pg_statistic############## ######## ######################################

table_and_columns########ANALYZE### ############ ########ANALYZE################## ############################ ##########

#####

VERBOSE

##############

SKIP_LOCKED

##################ANALYZE## ####################### ### ######################ANALYZE######################## #####ANALYZE###### ######################################ANALYZE######################### ##################

BUFFER_USAGE_LIMIT

ANALYZE############################## ################################### ############ 0######################### ################ ##ANALYZE#vacuum_buffer_usage_limit######## #######ANALYZE################ ### ####128 kB######16 GB###

boolean

########################### ##############TRUE#ON###1################ #FALSE#OFF###0########## *boolean*##################TRUE########

size

################### ################B(###)#kB(#####)#MB(#####)#GB(#####)## #TB(#####)#####1###########################

table_name

########################### ##
########################### ##
######################

column_name

############ ############

##

VERBOSE########ANALYZE######################## ##########################

##

############################MAINTAIN####### ################################
################## ANALYZE################################

########################## #############ANALYZE################ ########
#ANALYZE############################

######PostgreSQL#################24.1.6#######################################
############ ####################ANALYZE#################################
#########
######VACUUM#ANALYZE##1#1################################## ###############
############

######ANALYZE#####search_path#####pg_catalog, pg_temp#######

#####[87.943262]#ANALYZE########################## ##########################
######### ######ANALYZE###DDL#########
########

###ANALYZE## ANALYZE#######
##
###################### #24######################

#########ANALYZE## ###############
######################   ##
#ANALYZE###################### ######EXPLAIN############################## #
######################ANALYZE################################ ###############
##ANALYZE####################

########default_statistics_target##########ALTER TABLE ... ALTER COLUMN ...
SET STATISTICS### ###################
############################### #########100### ##################ANALYZE####
##pg_statistic#################################### ####0##################
########## ###WHERE##GROUP BY##ORDER BY########################## #######
##################

################ANALYZE##
################

ANALYZE#######1#################### ##################################
############################## ######################################ALTER
TABLE ... ALTER COLUMN ... SET (n_distinct = ...)###########

######################ANALYZE#####2######1##################2###########
################### ############################2############# ###############

########## ## #
#######################################ANALYZE###################

###############ANALYZE###
######## ##1########## ####################
##########(################)####################

#################
############ANALYZE###############

##########################ANALYZE##
###

#######################ANALYZE######################### ####################

ANALYZE####################pg_stat_progress_analyze########## ###27.4.1#####
#####

###

##SQL##ANALYZE########

######PostgreSQL#####11##################################

```
ANALYZE [ VERBOSE ] [ table_and_columns [, ...] ]
```

####

VACUUM, vacuumdb, 19.4.4, 24.1.6, 27.4.1

BEGIN

BEGIN — ################

##

```
BEGIN [ WORK | TRANSACTION ] [ transaction_mode [, ...] ]
```

transaction_mode###########

```
    ISOLATION LEVEL { SERIALIZABLE | REPEATABLE READ | READ COMMITTED |
 READ UNCOMMITTED }
    READ WRITE | READ ONLY
    [ NOT ] DEFERRABLE
```

##

BEGIN#################### BEGIN###############COMMIT####ROLLBACK###### ###################### ######BEGIN#######PostgreSQL#################### ## ## #########

############################ ######################CPU#################### ##1## ##

###SET TRANSACTION################ ##

#####

WORK
TRANSACTION

  ###########################

BEGIN################SET TRANSACTION##########

##

START TRANSACTION##BEGIN###########

###############COMMIT###ROLLBACK##########

#############BEGIN#################### ######################## ########### ################################### #####SAVEPOINT###########

###############*transaction_modes*################

#

##################

```
BEGIN;
```

###

BEGIN#PostgreSQL###### ##SQL#START TRANSACTION######### ################### ###########

DEFERRABLE *transaction_mode*#PostgreSQL########

####SQL###BEGIN######################## ## ################

####

COMMIT, ROLLBACK, START TRANSACTION, SAVEPOINT

CALL

CALL — ##########

##

```
CALL name ( [ argument ] [, ...] )
```

##

CALL##############

##

#####

name

##########################

argument

#######################

####name => value########################## ##########################4.3#### #######

####OUT## #######OUT########## ######################NULL############ (OUT####################PostgreSQL#### ##########################)

##

#####################################EXECUTE#############

##########################SELECT#######

##############CALL## #############CALL# ############################

PL/pgSQL##CALL######################## 41.6.3##########

#

```
CALL do_db_maintenance();
```

###

CALL##################SQL######### ####################################

####

CREATE PROCEDURE

CHECKPOINT

CHECKPOINT — ##########################

##

```
CHECKPOINT
```

##

######################28.5##########

CHECKPOINT##################19.5.2### ################# #######CHECKPOINT#################

#############CHECKPOINT###########################28.5##########

CHECKPOINT#######################pg_checkpoint################

###

CHECKPOINT######PostgreSQL######

CLOSE

CLOSE — #######

##

```
CLOSE { name | ALL }
```

##

CLOSE######################## ############################### ################ ########

########COMMIT####ROLLBACK################################### ROLLBACK## ### ######################### ############CLOSE##

#####

name

#########################

ALL

#############

##

PostgreSQL###########OPEN######## ###################### #########DECLARE###### #####

pg_cursors##

##################################CLOSE############# #####################

#

####liahona######

```
CLOSE liahona;
```

###

CLOSE###SQL############# ####CLOSE ALL#PostgreSQL######

####

DECLARE, FETCH, MOVE

CLUSTER

CLUSTER — ####################

##

```
CLUSTER [ ( option [, ...] ) ] [ table_name [ USING index_name ] ]

where option can be one of:

    VERBOSE [ boolean ]
```

##

CLUSTER##*index_name*################*table_name*################## ##PostgreSQL###### ############*table_name*################

######1######## ######### ######################### ###################################### ######### ################################## ### #fillfactor#######100%##

##############PostgreSQL######################## CLUSTER *table_name*###### ################################## ##ALTER TABLE#CLUSTER####SET WITHOUT CLUSTER###

#####[79.794521]#*table_name*######CLUSTER################################## ## #####CLUSTER###### ################## ######*table_name*#####CLUSTER########################### ############################ #####CLUSTER######################

#################ACCESS EXCLUSIVE############ ######CLUSTER############## ########################

#####

table_name

    ####################

index_name

    ############

VERBOSE

    ############################

boolean

    ########################## ##############TRUE#ON###1################> #FALSE#OFF###0########## *boolean*##################TRUE########

##

############################MAINTAIN#######

##CLUSTER############ ##########################
##1#################CLUSTER######## ##
###

CLUSTER#######################################B-Tree#########################
#################### ##

######CLUSTER#####search_path#####pg_catalog, pg_temp#######

######################
####### ##

####################################
############ ##
#enable_sort#off#######################

#########maintenance_work_mem################### ####CLUSTER###########RAM#
#################

##ANALYZE############## #####
################################

CLUSTER###
##########################CLUSTER############################

CLUSTER###################pg_stat_progress_cluster######### ###27.4.2#####
#####

#########
########################### ############CLUSTER#######################

# #

######employees_ind##########emp#########

```
CLUSTER employees USING employees_ind;
```

#######################employees#########

```
CLUSTER employees;
```

##################################

```
CLUSTER;
```

###

##SQL##CLUSTER#######

######PostgreSQL 17#######################

```
CLUSTER [ VERBOSE ] [ table_name [ USING index_name ] ]
```

######PostgreSQL 8.3#########################

```
CLUSTER index_name ON table_name
```

####

clusterdb, 27.4.2

COMMENT

COMMENT — ########################

##

```
COMMENT ON
{
  ACCESS METHOD object_name |
  AGGREGATE aggregate_name ( aggregate_signature ) |
  CAST (source_type AS target_type) |
  COLLATION object_name |
  COLUMN relation_name.column_name |
  CONSTRAINT constraint_name ON table_name |
  CONSTRAINT constraint_name ON DOMAIN domain_name |
  CONVERSION object_name |
  DATABASE object_name |
  DOMAIN object_name |
  EXTENSION object_name |
  EVENT TRIGGER object_name |
  FOREIGN DATA WRAPPER object_name |
  FOREIGN TABLE object_name |
  FUNCTION function_name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] |
  INDEX object_name |
  LARGE OBJECT large_object_oid |
  MATERIALIZED VIEW object_name |
  OPERATOR operator_name (left_type, right_type) |
  OPERATOR CLASS object_name USING index_method |
  OPERATOR FAMILY object_name USING index_method |
  POLICY policy_name ON table_name |
  [ PROCEDURAL ] LANGUAGE object_name |
  PROCEDURE procedure_name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] |
  PUBLICATION object_name |
  ROLE object_name |
  ROUTINE routine_name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
 |
  RULE rule_name ON table_name |
  SCHEMA object_name |
  SEQUENCE object_name |
  SERVER object_name |
  STATISTICS object_name |
  SUBSCRIPTION object_name |
  TABLE object_name |
  TABLESPACE object_name |
  TEXT SEARCH CONFIGURATION object_name |
  TEXT SEARCH DICTIONARY object_name |
  TEXT SEARCH PARSER object_name |
  TEXT SEARCH TEMPLATE object_name |
  TRANSFORM FOR type_name LANGUAGE lang_name |
  TRIGGER trigger_name ON table_name |
  TYPE object_name |
  VIEW object_name
} IS { string_literal | NULL }


###aggregate_signature#########

* |
```

```
[ argmode ] [ argname ] argtype [ , ... ] |
[ [ argmode ] [ argname ] argtype [ , ... ] ] ORDER BY [ argmode ]
 [ argname ] argtype [ , ... ]
```

##

COMMENT##########################

####################1###### ################################COMMENT######### ##### #######################NULL######### ############################

#################SHARE UPDATE EXCLUSIVE##########

################COMMENT ON ROLE#### ##CREATEROLE###############ADMIN OPTION################### ### #########

######psql#\d############## obj_description()#col_description()#shobj_description#######psql####### ## ## 9.80##########

#####

object_name
relation_name.column_name
aggregate_name
constraint_name
function_name
operator_name
policy_name
procedure_name
routine_name
rule_name
trigger_name

> ################### ###############(#########)############### ############ ##*relation_name*####################################

table_name
domain_name

> ##

source_type

> #################

target_type

> ##################

argmode

> #####################IN#OUT#INOUT#VARIADIC####### #########IN### # #####################COMMENT#####OUT################# #####IN#INOUT## #VARIADIC##############

argname

> ###################### ######################COMMENT#################### #####

argtype

#######################

large_object_oid

##########OID###

left_type
right_type

####################### ####################NONE##########

PROCEDURAL

############

type_name

############

lang_name

###########

string_literal

############################

NULL

###########NULL#######

##

############################## ### ### ## ######################### ##################################

# #

####mytable###########

```
COMMENT ON TABLE mytable IS 'This is my table.';
```

###############

```
COMMENT ON TABLE mytable IS NULL;
```

###############

```
COMMENT ON ACCESS METHOD gin IS 'GIN index access method';
COMMENT ON AGGREGATE my_aggregate (double precision) IS 'Computes sample
 variance';
COMMENT ON CAST (text AS int4) IS 'Allow casts from text to int4';
COMMENT ON COLLATION "fr_CA" IS 'Canadian French';
COMMENT ON COLUMN my_table.my_column IS 'Employee ID number';
COMMENT ON CONVERSION my_conv IS 'Conversion to UTF8';
COMMENT ON CONSTRAINT bar_col_cons ON bar IS 'Constrains column col';
```

```
COMMENT ON CONSTRAINT dom_col_constr ON DOMAIN dom IS 'Constrains col of
 domain';
COMMENT ON DATABASE my_database IS 'Development Database';
COMMENT ON DOMAIN my_domain IS 'Email Address Domain';
COMMENT ON EVENT TRIGGER abort_ddl IS 'Aborts all DDL commands';
COMMENT ON EXTENSION hstore IS 'implements the hstore data type';
COMMENT ON FOREIGN DATA WRAPPER mywrapper IS 'my foreign data wrapper';
COMMENT ON FOREIGN TABLE my_foreign_table IS 'Employee Information in
 other database';
COMMENT ON FUNCTION my_function (timestamp) IS 'Returns Roman Numeral';
COMMENT ON INDEX my_index IS 'Enforces uniqueness on employee ID';
COMMENT ON LANGUAGE plpython IS 'Python support for stored procedures';
COMMENT ON LARGE OBJECT 346344 IS 'Planning document';
COMMENT ON MATERIALIZED VIEW my_matview IS 'Summary of order history';
COMMENT ON OPERATOR ^ (text, text) IS 'Performs intersection of two
 texts';
COMMENT ON OPERATOR - (NONE, integer) IS 'Unary minus';
COMMENT ON OPERATOR CLASS int4ops USING btree IS '4 byte integer operators
 for btrees';
COMMENT ON OPERATOR FAMILY integer_ops USING btree IS 'all integer
 operators for btrees';
COMMENT ON POLICY my_policy ON mytable IS 'Filter rows by users';
COMMENT ON PROCEDURE my_proc (integer, integer) IS 'Runs a report';
COMMENT ON PUBLICATION alltables IS 'Publishes all operations on all
 tables';
COMMENT ON ROLE my_role IS 'Administration group for finance tables';
COMMENT ON ROUTINE my_routine (integer, integer) IS 'Runs a routine (which
 is a function or procedure)';
COMMENT ON RULE my_rule ON my_table IS 'Logs updates of employee records';
COMMENT ON SCHEMA my_schema IS 'Departmental data';
COMMENT ON SEQUENCE my_sequence IS 'Used to generate primary keys';
COMMENT ON SERVER myserver IS 'my foreign server';
COMMENT ON STATISTICS my_statistics IS 'Improves planner row estimations';
COMMENT ON SUBSCRIPTION alltables IS 'Subscription for all operations on
 all tables';
COMMENT ON TABLE my_schema.my_table IS 'Employee Information';
COMMENT ON TABLESPACE my_tablespace IS 'Tablespace for indexes';
COMMENT ON TEXT SEARCH CONFIGURATION my_config IS 'Special word
 filtering';
COMMENT ON TEXT SEARCH DICTIONARY swedish IS 'Snowball stemmer for Swedish
 language';
COMMENT ON TEXT SEARCH PARSER my_parser IS 'Splits text into words';
COMMENT ON TEXT SEARCH TEMPLATE snowball IS 'Snowball stemmer';
COMMENT ON TRANSFORM FOR hstore LANGUAGE plpython3u IS 'Transform between
 hstore and Python dict';
COMMENT ON TRIGGER my_trigger ON my_table IS 'Used for RI';
COMMENT ON TYPE complex IS 'Complex number data type';
COMMENT ON VIEW my_view IS 'View of departmental costs';
```

###

##SQL##COMMENT#######

COMMIT

COMMIT — ################

##

```
COMMIT [ WORK | TRANSACTION ] [ AND [ NO ] CHAIN ]
```

##

COMMIT################### ##

#####

WORK
TRANSACTION

#######################

AND CHAIN

AND CHAIN##(SET TRANSACTION#########)########### ###########################

##

################ROLLBACK##########

############COMMIT################## ################### ############COMMIT AND CHAIN##############

#

#################################

```
COMMIT;
```

###

####COMMIT###SQL######### COMMIT TRANSACTION####PostgreSQL#######

####

BEGIN, ROLLBACK

COMMIT PREPARED

COMMIT PREPARED — ##############################

##

```
COMMIT PREPARED transaction_id
```

##

```
COMMIT PREPARED#########################
```

#####

transaction_id

  ##############################

##

############################

################################ #########################

#####################pg_prepared_xacts##############

#

```
foobar#################################
```

```
COMMIT PREPARED 'foobar';
```

###

COMMIT PREPARED#PostgreSQL###### ################################## ########## ########X/Open XA#######################SQL############

####

PREPARE TRANSACTION, ROLLBACK PREPARED

COPY

COPY — ####################

##

```
COPY table_name [ ( column_name [, ...] ) ]
    FROM { 'filename' | PROGRAM 'command' | STDIN }
    [ [ WITH ] ( option [, ...] ) ]
    [ WHERE condition ]

COPY { table_name [ ( column_name [, ...] ) ] | ( query ) }
    TO { 'filename' | PROGRAM 'command' | STDOUT }
    [ [ WITH ] ( option [, ...] ) ]


###option###########

    FORMAT format_name
    FREEZE [ boolean ]
    DELIMITER 'delimiter_character'
    NULL 'null_string'
    DEFAULT 'default_string'
    HEADER [ boolean | MATCH ]
    QUOTE 'quote_character'
    ESCAPE 'escape_character'
    FORCE_QUOTE { ( column_name [, ...] ) | * }
    FORCE_NOT_NULL { ( column_name [, ...] ) | * }
    FORCE_NULL { ( column_name [, ...] ) | * }
    ON_ERROR error_action
    ENCODING 'encoding_name'
    LOG_VERBOSITY verbosity
```

##

COPY######PostgreSQL################################ COPY TO################ ######## ###COPY FROM## #### ###COPY TO###SELECT####################

###############COPY TO######################### COPY FROM################################### ########### COPY FROM#################################

########COPY######PostgreSQL############################## ###########PostgreSQL ###(#############ID)################ ################################ PROGRAM## ## ######################## #####PostgreSQL#################################### STDIN#STDOUT#############################

COPY###################pg_stat_progress_copy########## ###27.4.3##########

##############COPY##################### ############################## ##ON_ERROR#################

#####

table_name

    ########################

column_name

##################### ##

query

#####[78.260870]#SELECT#VALUES#INSERT#UPDATE####DELETE#################
#############
#SELECT#VALUES#INSERT#UPDATE#DELETE####MERGE##### ################

######INSERT#UPDATE#DELETE#MERGE#######RETURNING####################
############ALSO####INSTEAD##########

filename

############## ##
Windows#######E''#####################################

PROGRAM

########### COPY FROM######################COPY TO#######################

###
############################### ##
######################

STDIN

##############################

STDOUT

#############################

boolean

########################### #############TRUE#ON###1#########FALSE#OFF##
#0####### ##*boolean*#############TRUE#######

FORMAT

########################### text#csv############binary### ######text###

FREEZE

#####[80.180180]#####VACUUM FREEZE##################################
############################
####
##############COPY FREEZE######## ############################
##VACUUM FREEZE################# ####################################
##
############################## ##########################COPY FREEZE#######
####### ########COPY FROM##########

###MVCC##############
##

DELIMITER

###################### ##################CSV########## ######1#############
########binary################

NULL

NULL############# ##############\N########N##CSV############### NULL## ## #######binary########## ######

> ##
>
> COPY FROM##################NULL########## COPY TO############## ############

DEFAULT

################## ##################################### ########COPY FROM#######binary##################

HEADER

################################## ######################## ############ #true(########)##################### #######MATCH##################### ## binary################### ####### MATCH######COPY FROM###########

QUOTE

############################## ############## ######1############### ####### #CSV#################

ESCAPE

#######QUOTE################################ ######QUOTE############### ##################### ######1############### ########CSV################

FORCE_QUOTE

###############NULL############### NULL############# *##########NULL###### ########### #######COPY TO#######CSV################

FORCE_NOT_NULL

#####[75.175644]##########NULL############# NULL########################## ######NULL#######0#################### ########COPY FROM#######CSV#### ############# ###############NULL############### ########NULL########### ####################NULL######0#################### *################### ########### ########COPY FROM######CSV##################

FORCE_NULL

#####[76.233184]############################NULL################NULL###### ## NULL###########################NULL####### #######COPY FROM####CSV# ################ ############################NULL######################## #NULL####### ########NULL########## ################CSV################# ## *######################### ########COPY FROM########CSV############## ###

ON_ERROR

error_action###stop########## ##ignore#################### #####stop###

######ignore#######FORMAT#text###csv####COPY FROM#########

##########1#########COPY FROM##############NOTICE##########
LOG_VERBOSITY#####verbose#######################################
##NOTICE############

ENCODING

#####*encoding_name*################ ##################################
##############

LOG_VERBOSITY

######COPY#################### default###verbose### verbose##########
#################

############ON_ERROR######ignore##########COPY FROM###########

WHERE

#####WHERE###############

WHERE *condition*

###*condition*#######boolean########## ######################## #########
###########################

######WHERE#########################COPY####################(#####
#VOLATILE##################)#

##

##########COPY######################

COPY *count*

count###########

##
psql######COPY ... TO STDOUT################psql#######\copy ... to stdout###################### ##################################

##

COPY TO###
####COPY *table* TO#SELECT * FROM ONLY *table*###########
#################COPY (SELECT * FROM *table*) TO ...########

COPY FROM############################INSTEAD OF INSERT##################
######

COPY TO#################SELECT####COPY FROM#################INSERT######
##########################

######################SELECT#####COPY *table* TO######## ######COPY FROM#
############################## #########INSERT########

COPY##/########## ################
#############

##PostgreSQL#################ID############################ ####PROGRAM#######
#####################################PostgreSQL#################### ########
#######COPY####################
#pg_read_server_files#pg_write_server_files#pg_execute_server_program#
##1###
############

COPY#psql#\copy########### \copy#COPY FROM STDIN#COPY TO STDOUT####
##psql###########################/########## ######\copy#######################
##

COPY################################## COPY TO############################
##COPY FROM#################################### ############################
##

PROGRAM###########SELinux#######################################

COPY FROM##

IDENTITY#######COPY FROM#####INSERT######OVERRIDING SYSTEM VALUE#######
######################

COPY#####DateStyle######## ########DateStyle############PostgreSQL######
##################COPY#####DateStyle#ISO############# #
##IntervalStyle#sql_standard######################### ########
#IntervalStyle#########################

##ENCODING###################
##################ENCODING##############################

######COPY FROM######################### ################################ ####
###################### ### ########
#####VACUUM##########

FORCE_NULL#FORCE_NOT_NULL#################### ###############NULL####
NULL###########NULL###############

#######

######

text#########################1####1################### ################## #####
##NULL#############NULL######
######### ##############################COPY FROM###########

######################\.##########1######## ################################
############## ####3.0###
###############

###########\###COPY## ############
##

#####NULL################COPY TO####### ###COPY FROM##################NULL#
############ ######\N####NULL#######\N##################### ####\\N###########

COPY FROM##############################

#####	##
\b	########ASCII 8#
\f	#####ASCII 12#

#####	##
\n	###ASCII 10#
\r	###ASCII 13#
\t	###ASCII 9#
\v	#####ASCII 11#
\##	##########1##3##8##
\x##	########x#######1##2##16##

###COPY TO##############8###16###

##########\.##############\N###########################

COPY#################################\n#\r################## ## ## ## ###COPY## #####Unix##Windows#######

############8###16###

COPY TO######Unix#######\n######### ###Microsoft Windows#######################################COPY########/####\r\n######### ##COPY TO STDOUT####\n######## COPY FROM#########/################## ##COPY FROM##########################

CSV##

##CSV##PostgreSQL################################CSV#################

########DELIMITER######### ######QUOTE###NULL###QUOTE#############QUOTE###ESCAPE## ##FORCE_QUOTE############NULL###########################

CSV####NULL######################### PostgreSQL#COPY###################NULL#NULL########################### ###NULL##############NULL########## ##############NULL################################2######""######## ################## FORCE_NOT_NULL############NULL###################### ##FORCE_NULL############NULL#########NULL############

CSV##################################\.################## ################################### ##1#########\.##

##
CSV################## ##################DELIMITER##CSV## ##########PostgreSQL#################CSV###########################

##

CSV##############################CSV############### ###################
#######1#######1###############

##

###########CSV## ######
##############################COPY##############################

######

binary####################################/############### #####CSV#########
###########################PostgreSQL################### ###################
#####smallint#################integer##################################
######

binary#######################0####################### #######################

##

7.4########PostgreSQL########################

#######

########15################################### ################

##

PGCOPY\n\377\r\n\0###11######## #########NULL#################### ####
###8## ###########NULL##
################################

########

################32############### #####0#LSB# ##31#MSB############
##
########### ####################16–31############# ########################
########################### #####################0–15############# #########
########################### ###1###########################0###########

###16

1####OID############0############# OID#########PostgreSQL##############
##################

########

#########################32######### ######0####################### ##########
########################### ##

################################
################################

##
##################################

####################16###
################### ##
32################ ##############################0############ #######NULL####
####-1######### ##NULL####################

################################

#####################################1############## ##########################
###################

##############################PostgreSQL###############*send ###*recv#######
#######src/backend/utils/adt#######################

#######OID########OID####################### ################################
OID######PostgreSQL################################

########

##########16###############-1######## #########################

####################-1################################## #######################
##########################

# #

#####################|#######################

```
COPY country TO STDOUT (DELIMITER '|');
```

```
######country###############
```

```
COPY country FROM '/usr1/proj/bray/sql/country_data';
```

```
###'A'####################
```

```
COPY (SELECT * FROM country WHERE country_name LIKE 'A%') TO '/usr1/proj/
bray/sql/a_list_countries.copy';
```

```
#####################################################
```

```
COPY country TO PROGRAM 'gzip > /usr1/proj/bray/sql/country_data.gz';
```

```
###STDIN#######################
```

```
AF      AFGHANISTAN
AL      ALBANIA
DZ      ALGERIA
ZM      ZAMBIA
ZW      ZIMBABWE
```

```
##############################
```

```
######################### ####Unix#######od  -c################# ######3#######
#####char(2)#2###text#3###integer### #####3###NULL####
```

```
0000000   P   G   C   O   P   Y  \n 377  \r  \n  \0  \0  \0  \0  \0  \0
0000020  \0  \0  \0  \0 003  \0  \0  \0 002   A   F  \0  \0  \0 013   A
0000040   F   G   H   A   N   I   S   T   A   N 377 377 377 377  \0 003
0000060  \0  \0  \0 002   A   L  \0  \0  \0 007   A   L   B   A   N   I
0000100   A 377 377 377 377  \0 003  \0  \0  \0 002   D   Z  \0  \0  \0
0000120 007   A   L   G   E   R   I   A 377 377 377 377  \0 003  \0  \0
0000140  \0 002   Z   M  \0  \0  \0 006   Z   A   M   B   I   A 377 377
0000160 377 377  \0 003  \0  \0  \0 002   Z   W  \0  \0  \0  \b   Z   I
0000200   M   B   A   B   W   E 377 377 377 377 377 377
```

###

##SQL##COPY#######

#######PostgreSQL#####9.0##########################

```
COPY table_name [ ( column_name [, ...] ) ]
    FROM { 'filename' | STDIN }
    [ [ WITH ]
          [ BINARY ]
          [ DELIMITER [ AS ] 'delimiter_character' ]
          [ NULL [ AS ] 'null_string' ]
          [ CSV [ HEADER ]
                [ QUOTE [ AS ] 'quote_character' ]
                [ ESCAPE [ AS ] 'escape_character' ]
                [ FORCE NOT NULL column_name [, ...] ] ] ]

COPY { table_name [ ( column_name [, ...] ) ] | ( query ) }
    TO { 'filename' | STDOUT }
    [ [ WITH ]
          [ BINARY ]
          [ DELIMITER [ AS ] 'delimiter_character' ]
          [ NULL [ AS ] 'null_string' ]
          [ CSV [ HEADER ]
                [ QUOTE [ AS ] 'quote_character' ]
                [ ESCAPE [ AS ] 'escape_character' ]
                [ FORCE QUOTE { column_name [, ...] | * } ] ] ]
```

#######BINARY#CSV#FORMAT#######################################

#######PostgreSQL#####7.3##########################

```
COPY [ BINARY ] table_name
    FROM { 'filename' | STDIN }
    [ [USING] DELIMITERS 'delimiter_character' ]
    [ WITH NULL AS 'null_string' ]

COPY [ BINARY ] table_name
    TO { 'filename' | STDOUT }
    [ [USING] DELIMITERS 'delimiter_character' ]
    [ WITH NULL AS 'null_string' ]
```

####

27.4.3

CREATE ACCESS METHOD

CREATE ACCESS METHOD — ###############

##

```
CREATE ACCESS METHOD name
    TYPE access_method_type
    HANDLER handler_function
```

##

CREATE ACCESS METHOD##################

################################

###########################

#####

name

#################

access_method_type

######################### #######TABLE#INDEX#############

handler_function

handler_function#################################### #######internal## ###1####################### ####################TABLE########## #table_am_handler########INDEX############index_am_handler######## ## #################C####API#################### TABLE##########API###### 61###INDEX#########API######62##########

#

INDEX########heptree######heptree_handler################

```
CREATE ACCESS METHOD heptree TYPE INDEX HANDLER heptree_handler;
```

###

CREATE ACCESS METHOD#PostgreSQL######

####

DROP ACCESS METHOD, CREATE OPERATOR CLASS, CREATE OPERATOR FAMILY

CREATE AGGREGATE

CREATE AGGREGATE — ############

##

```
CREATE [ OR REPLACE ] AGGREGATE name ( [ argmode ] [ argname
 ] arg_data_type [ , ... ] ) (
    SFUNC = sfunc,
    STYPE = state_data_type
    [ , SSPACE = state_data_size ]
    [ , FINALFUNC = ffunc ]
    [ , FINALFUNC_EXTRA ]
    [ , FINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE } ]
    [ , COMBINEFUNC = combinefunc ]
    [ , SERIALFUNC = serialfunc ]
    [ , DESERIALFUNC = deserialfunc ]
    [ , INITCOND = initial_condition ]
    [ , MSFUNC = msfunc ]
    [ , MINVFUNC = minvfunc ]
    [ , MSTYPE = mstate_data_type ]
    [ , MSSPACE = mstate_data_size ]
    [ , MFINALFUNC = mffunc ]
    [ , MFINALFUNC_EXTRA ]
    [ , MFINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE } ]
    [ , MINITCOND = minitial_condition ]
    [ , SORTOP = sort_operator ]
    [ , PARALLEL = { SAFE | RESTRICTED | UNSAFE } ]
)

CREATE [ OR REPLACE ] AGGREGATE name ( [ [ argmode ] [ argname
 ] arg_data_type [ , ... ] ]
                        ORDER BY [ argmode ] [ argname ] arg_data_type
 [ , ... ] ) (
    SFUNC = sfunc,
    STYPE = state_data_type
    [ , SSPACE = state_data_size ]
    [ , FINALFUNC = ffunc ]
    [ , FINALFUNC_EXTRA ]
    [ , FINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE } ]
    [ , INITCOND = initial_condition ]
    [ , PARALLEL = { SAFE | RESTRICTED | UNSAFE } ]
    [ , HYPOTHETICAL ]
)


#########

CREATE [ OR REPLACE ] AGGREGATE name (
    BASETYPE = base_type,
    SFUNC = sfunc,
    STYPE = state_data_type
    [ , SSPACE = state_data_size ]
    [ , FINALFUNC = ffunc ]
    [ , FINALFUNC_EXTRA ]
    [ , FINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE } ]
    [ , COMBINEFUNC = combinefunc ]
    [ , SERIALFUNC = serialfunc ]
    [ , DESERIALFUNC = deserialfunc ]
    [ , INITCOND = initial_condition ]
```

```
    [ , MSFUNC = msfunc ]
    [ , MINVFUNC = minvfunc ]
    [ , MSTYPE = mstate_data_type ]
    [ , MSSPACE = mstate_data_size ]
    [ , MFINALFUNC = mffunc ]
    [ , MFINALFUNC_EXTRA ]
    [ , MFINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE } ]
    [ , MINITCOND = minitial_condition ]
    [ , SORTOP = sort_operator ]
)
```

##

CREATE AGGREGATE###############CREATE OR REPLACE AGGREGATE############## ############### ##9.21########### ##CREATE AGGREGATE###########

#################(################ ###)###########

###################CREATE AGGREGATE myschema.myagg ...################# ###### ##############################

##############(###)################ ### ####### 1## ###### ######################(CREATE FUNCTION########)#

########1##2############### ######*sfunc*#######*ffunc*(####)### ################

sfunc(####, ######) ---> ######
ffunc(####) ---> #####

PostgreSQL#################*stype*################ ############################## ##################################### ################1#################### ## ##########################

################################### ###text################################### ################ ###################NULL#######

########strict############NULL#################### ########################## NULL############### ########################## ######NULL################ NULL##################################NULL########################## ###### ###max################ ####*state_data_type*####*arg_data_type*############## ########## #############NULL##########strict###################

#######strict#####################################NULL###NULL################## ## ###################NULL#####################

######strict################NULL################ #####NULL############### ### ######strict################ ############NULL########### ####avg##########0##### ##NULL######

##############NULL########(## #FINALFUNC_EXTRA###################strict##########)################ ######### ########################get_fn_expr_argtype###########

####36.12.1################################ #######MSFUNC#MINVFUNC#MSTYPE### ####################
#MSSPACE#MFINALFUNC#MFINALFUNC_EXTRA#MFINALFUNC_MODIFY#MINITCOND###

########## MINVFUNC##############M###
#########

#########ORDER BY############################## ##HYPOTHETICAL#########
############## ##
############ ######################### ################################ ########
####################
########################

36.12.4################################ ######COMBINEFUNC###############
state_data_type#internal######SERIALFUNC###DESERIALFUNC############
################ ################PARALLEL SAFE######################

MIN#MAX## ##############
######################## ##
#######

```
SELECT agg(col) FROM tab;
```

######################

```
SELECT col FROM tab ORDER BY col USING sortop LIMIT 1;
```

######NULL#############NULL#################NULL################## ######
##<####MIN##############>####MAX############## ########B-tree#################
###

################################USAGE##############EXECUTE##############

#####

name

#########################

argmode

########IN###VARIADIC### (#####OUT############) ############IN### VARIADIC##################

argname

######## ###################### ###################

arg_data_type

################# ############################*######## #### count(*)##### ####

base_type

CREATE AGGREGATE################################basetype############## #############1################## ######################basetype#"ANY" #*################# ####################

sfunc

############################## ###*N*############*sfunc*#*N*+1################ ## ####*state_data_type*## #### #*state_data_type*############## ######################################

####(#######)## ##############

state_data_type

##############

state_data_size

################(######)### ############################*state_data_type*## ################## ##

ffunc

#*state_data_type*#####1############ ########################### *ffunc*#### ################################*state_data_type*######

####(#######)##

FINALFUNC_EXTRA####################################(#####)########## NULL######## ##

FINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE }

READ_ONLY###############2####### ################### ################## ##############READ_WRITE######### ######READ_ONLY###

combinefunc

#########################*combinefunc*############ ######## ##*combinefunc*#########################2##*state_data_type*############# ###########*state_data_type*############# ################################ #########################*sfunc*#############

combinefunc##*state_data_type*####2####*state_data_type*############### ######## ######strict########### #############NULL########################### ########

state_data_type#internal#######*combinefunc*#STRICT########### ### ##*combinefunc*#NULL###

serialfunc

state_data_type#internal######*serialfunc*########################## *serialfunc*####################bytea################## #####internal## ###1####bytea############# #######*deserialfunc*######

deserialfunc

##########################*state_data_type*###### #####bytea####internal## 2#######internal################# ####2###internal#################### ###########

initial_condition

########### ####*state_data_type*######################### ################ ####NULL#######

msfunc

############*mstate_data_type*# ##*state_data_type*################################

minvfunc

########################## #####*msfunc*#################################### ############### ################strict##################

mstate_data_type

##########################

mstate_data_size

#############################(#####)### *state_data_size*###########

mffunc

###########
#*mstate_data_type*####MFINALFUNC_EXTRA#####################*ffunc*### ######## *mffunc*###*mstate_data_type*#################################### ###########

MFINALFUNC_MODIFY = { READ_ONLY | SHAREABLE | READ_WRITE }

########FINALFUNC_MODIFY#########################

minitial_condition

######################### ####*initial_condition*###########

sort_operator

MIN###MAX######################### ######################### ############### ######################################

PARALLEL = { SAFE | RESTRICTED | UNSAFE }

PARALLEL SAFE#PARALLEL RESTRICTED#PARALLEL UNSAFE####CREATE FUNCTION########### ########PARALLEL UNSAFE################PARALLEL RESTRICTED######################## ################################### ################

HYPOTHETICAL

####################
(WITHIN GROUP#)############### HYPOTHETICAL###########################
#######################

CREATE AGGREGATE##

##

#########################SFUNC = public.sum################### ############# ######### ################################

##PostgreSQL############################## ################################### ###################### ##

######
#FINALFUNC_MODIFY################## READ_WRITE########################### ######## ####window################## ################################### SHAREABLE## ####window# ################################ ## ################### ##########READ_WRITE#########################

##UNBOUNDED PRECEDING############## ###################### ## ################# ################################### ###################### ################################### ##############################1########### ### (##)

#########################NULL############ ######NULL######################## ## ######################## ##

###PostgreSQL############## # #########################PostgreSQL############################ ############## ############### ###window###################################### ############# ##

###########VARIADIC#################(WITHIN GROUP)######################## ##########VARIADIC####2############ 1############VARIADIC "any"####### ##variadic############### 2##############VARIADIC "any"###########1###### ######VARIADIC "any"################# (#######################2########1# #VARIADIC "any"########### #####pg_proc#2####VARIADIC################### ##) #########VARIADIC "any"## ###############

###

#################################### #######DISTINCT####ORDER BY########### ################ ################################

# #

36.12#########

###

CREATE AGGREGATE#PostgreSQL####### ##SQL##########################

####

ALTER AGGREGATE, DROP AGGREGATE

CREATE CAST

CREATE CAST — ###########

##

```
CREATE CAST (source_type AS target_type)
    WITH FUNCTION function_name [ (argument_type [, ...]) ]
    [ AS ASSIGNMENT | AS IMPLICIT ]

CREATE CAST (source_type AS target_type)
    WITHOUT FUNCTION
    [ AS ASSIGNMENT | AS IMPLICIT ]

CREATE CAST (source_type AS target_type)
    WITH INOUT
    [ AS ASSIGNMENT | AS IMPLICIT ]
```

##

CREATE CAST#################### ######2######################## ##########

```
SELECT CAST(42 AS float8);
```

#########################float8(int4)###########42#float8####### ########## ####################

2######################### ## ####### ######################### ########text#varchar################### ########## ############## ##########xml##text### ### #2#####################################

WITH INOUT#######*I/O*######################## I/O############################ ######################### ## I/O#### ######################################

################################### ##############CAST(*x* AS *typename*)#### ##*x*::*typename*####

#####AS ASSIGNMENT## ## ##foo.f1#text##########

```
INSERT INTO foo (f1) VALUES (42);
```

integer##text##########AS ASSIGNMENT#################SQL######## ####AS ASSIGNMENT#################### ##########################

#####AS IMPLICIT## #### ########################## ####################

```
SELECT 2 + 4.0;
```

############integer#numeric########## ##########integer + numeric####### ######numeric + numeric########### ######integer##numeric################ #####AS IMPLICIT################################## ######################### ####################

```
SELECT CAST ( 2 AS numeric ) + 4.0;
```

##########numeric##integer############## ########AS IMPLICIT############ ##########################numeric###integer######integer + integer####### ################ ## 2########1### ##################numeric#integer######numeric######################### ### ##################

################### ###############PostgreSQL########################## ####### ##################################### ######2################################## ### ####int2###int4################### ######float8###int4############################ text###int4############### ############################

##
###CREATE TYPE#########

##USAGE############# ########### ################################### ## ####################

#####

source_type

##################

target_type

##################

function_name[(*argument_type* [, ...])]

##################### ################### ################################ ## ################################### ############### ################### ################

WITHOUT FUNCTION

##########################

WITH INOUT

###I/O################

AS ASSIGNMENT

##############################

AS IMPLICIT

################################

###########1#3############## 1## ########### 2#############integer################################## ######

####-1###### 3#############boolean############################true########
#false###### ##########SQL## ##########
############################ ################################

##

#################################### ####2####################################
####################### ### ############
#####2###########################

###1################
###1##2################

################################### ##
##

##

################DROP CAST##########

###

#############text#varchar#char(*n*)##
########## PostgreSQL##########I/O############# ################################
######################### ## ########
######################
############# ##################################### ##########################
###########SQL############

######
#*typename*(*x*)########################### ############################## ####
################### ## PostgreSQL###
##
###

##
####################### ####################################2####### ### ###*name*(*x*)##############*name*###########pg_cast#*x*################### ################################### ################################### ######################### ####pg_cast################################### ################### ###################################

##
############## ###########I/O#######################################CAST## ###::################### ##########I/O################################### #########################

# #

##int4(bigint)#####bigint###int4###############

```
CREATE CAST (bigint AS int4) WITH FUNCTION int4(bigint) AS ASSIGNMENT;
```

##########################

###

SQL##AS IMPLICIT##PostgreSQL#### ## ##########CREATE CAST###SQL########

####

CREATE FUNCTION, CREATE TYPE, DROP CAST

CREATE COLLATION

CREATE COLLATION — ###########

##

```
CREATE COLLATION [ IF NOT EXISTS ] name (
    [ LOCALE = locale, ]
    [ LC_COLLATE = lc_collate, ]
    [ LC_CTYPE = lc_ctype, ]
    [ PROVIDER = provider, ]
    [ DETERMINISTIC = boolean, ]
    [ RULES = rules, ]
    [ VERSION = version ]
)
CREATE COLLATION [ IF NOT EXISTS ] name FROM existing_collation
```

##

CREATE COLLATION##

#############################CREATE########

#####

IF NOT EXISTS

################################# ################### ########################### ###########################

name

########## ######################## ################################### ##### ######################## ## ######## #####################

locale

############### ####23.2.2.3.1#23.2.2.3.2########

provider#libc#########LC_COLLATE###LC_CTYPE########### *locale*######### ####################

######*provider*#builtin####*locale*#####C###C.UTF-8#################

lc_collate

provider#libc###LC_COLLATE#######################################

lc_ctype

provider#libc###LC_CTYPE#######################################

provider

#####[81.944444]#################################### ######icu(####ICU###### ########)#libc### libc######### ####23.1.4########## ######################

################# #######builtin#icu#####ICU#####################libc### libc######## ###23.1.4#########

DETERMINISTIC

######################### ######### ###################################### ######################### PostgreSQL################## ################### ################################## #########LOCALE#################### ##############

###########ICU##################

rules

################################## ###ICU############## ###23.2.3.4#########

version

########################## ## ##### ########pg_upgrade##

#########################ALTER COLLATION#########

existing_collation

################# ######################################

##

CREATE COLLATION#SHARE ROW EXCLUSIVE############pg_collation######### #########CREATE COLLATION####1##########

##################DROP COLLATION#########

####################23.2.2.3#########

libc## #########CREATE DATABASE#########

# #

################fr_FR.utf8########### ################UTF8#########

```
CREATE COLLATION french (locale = 'fr_FR.utf8');
```

ICU##################################

```
CREATE COLLATION german_phonebook (provider = icu, locale = 'de-u-co-
phonebk');
```

ICU############ICU##

```
CREATE COLLATION custom (provider = icu, locale = 'und', rules = '&V << w
 <<< W');
```

################23.2.3.4#########

###################

```
CREATE COLLATION german FROM "de_DE";
```

##

###

##SQL##CREATE COLLATION############################ ################## PostgreSQL######

####

ALTER COLLATION, DROP COLLATION

CREATE CONVERSION

CREATE CONVERSION — ##############

##

```
CREATE [ DEFAULT ] CONVERSION name
    FOR source_encoding TO dest_encoding FROM function_name
```

##

CREATE CONVERSION#######2##########################

DEFAULT## ######################### A#######B###########B#######A####2################

################EXECUTE#################CREATE################

#####

DEFAULT

DEFAULT### 1############## #################1#####1######

name

######## #################### ################################ ############ ###########

source_encoding

############

dest_encoding

############

function_name

################### #################### ############################

#########################

```
conv_proc(

    integer,  -- ########ID
    integer,  -- ########ID
    cstring,  -- #######NULL####C######
    internal, -- #######NULL####C######
    integer,  -- #######
    boolean   -- ####################
) RETURNS integer;
```

##################### ##

##

SQL_ASCII###SQL_ASCII#######

###############DROP CONVERSION######

##################################

#

myfunc#######UTF8##LATIN1################

```
CREATE CONVERSION myconv FOR 'UTF8' TO 'LATIN1' FROM myfunc;
```

###

CREATE CONVERSION##PostgreSQL###### ##SQL##CREATE CONVERSION###### ##CREATE TRANSLATION###################

####

ALTER CONVERSION, CREATE FUNCTION, DROP CONVERSION

CREATE DATABASE

CREATE DATABASE — ##############

##

```
CREATE DATABASE name
    [ WITH ] [ OWNER [=] user_name ]
           [ TEMPLATE [=] template ]
           [ ENCODING [=] encoding ]
           [ STRATEGY [=] strategy ]
           [ LOCALE [=] locale ]
           [ LC_COLLATE [=] lc_collate ]
           [ LC_CTYPE [=] lc_ctype ]
           [ BUILTIN_LOCALE [=] builtin_locale ]
           [ ICU_LOCALE [=] icu_locale ]
           [ ICU_RULES [=] icu_rules ]
           [ LOCALE_PROVIDER [=] locale_provider ]
           [ COLLATION_VERSION = collation_version ]
           [ TABLESPACE [=] tablespace_name ]
           [ ALLOW_CONNECTIONS [=] allowconn ]
           [ CONNECTION LIMIT [=] connlimit ]
           [ IS_TEMPLATE [=] istemplate ]
           [ OID [=] oid ]
```

##

CREATE DATABASE####PostgreSQL############

##########################CREATEDB######################### CREATE ROLE###########

##############################template1################## ##############
##TEMPLATE *name*####### ###TEMPLATE template0#################PostgreSQL#####
##################(##)###################
####template1############################

#####

name

###############

user_name

######################### ################################DEFAULT####### ##
#####################################SET ROLE############

template

############################ ###########template1#######DEFAULT#######

encoding

########################### #########'SQL_ASCII'############DEFAULT######
DEFAULT## PostgreSQL#####
################23.3.1####### ################

strategy

###################### WAL_LOG###
######### ## ##FILE_COPY##########
##################
############################ ##
################################
#########

locale

################################## ##########ORDER BY####################
############################
######## ########################LC_COLLATE###LC_CTYPE####### ###########
############### ####23.2.2.3.1###23.2.2.3.2##########

#####[66.956522]#*lc_collate*#*lc_ctype*####*icu_locale*################## ###
###*lc_collate*#*lc_ctype*#*builtin_locale*#*icu_locale*##################

######*locale_provider*#builtin####*locale*###*builtin_locale*#####C##
#C.UTF-8############

###
##########lc_messages#lc_monetary#lc_numeric#lc_time#################### ########## ########################ALTER DATABASE ... SET######

lc_collate

#########################LC_COLLATE####### ################*locale*########
######################## #####################

locale_provider#libc################################*locale*###########

lc_ctype

#########################LC_CTYPE####### ################*locale*##########
######################## #####################

locale_provider#libc##################################*locale*##########

builtin_locale

#####[67.298578]##########################ICU#####23.2.2.3.2 ###########
locale ############## ##########ICU########## ################*locale*####
############################ ##
locale_provider#builtin############ ######*locale_provider*###### #####
#########################

######builtin################C#C.UTF-8###

icu_locale

##########################ICU#####23.2.2.3.2 ########## *locale* ##############
##########ICU########## ################*locale*##########################
####

icu_rules

###ICU##############
#23.2.3.4##########

locale_provider

#####[81.127451]#################################### #############ICU#######
##############icu###libc### libc######## ##############*template*######
###23.1.4########## ##
#builtin#icu#####ICU#####################libc### #############
#*template*######## ###23.1.4##########

collation_version

############################## ##
######################### #########pg_upgrade############################
##########

##################################ALTER DATABASE##########

tablespace_name

################################## ############################DEFAULT#######
###CREATE TABLESPACE###
#######

allowconn

false######################## ######true##########GRANT/REVOKE CONNECT#######
#################

connlimit

###################### -1##################

istemplate

true####CREATEDB############################## false##########################
###############

oid

########################### #################PostgreSQL####OID########### ####
######pg_upgrade###################pg_upgrade###16384##############

##

##

CREATE DATABASE#########################

#########could not initialize database directory###
######################

#############DROP DATABASE##########

createdb#####################################

#################(ALTER DATABASE###########)#############(GRANT###########)###
####################

template1###COPY
DATABASE######################## ##
CREATE DATABASE##################### ##################CREATE DATABASE###
########## ###22.3##########

```
#####################################LC_COLLATE###LC_CTYPE###############
#####C#####POSIX##############################################1########   ####
Windows##UTF-8#################### CREATE   DATABASE##################
#SQL_ASCII#################################################################
#######################

#############################################   ###template0################
######   ##################################LC_COLLATE###LC_CTYPE##########
###########################   ########################################   ##
#template0####################################

########################################(CREATE COLLATION########)# #######
######################

CONNECTION  LIMIT############# #################1##########2##############
###################### ##########################################
```

# #

```
###############


CREATE DATABASE lusiadas;

###salesapp#####salesspace####################sales#######


CREATE DATABASE sales OWNER salesapp TABLESPACE salesspace;

############music######


CREATE DATABASE music
    LOCALE 'sv_SE.utf8'
    TEMPLATE template0;

################template1###########TEMPLATE template0########## #######
##########################

############################music2#######


CREATE DATABASE music2
    LOCALE 'sv_SE.iso885915'
    ENCODING LATIN9
    TEMPLATE template0;

##################################################

######OS#################################################
```

###

```
##SQL##CREATE DATABASE######## ##############################
```

####

ALTER DATABASE, DROP DATABASE

CREATE DOMAIN

CREATE DOMAIN — ############

##

```
CREATE DOMAIN name [ AS ] data_type
    [ COLLATE collation ]
    [ DEFAULT expression ]
    [ domain_constraint [ ... ] ]

where domain_constraint is:

[ CONSTRAINT constraint_name ]
{ NOT NULL | NULL | CHECK (expression) }
```

##

CREATE DOMAIN############## ### ##### ###################

###################CREATE DOMAIN myschema.mydomain ...################## ###### ##################################### ################################## #############

################1#################### #####E-mail############################### ###############CHECK################## ##################################### ###

######################USAGE############

#####

name

#########################

data_type

################ ################

collation

########(####)### ## COLLATE########## ##########################

DEFAULT *expression*

DEFAULT######################### ## ################################### #######################NULL#######

##

CONSTRAINT *constraint_name*

############## #######################

NOT NULL

###########NULL#############################

NULL

#########NULL########## ############

#########SQL########################### ##############################

CHECK (*expression*)

CHECK################################## #####Boolean#################### # ############VALUE################ TRUE###UNKNOWN############### ##### FALSE##################################

######CHECK##############VALUE#####################

########CHECK############################## #PostgreSQL#9.5############## #CHECK############################

##

#########NOT NULL######################## ##############NOT NULL####### #NULL################## ##############################NULL################### ## ############

```
INSERT INTO tab (domcol) VALUES ((SELECT domcol FROM tab WHERE false));
```

########SELECT#################NULL######## ############################## ##

SQL##NULL### ########### ##NOT NULL####################NULL###########################NOT NULL##### #####################

PostgreSQL#CHECK#####immutable################################## ############# #############CHECK############################## (####5.5.1###########CHECK#### ###########)

##############CHECK#################################### PostgreSQL############ ###############CHECK########################### ############################## ########## ##################(ALTER DOMAIN####)############################## ############################

####CHECK##########################

# #

##########us_postal_code######################## #######US################### ##############

```
CREATE DOMAIN us_postal_code AS TEXT
CHECK(
   VALUE ~ '^\d{5}$'
OR VALUE ~ '^\d{5}-\d{4}$'
);

CREATE TABLE us_snail_addy (
```

```
  address_id SERIAL PRIMARY KEY,
  street1 TEXT NOT NULL,
  street2 TEXT,
  street3 TEXT,
  city TEXT NOT NULL,
  postal us_postal_code NOT NULL
);
```

###

CREATE DOMAIN######SQL########

#############NOT NULL######PostgreSQL##########################CHECK (VALUE IS NOT NULL#### ################################## NULL######PostgreSQL##### ##########################

####

ALTER DOMAIN, DROP DOMAIN

CREATE EVENT TRIGGER

CREATE EVENT TRIGGER — ##############

##

```
CREATE EVENT TRIGGER name
    ON event
    [ WHEN filter_variable IN (filter_value [, ... ]) [ AND ... ] ]
    EXECUTE { FUNCTION | PROCEDURE } function_name()
```

##

CREATE EVENT TRIGGER################# #####################WHEN########### ################# ####################38########## ##########################

#####

name

##############################

event

########################## ############38.1#########

filter_variable

######################### ## # #*filter_variable*############TAG#####

filter_value

#####################*filter_variable*######### TAG################'DROP FUNCTION'###########

function_name

######event_trigger#####################

CREATE EVENT TRIGGER##########FUNCTION#PROCEDURE################### ############################### ########PROCEDURE#####################

##

#############################

#####[52.572707]###########postgres#################### ########################### ### ############# ########### ##############postgres####event_triggers#false################## ################ ####################event_triggers#false######################### ############ ############

#

####DDL############

```
CREATE OR REPLACE FUNCTION abort_any_command()
  RETURNS event_trigger
 LANGUAGE plpgsql
  AS $$
BEGIN
  RAISE EXCEPTION 'command % is disabled', tg_tag;
END;
$$;

CREATE EVENT TRIGGER abort_ddl ON ddl_command_start
   EXECUTE FUNCTION abort_any_command();
```

###

##SQL##CREATE EVENT TRIGGER########

####

ALTER EVENT TRIGGER, DROP EVENT TRIGGER, CREATE FUNCTION

CREATE EXTENSION

CREATE EXTENSION — ##########

##

```
CREATE EXTENSION [ IF NOT EXISTS ] extension_name
    [ WITH ] [ SCHEMA schema_name ]
             [ VERSION version ]
             [ CASCADE ]
```

##

CREATE EXTENSION######################### ##############################

#################################### ####################################SQL ################## CREATE EXTENSION##########################DROP EXTENSION###################

CREATE EXTENSION## #######################

#############trusted#######################CREATE######################### ### ##(################################ #)#################### ###

#####

IF NOT EXISTS

########################## ############# ################################## ###################

extension_name

################ PostgreSQL#####SHAREDIR/extension/*extension_name*.control##################

schema_name

##

###########schema####################SCHEMA################ SCHEMA# ###########schema########################## ####CASCADE########## ##*schema_name*################ ##############################schema### #############*schema_name*#######

#################################### ################################### ### ##############################

version

################### ############################### ######################### #########

CASCADE

###########################
SCHEMA### ##########################
######### ########################

##

#################CREATE EXTENSION##
PostgreSQL######################################

##########pg_available_extensions###pg_available_extension_versions#
#############

##
############################search_path############### ### ############################## ##############CREATE## # ###################   PostgreSQL## ### ###

############36.17#########

#

#########addons###################hstore############

```
CREATE EXTENSION hstore SCHEMA addons;
```

##################

```
SET search_path = addons;
CREATE EXTENSION hstore;
```

###

CREATE EXTENSION#PostgreSQL######

####

ALTER EXTENSION, DROP EXTENSION

CREATE FOREIGN DATA WRAPPER

CREATE FOREIGN DATA WRAPPER — ################

##

```
CREATE FOREIGN DATA WRAPPER name
    [ HANDLER handler_function | NO HANDLER ]
    [ VALIDATOR validator_function | NO VALIDATOR ]
    [ OPTIONS ( option 'value' [, ... ] ) ]
```

##

CREATE FOREIGN DATA WRAPPER################### ############################

################################

################################

#####

name

###################

HANDLER *handler_function*

handler_function## ##########
####fdw_handler################

###

VALIDATOR *validator_function*

validator_function###
##########NO VALIDATOR#
######################### ##
#####2################ 1##text[]############################## ##1#
#oid###################OID### ############ ###ereport()################
##########

OPTIONS (*option* 'value' [, ...])

############################ ##
############ ###################

##

PostgreSQL######################### ##
########################

#

############dummy#######

```
CREATE FOREIGN DATA WRAPPER dummy;
```

file_fdw_handler#################file#######

```
CREATE FOREIGN DATA WRAPPER file HANDLER file_fdw_handler;
```

######################mywrapper#######

```
CREATE FOREIGN DATA WRAPPER mywrapper
    OPTIONS (debug 'true');
```

###

CREATE FOREIGN DATA WRAPPER#ISO/IEC 9075-9 (SQL/MED)###### ####HANDLER# #VALIDATOR########PostgreSQL#####LIBRARY##LANGUAGE############

####SQL/MED##########################

####

ALTER FOREIGN DATA WRAPPER, DROP FOREIGN DATA WRAPPER, CREATE SERVER, CREATE USER MAPPING, CREATE FOREIGN TABLE

CREATE FOREIGN TABLE

CREATE FOREIGN TABLE — ##############

##

```
CREATE FOREIGN TABLE [ IF NOT EXISTS ] table_name ( [
  { column_name data_type [ OPTIONS ( option 'value' [, ... ] ) ]
 [ COLLATE collation ] [ column_constraint [ ... ] ]
    | table_constraint }
    [, ... ]
] )
[ INHERITS ( parent_table [, ... ] ) ]
  SERVER server_name
[ OPTIONS ( option 'value' [, ... ] ) ]

CREATE FOREIGN TABLE [ IF NOT EXISTS ] table_name
  PARTITION OF parent_table [ (
  { column_name [ WITH OPTIONS ] [ column_constraint [ ... ] ]
    | table_constraint }
    [, ... ]
) ]
{ FOR VALUES partition_bound_spec | DEFAULT }
  SERVER server_name
[ OPTIONS ( option 'value' [, ... ] ) ]


###column_constraint#########

[ CONSTRAINT constraint_name ]
{ NOT NULL |
  NULL |
  CHECK ( expression ) [ NO INHERIT ] |
  DEFAULT default_expr |
  GENERATED ALWAYS AS ( generation_expr ) STORED }


##table_constraint#########

[ CONSTRAINT constraint_name ]
CHECK ( expression ) [ NO INHERIT ]


##partition_bound_spec#########

IN ( partition_bound_expr [, ...] ) |
FROM ( { partition_bound_expr | MINVALUE | MAXVALUE } [, ...] )
  TO ( { partition_bound_expr | MINVALUE | MAXVALUE } [, ...] ) |
WITH ( MODULUS numeric_literal, REMAINDER numeric_literal )
```

##

CREATE FOREIGN TABLE##

################CREATE FOREIGN TABLE myschema.mytable ...############
############ ######################## ##
##

CREATE FOREIGN TABLE#################################### ##################
############################

PARTITION OF##############parent_table#######################

#######################USAGE######################USAGE##############

#####

IF NOT EXISTS

############################## ############# ##############################
###########

table_name

#######################

column_name

###################

data_type

######## #################### PostgreSQL#######################8###########

COLLATE *collation*

COLLATE################################## ##############################
##

INHERITS (*parent_table* [, ...])

######INHERITS## ###############
############## ####CREATE TABLE##############

PARTITION OF *parent_table* { FOR VALUES *partition_bound_spec* | DEFAULT }

#######
##CREATE TABLE############### ############UNIQUE#######################
#################################### (ALTER TABLE ATTACH PARTITION####
######)

CONSTRAINT *constraint_name*

###################### ##############################col must be positive## ###################
############## #########################

NOT NULL

####NULL##############

NULL

####NULL############# ###########

#########SQL######################## ############################

CHECK (*expression*) [NO INHERIT]

CHECK## #######################TRUE
###UNKNOWN########FALSE######## ######################################
######################

#######CHECK#################################### #####tableoid#######
############################

NO INHERIT########################

DEFAULT *default_expr*

DEFAULT### ############################## ##############################
################### ##############################

################################## #####################NULL###

GENERATED ALWAYS AS (*generation_expr*) STORED

################## #####################################

#####STORED############################ (################################
############)

################################## #############immutable########### #######
##########

server_name

######################### #############CREATE SERVER##########

OPTIONS (*option* '*value*' [, ...])

############################## ##
#################### ##
###

##

###########CHECK##NOT NULL#####PostgreSQL#################################
############# ########################## ##################################
#################
####################################

###
#################### ##

PostgreSQL## ##############
############
#######################

##
##

##UPDATE#########################
##########################
########################### ############UPDATE################################
################

################### ############PostgreSQL#####################################
############
#######################

# #

```
###film_server#################films#######


CREATE FOREIGN TABLE films (
    code        char(5) NOT NULL,
    title       varchar(40) NOT NULL,
    did         integer NOT NULL,
    date_prod   date,
    kind        varchar(10),
    len         interval hour to minute
)
SERVER film_server;

############measurement##############server_07###############
#measurement_y2016m07#######


CREATE FOREIGN TABLE measurement_y2016m07
    PARTITION OF measurement FOR VALUES FROM ('2016-07-01') TO
 ('2016-08-01')
    SERVER server_07;
```

###

CREATE FOREIGN TABLE#######SQL####### ###CREATE TABLE########NULL######## ########### ################PostgreSQL###### PostgreSQL#########################

####

ALTER FOREIGN TABLE, DROP FOREIGN TABLE, CREATE TABLE, CREATE SERVER, IMPORT FOREIGN SCHEMA

CREATE FUNCTION

CREATE FUNCTION — ##########

##

```
CREATE [ OR REPLACE ] FUNCTION
    name ( [ [ argmode ] [ argname ] argtype [ { DEFAULT |
 = } default_expr ] [, ...] ] )
    [ RETURNS rettype
      | RETURNS TABLE ( column_name column_type [, ...] ) ]
  { LANGUAGE lang_name
    | TRANSFORM { FOR TYPE type_name } [, ... ]
    | WINDOW
    | { IMMUTABLE | STABLE | VOLATILE }
    | [ NOT ] LEAKPROOF
    | { CALLED ON NULL INPUT | RETURNS NULL ON NULL INPUT | STRICT }
    | { [ EXTERNAL ] SECURITY INVOKER | [ EXTERNAL ] SECURITY DEFINER }
    | PARALLEL { UNSAFE | RESTRICTED | SAFE }
    | COST execution_cost
    | ROWS result_rows
    | SUPPORT support_function
    | SET configuration_parameter { TO value | = value | FROM CURRENT }
    | AS 'definition'
    | AS 'obj_file', 'link_symbol'
    | sql_body
  } ...
```

##

CREATE FUNCTION############ CREATE OR REPLACE FUNCTION################## ############### ###################USAGE########

################################## ############################ ############## ### ################################## ######## ##################

################CREATE OR REPLACE FUNCTION######### ###################### ###### ######################### ###CREATE OR REPLACE FUNCTION############ ############### ################################## #####OUT##################### ####OUT#######################

CREATE OR REPLACE FUNCTION################################## ############ ################################## ############################### ############ ###################

################################## ### ##############################CREATE OR REPLACE FUNCTION######### ###ALTER FUNCTION################################

##########################

############################USAGE###############

##################36.3##########

#####

name

#####################

argmode

#######IN#OUT#INOUT#VARIADIC####### #########IN### OUT#####VARIADIC## ########### ###RETURNS TABLE#####OUT#INOUT#################

argname

####### #SQL###PL/pgSQL################################ ############## ################################ ##################################### #4.3### ## ########################## #########

argtype

######################## ##

################cstring#################### ########################## #SQL###################

###########*table_name.column_name*%TYPE###### ######################### ########################

default_expr

################################## ########################### ###### #INOUT######################### ### ########

rettype

######################## ## ########### #######cstring#################### ############################void##### #####

OUT####INOUT############RETURNS############# ######################### ########################RECORD################################

SETOF##########1########################

#####*table_name.column_name*%TYPE##############

column_name

RETURNS TABLE############## ###########OUT#################RETURNS TABLE#RETURNS SETOF#############

column_type

RETURNS TABLE################

lang_name

################ #########sql#c#internal##############(##plpgsql)###### ##### *sql_body*###############sql### ####################################

TRANSFORM { FOR TYPE *type_name* } [, ...] }

######################### ###SQL######################CREATE TRANSFORM# #### ## ######### ##

WINDOW

WINDOW################################## #####C######################### #### ############WINDOW################

IMMUTABLE
STABLE
VOLATILE

##################################### ####1################ ############# #VOLATILE########

IMMUTABLE## ## ## ########################### ########################

STABLE############################1#################################### ##SQL############################## ## ########## (#########################AFTER###########) # ##current_timestamp######1######################STABLE###############

VOLATILE##1## ########### #volatile#################### ####random()#currval()#timeofday()########## ## #######setval()##VOLATILE### ########################

###36.7##########

LEAKPROOF

LEAKPROOF#################### ######################### ################### ######################################(leakproof)######## ## #security_barrier### ############## ## ############################### ### ################################## ### ##################################### CREATE VIEW###39.5########## ######## ########################

CALLED ON NULL INPUT
RETURNS NULL ON NULL INPUT
STRICT

CALLED ON NULL INPUT################NULL######################## ###### ######NULL############################

RETURNS NULL ON NULL INPUT####STRICT############1###NULL########NULL ###### ###############NULL################### #####NULL#################

[EXTERNAL] SECURITY INVOKER
[EXTERNAL] SECURITY DEFINER

SECURITY INVOKER################################## ########### SECURITY DEFINER################################## SECURITY DEFINER############## ##############

EXTERNAL#######SQL################## ####SQL#############################
#####################

PARALLEL

#####[74.257426]# PARALLEL UNSAFE################################SQL#####
########################## ########## PARALLEL RESTRICTED##############
################################### PARALLEL SAFE#######################
############## ######PARALLEL UNSAFE######################## ########SQL
##################### ########## PARALLEL RESTRICTED##################
############################## PARALLEL SAFE##############################
###############

#####[80.555556]###
###################setval######parallel unsafe###################### #######
###parallel
restricted##########################setseed################################
####################### #####restricted####unsafe####safe###############unsafe
####restricted### C###
######C#######################
################################## ################UNSAFE#############
##
###################currval##
##
##############################setseed#################################
################ ##
C##################
################################ #######C################################
############# ###########UNSAFE############### ###########

COST *execution_cost*

#####################cpu_operator_cost### ##############1############## #######
#######C###########1###########100###### ##################################
#####

ROWS *result_rows*

############################## ############################## #########1000#
###

SUPPORT *support_function*

#######################(#########)# ###36.11######### ######################
########

configuration_parameter
value

SET## SET FROM
CURRENT##CREATE FUNCTION##################################

###SET###################SET LOCAL########################### ##########
##################### ########LOCAL####SET#####SET######### ##########SET
LOCAL################ ##
###

#################SET####19###########

definition

############### ################### #######################SQL##############
################

#####################################4.1.2.4###
###

obj_file, link_symbol

#####AS############C########C##############SQL##################
obj_file##############C##################LOAD################## ##
#*link_symbol*################C################ ####################SQL
################### #########C##################################C#######
###C###############(####C#######################)#

#################CREATE FUNCTION#######################################
##

sql_body

LANGUAGE SQL####### #############

RETURN *expression*

##########

BEGIN ATOMIC
  statement;
  statement;
  ...
  statement;
END

#########################(###*definition*########)############# ####
#LANGUAGE SQL################################ ########################
############## ### ################
#########################DROP ... CASCADE##############################
############# ###########SQL###SQL##############

########

PostgreSQL################# ### ##########
###10.3##########

####################2################# OUT############ ####################
##

```
CREATE FUNCTION foo(int) ...
CREATE FUNCTION foo(int, out text) ...
```

#############
##

```
CREATE FUNCTION foo(int) ...
CREATE FUNCTION foo(int, int default 42) ...
```

foo(10)##

##

###################SQL########### ##################numeric#########
##CREATE FUNCTION######### ####CREATE FUNCTION foo
(varchar(10)) ...#CREATE FUNCTION foo (varchar) ...############

######CREATE OR REPLACE FUNCTION################################ ########## ########################### ## ########## ## ################## ################################

###VARIADIC#####STRICT##################variadic######NULL########## ### NULL######################

#

SQL######2########

```
CREATE FUNCTION add(integer, integer) RETURNS integer
    AS 'select $1 + $2;'
    LANGUAGE SQL
    IMMUTABLE
    RETURNS NULL ON NULL INPUT;
```

####################SQL###############

```
CREATE FUNCTION add(a integer, b integer) RETURNS integer
    LANGUAGE SQL
    IMMUTABLE
    RETURNS NULL ON NULL INPUT
    RETURN a + b;
```

PL/pgSQL##############1######

```
CREATE OR REPLACE FUNCTION increment(i integer) RETURNS integer AS $$
        BEGIN
                RETURN i + 1;
        END;
$$ LANGUAGE plpgsql;
```

########################

```
CREATE FUNCTION dup(in int, out f1 int, out f2 text)
    AS $$ SELECT $1, CAST($1 AS text) || ' is text' $$
    LANGUAGE SQL;

SELECT * FROM dup(42);
```

##

```
CREATE TYPE dup_result AS (f1 int, f2 text);

CREATE FUNCTION dup(int) RETURNS dup_result
    AS $$ SELECT $1, CAST($1 AS text) || ' is text' $$
    LANGUAGE SQL;

SELECT * FROM dup(42);
```

############TABLE############

```
CREATE FUNCTION dup(int) RETURNS TABLE(f1 int, f2 text)
    AS $$ SELECT $1, CAST($1 AS text) || ' is text' $$
    LANGUAGE SQL;
```

```
SELECT * FROM dup(42);
```

###########1#########################TABLE#############

SECURITY DEFINER########

SECURITY DEFINER###
####search_path##################################### ##########################
#####
################################ ################################## #####
##pg_temp#search_path################ ##################

```
CREATE FUNCTION check_password(uname TEXT, pass TEXT)
RETURNS BOOLEAN AS $$
DECLARE passed BOOLEAN;
BEGIN
        SELECT  (pwd = $2) INTO passed
        FROM    pwds
        WHERE   username = $1;

        RETURN passed;
END;
$$  LANGUAGE plpgsql
    SECURITY DEFINER

    -- ##############pg_temp#####search_path##########
    SET search_path = admin, pg_temp;
```

############admin.pwds########### ###SET##########SET##admin###########
##pwds#####################################

SECURITY DEFINER#####################################
##createrole_self_grant#SET#####################

##################################PUBLIC############### ####5.8##########
SECURITY DEFINER############################ #############PUBLIC#########
#################### ###
######## #########

```
BEGIN;
CREATE FUNCTION check_password(uname TEXT, pass TEXT) ... SECURITY
 DEFINER;
REVOKE ALL ON FUNCTION check_password(uname TEXT, pass TEXT) FROM PUBLIC;
GRANT EXECUTE ON FUNCTION check_password(uname TEXT, pass TEXT) TO admins;
COMMIT;
```

###

CREATE FUNCTION#######SQL######### PostgreSQL##############################
#####SQL##PostgreSQL############################

###############

- OR REPLACE#PostgreSQL######
- ####################*argmode#argname*################## ####1############
  ###
- ##################SQL##DEFAULT################ =######T-SQL###Firebird#####
  #####

- `SETOF`####PostgreSQL######
- `SQL`#################
- `CALLED ON NULL INPUT`#`RETURNS NULL ON NULL INPUT`##################
- `LANGUAGE SQL`###########SQL#*`sql_body`*############

###`LANGUAGE SQL`## ######################## ##############PostgreSQL#############

####

ALTER FUNCTION, DROP FUNCTION, GRANT, LOAD, REVOKE

CREATE GROUP

CREATE GROUP — ################

##

```
CREATE GROUP name [ [ WITH ] option [ ... ] ]
```

option###########

```
      SUPERUSER | NOSUPERUSER
    | CREATEDB | NOCREATEDB
    | CREATEROLE | NOCREATEROLE
    | INHERIT | NOINHERIT
    | LOGIN | NOLOGIN
    | REPLICATION | NOREPLICATION
    | BYPASSRLS | NOBYPASSRLS
    | CONNECTION LIMIT connlimit
    | [ ENCRYPTED ] PASSWORD 'password' | PASSWORD NULL
    | VALID UNTIL 'timestamp'
    | IN ROLE role_name [, ...]
    | IN GROUP role_name [, ...]
    | ROLE role_name [, ...]
    | ADMIN role_name [, ...]
    | USER role_name [, ...]
    | SYSID uid
```

##

CREATE GROUP#CREATE ROLE##########

###

##SQL##CREATE GROUP#######

####

CREATE ROLE

CREATE INDEX

CREATE INDEX — ##############

##

```
CREATE [ UNIQUE ] INDEX [ CONCURRENTLY ] [ [ IF NOT EXISTS ] name ] ON
 [ ONLY ] table_name [ USING method ]
    ( { column_name | ( expression ) } [ COLLATE collation ] [ opclass
 [ ( opclass_parameter = value [, ... ] ) ] ] [ ASC | DESC ] [ NULLS
 { FIRST | LAST } ] [, ...] )
    [ INCLUDE ( column_name [, ...] ) ]
    [ NULLS [ NOT ] DISTINCT ]
    [ WITH ( storage_parameter [= value] [, ... ] ) ]
    [ TABLESPACE tablespace_name ]
    [ WHERE predicate ]
```

##

CREATE INDEX##################(###)############### ########################### ########### ############################## ################################### ###

###################################### ####################################### #############

######################1#################### ################################### ######################### ####upper(col)###################WHERE upper(col) = 'JIM'#########################

PostgreSQL#B-tree######GiST#SP-GiST#GIN#BRIN################### ############## ############################

WHERE######################### ## ############# ## #################################### ####################UNIQUE###WHERE#### ############################## ###11.8##########

WHERE########################## ## ###### ###############WHERE############## ################################

##############################immutable########### ########################### ############################ ################################## ####### #WHERE##########################IMMUTABLE########################

#####

UNIQUE

  ## ######## ######################

  ##CREATE TABLE##########

CONCURRENTLY

  ##############PostgreSQL## ## ######################### ########## ########################

##########CREATE INDEX###
#######

IF NOT EXISTS

################################ ############## ############################
###################### IF NOT EXISTS###################

INCLUDE

######INCLUDE############################## ############################
############################
#################
############################

############################
################# ##
############ ##################B-tree##############

INCLUDE###
##############

###########################INCLUDE##############

#########B-tree#GiST#SP-GiST###################### ###########INCLUDE#####
###

name

################ ######################### ############################
##
######## ##########PostgreSQL#######################################

ONLY

###############

table_name

#####################################

method

################### btree#hash#gist#spgist#gin#brin####bloom##########
################### ##########btree###

column_name

###########

expression

######1############## ########################### ###########################
##########

collation

##################### ##
########## ##

opclass

########### ###############

opclass_parameter

################ ##############

ASC

##############(##########)#

DESC

##############

NULLS FIRST

NULL##NULL################# ###DESC################

NULLS LAST

NULL##NULL################# ###DESC#################

NULLS DISTINCT
NULLS NOT DISTINCT

##########NULL####(#####)############## ########NULL####################1# ######NULL###########

storage_parameter

######################### ############################

tablespace_name

################### ###########default_tablespace################### ##temp_tablespaces########

predicate

###############

##############

WITH########################## ################################## B-tree### ###GiST###SP-GiST##########################

fillfactor(integer)

B-tree### ### ## B-tree###############90# ###10##100#####################

###################B-tree#########(##############)CREATE INDEX############## ##################### 50##90#######B-tree################################## ####(###) 64.1.4.2#####B-tree###(# ###################)#####################

#######################1#######CREATE INDEX##########100################ ##########(###################)######################## ################100 ##

###

B-tree##########################

`deduplicate_items` (`boolean`)

64.1.4.3#######B-tree############### ##############ON#OFF####### (19.1######## ##ON#OFF##############) ######ON###

```
##

ALTER INDEX#deduplicate_items#################################
#####################################
```

GiST##########################

`buffering` (`enum`)

64.2.4.1## #######OFF#####ON#######
##AUTO##########################effective_cache_size#################
#AUTO### ###############buffering=ON##################################
##

GIN##########################

`fastupdate` (`boolean`)

#####64.4.4.1########################## ###############ON##########OFF######
######ON###

```
##

ALTER     INDEX#####fastupdate#################################
#########  ###########################  #########################
#VACUUM####gin_clean_pending_list###############
```

`gin_pending_list_limit` (`integer`)

gin_pending_list_limit############ ###############

BRIN######################

`pages_per_range` (`integer`)

BRIN###############1#################################64.5.1#### ######128#
##

`autosummarize` (`boolean`)

###64.5.1.1##########
######off###

###########

################################### ##PostgreSQL##############################
######################## ##
######################## ### ###########

############################ ##
########################

PostgreSQL############################## CREATE INDEX#CONCURRENTLY#########
############ ############PostgreSQL#####2############### ####################
################################
################# ##
###############CPU####################################

###################1##2###########2#####
########### ## 2####
##############2####################13#######################################
###
#################### ######################################CREATE INDEX######
################################
########

##CREATE INDEX##############
############## ## ###################
psql#\d#################INVALID########

```
postgres=# \d tab
       Table "public.tab"
 Column |  Type   | Collation | Nullable | Default
--------+---------+-----------+----------+---------
 col    | integer |           |          |
Indexes:
    "idx" btree (col) INVALID
```

###########################CREATE INDEX CONCURRENTLY######### ####REINDEX
INDEX CONCURRENTLY#########################

################################ 2###
########## ##
######## ###2##

################################ ##

############################1##########
########CREATE INDEX##############
##################CREATE INDEX CONCURRENTLY#######################

###################################### ######################################
##################
##################

##

##11###########

####B-tree#GiST#GIN#BRIN## ###############
#####INCLUDE################# #######INCLUDE#####32######## ######PostgreSQL##
############# ###B-tree######################

##############################
####4#########B-tree#########int4_ops######## ###########4################### ##
################################## ##############################2###############
#################2###############
#################################### ###############11.10#36.16##########

CREATE INDEX### ############# ### ## ## ONLY## #ALTER INDEX ... ATTACH PARTITION## #######ONLY######## #####CREATE TABLE ... PARTITION OF#################################### #########

##############################B-Tree######ASC#DESC#NULLS FIRST#NULLS LAST# (####)################################## ################################### ######DESC################### ############################## ############ ##SELECT ... ORDER BY x ASC, y DESC#################################### ################### NULLS##NULL####### ######################### ##########NULL#############

############################## ## ####### ##ANALYZE#######autovacuum###### #####################

######CREATE INDEX#####search_path#####pg_catalog, pg_temp#######

##############################maintenance_work_mem######### ################## ################# ##

PostgreSQL#####################CPU################## ##################### ## ####################################B-Tree########maintenance_work_mem# ## ############### ################################### ##############maintenance_work_mem## ######### ########maintenance_work_mem#########32MB################## ##maintenance_work_mem## ############ ###32MB################## max_parallel_maintenance_workers#################################### ##########################I/O################################### ############## CPU#########

ALTER TABLE####parallel_workers##################CREATE INDEX########## ##################### ################maintenance_work_mem############## ############## ALTER TABLE####parallel_workers#0########################## ################

###
##################parallel_workers#################################### parallel_workers##

CONCURRENTLY#######CREATE INDEX## ########

##############DROP INDEX#########

##############################CREATE INDEX###################VACUUM######## ##############

###PostgreSQL##R-tree################ GiST#################################### ## ##########GiST############USING rtree########CREATE INDEX#USING gist# ######

CREATE INDEX##################pg_stat_progress_create_index######### ##27.4.4#########

# #

####films##title#####B-tree############

```
CREATE UNIQUE INDEX title_idx ON films (title);
```

####films##title###director##rating########B-tree############

```
CREATE UNIQUE INDEX title_idx ON films (title) INCLUDE (director, rating);
```

##########B-Tree#######

```
CREATE INDEX title_idx ON films (title) WITH (deduplicate_items = off);
```

##########################lower(title)################

```
CREATE INDEX ON films ((lower(title)));
```

(########################## ########films_lower_idx#################)

#########################

```
CREATE INDEX title_idx_german ON films (title COLLATE "de_DE");
```

#########NULL######################

```
CREATE INDEX title_idx_nulls_low ON films (title NULLS FIRST);
```

###############################

```
CREATE UNIQUE INDEX title_idx ON films (title) WITH (fillfactor = 70);
```

##########GIN############

```
CREATE INDEX gin_idx ON documents_table USING GIN (locations) WITH
 (fastupdate = off);
```

####films###code################ ##################indexspace########

```
CREATE INDEX code_idx ON films (code) TABLESPACE indexspace;
```

###########box##############point###GiST############

```
CREATE INDEX pointloc
    ON points USING gist (box(location,location));
SELECT * FROM points
    WHERE box(location,location) && '(0,0),(1,1)'::box;
```

###############################

```
CREATE INDEX CONCURRENTLY sales_quantity_index ON sales_table (quantity);
```

###

`CREATE INDEX`#PostgreSQL###### ##SQL#####################

####

ALTER INDEX, DROP INDEX, REINDEX, 27.4.4

CREATE LANGUAGE

CREATE LANGUAGE — ############

##

```
CREATE [ OR REPLACE ] [ TRUSTED ] [ PROCEDURAL ] LANGUAGE name
    HANDLER call_handler [ INLINE inline_handler ] [ VALIDATOR valfunction
 ]
CREATE [ OR REPLACE ] [ TRUSTED ] [ PROCEDURAL ] LANGUAGE name
```

##

CREATE LANGUAGE#########PostgreSQL############ ############################ #######

CREATE LANGUAGE## ################56### #######

CREATE OR REPLACE LANGUAGE######################### ##################### ################################### ####################################

#####################################PostgreSQL#################### ######## ### (###### ######C###)

##############CREATE LANGUAGE######## ###################CREATE EXTENSION####### ## #################

#####

TRUSTED

TRUSTED### ############## #######PostgreSQL##

PROCEDURAL

############

name

############# ############################

HANDLER *call_handler*

call_handler## ###### ##Version-1##########C################################ #language_handler#########PostgreSQL################ language_handler#####################################

INLINE *inline_handler*

inline_handler##################DO################################ *inline_handler*################################### #########DO######### ##internal#####1###############void###### ###############

VALIDATOR *valfunction*

valfunction## ### ################################## #######oid#####1########### ############ OID###### #####void#######

################ereport()######## ##

##

###########DROP LANGUAGE#########

########pg_language# 51.29################################### ##psql#####\dL## ##################

###########################USAGE############### ####################PUBLIC#### #####USAGE######### ####################

################### #######template1###################### ################## ############################

# #

##########################

```
CREATE FUNCTION plsample_call_handler() RETURNS language_handler
    AS '$libdir/plsample'
    LANGUAGE C;
CREATE LANGUAGE plsample
    HANDLER plsample_call_handler;
```

###

```
CREATE EXTENSION plsample;
```

###

CREATE LANGUAGE#PostgreSQL######

####

ALTER LANGUAGE, CREATE FUNCTION, DROP LANGUAGE, GRANT, REVOKE

CREATE MATERIALIZED VIEW

CREATE MATERIALIZED VIEW — ##################

##

```
CREATE MATERIALIZED VIEW [ IF NOT EXISTS ] table_name
    [ (column_name [, ...] ) ]
    [ USING method ]
    [ WITH ( storage_parameter [= value] [, ... ] ) ]
    [ TABLESPACE tablespace_name ]
    AS query
    [ WITH [ NO ] DATA ]
```

##

CREATE MATERIALIZED VIEW######################### ######################### ###WITH NO DATA######################### #####REFRESH MATERIALIZED VIEW# ###############

CREATE MATERIALIZED VIEW#CREATE TABLE AS################################ ###################### ##

CREATE MATERIALIZED VIEW##########################CREATE#######

#####

IF NOT EXISTS

################################## ############### ######################### ##################################

table_name

############################## ## ######################################

column_name

#################### ##################################

USING *method*

###TABLE########### ######### ########61########### ## ################ ########default_table_access_method#########

WITH (*storage_parameter* [= *value*] [, ...])

##########################(####)####### #######CREATE TABLE############### ###### CREATE TABLE################CREATE MATERIALIZED VIEW######## ## #######CREATE TABLE#########

TABLESPACE *tablespace_name*

tablespace_name############################ ###########default_tablespace## #####

query

SELECT#TABLE###VALUES###### ############################## ############ ################

WITH [NO] DATA

##################################### ################################ ##REFRESH MATERIALIZED VIEW#####################

###

CREATE MATERIALIZED VIEW#PostgreSQL######

####

ALTER MATERIALIZED VIEW, CREATE TABLE AS, CREATE VIEW, DROP MATERIALIZED VIEW, REFRESH MATERIALIZED VIEW

CREATE OPERATOR

CREATE OPERATOR — ##########

##

```
CREATE OPERATOR name (
    {FUNCTION|PROCEDURE} = function_name
    [, LEFTARG = left_type ] [, RIGHTARG = right_type ]
    [, COMMUTATOR = com_op ] [, NEGATOR = neg_op ]
    [, RESTRICT = res_proc ] [, JOIN = join_proc ]
    [, HASHES ] [, MERGES ]
)
```

##

CREATE OPERATOR#######*name*####### ###################### ################### ################### ##############################

##########################NAMEDATALEN-1########63###########

+ - * / < > = ~ ! @ # % ^ & | ` ?

#########################

- --#/*##
- ########################1###########+###-#############

  ~ ! @ # % ^ & | ` ?

  ####@-###############*-######## ########PostgreSQL#SQL######################## ########################

- ##=>#SQL###########################

###!=#####<>#############2###########

########LEFTARG#RIGHTARG################ #######RIGHTARG################

function_name####CREATE FUNCTION##1## #2#####################

CREATE OPERATOR###########FUNCTION#PROCEDURE############################## ################ #########PROCEDURE########################

#####[90.370370]##############(####)### #######36.15##########

#####[95.017794]############## ##############USAGE##############EXECUTE######### ###### ##

#####

name

> ############## ################### ########CREATE OPERATOR myschema.+ (...)################ ############################## ######################### ######2#################### ################

function_name

###################

left_type

################################## #####################

right_type

#################

com_op

#############

neg_op

###########

res_proc

################

join_proc

################

HASHES

###########################

MERGES

##########################

#############*com_op*##########################OPERATOR()###########

```
COMMUTATOR = OPERATOR(myschema.===) ,
```

##

#####[57.142857]#######36.14######### ########36.14#36.15#########

################################ ######################## ######################
############################?######3###########

- ######1###################COMMUTATOR######2###########COMMUTATOR######## ########### PostgreSQL##########################2##########################
  #######COMMUTATOR########

- ######## 1 #################COMMUTATOR######### PostgreSQL########
  ##COMMUTATOR## ########
  ## PostgreSQL################## #####
  ############################## ##2#################2#########################
  ### ##

- #################COMMUTATOR############ALTER OPERATOR##################
  ######## #########ALTER#########

In all three cases, you must own both operators in order to mark them as commutators.

##################################

CREATE OPERATOR######################## ######################## ########### #4.1.6##########

##########SORT1#SORT2#LTCMP####GTCMP## #### ####B-tree### ######## ####################MERGES#####################

#######################DROP OPERATOR########## #################ALTER OPERATOR##########

# #

############box##########################

```
CREATE OPERATOR === (
    LEFTARG = box,
    RIGHTARG = box,
    FUNCTION = area_equal_function,
    COMMUTATOR = ===,
    NEGATOR = !==,
    RESTRICT = area_restriction_function,
    JOIN = area_join_function,
    HASHES, MERGES
);
```

###

CREATE OPERATOR#PostgreSQL###### ##SQL#########################

####

ALTER OPERATOR, CREATE OPERATOR CLASS, DROP OPERATOR

CREATE OPERATOR CLASS

CREATE OPERATOR CLASS — ##############

##

```
CREATE OPERATOR CLASS name [ DEFAULT ] FOR TYPE data_type
  USING index_method [ FAMILY family_name ] AS
  {  OPERATOR strategy_number operator_name [ ( op_type, op_type ) ] [ FOR
 SEARCH | FOR ORDER BY sort_family_name ]
    | FUNCTION support_number [ ( op_type [ , op_type ] ) ] function_name
 ( argument_type [, ...] )
    | STORAGE storage_type
  } [, ... ]
```

##

CREATE OPERATOR CLASS################ ##################################### ### ## ############## ### ########################## ##############################

##########################2#######################

########################### ################################## ############### ##

###CREATE OPERATOR CLASS## ################################ ############################

########################### ##########################CREATE OPERATOR CLASS#FAMILY############## ## ##################

#######36.16##########

#####

name

################ ########################

DEFAULT

DEFAULT## ########################### ############1######

data_type

#####################

index_method

############################

family_name

########################### ##
#####

strategy_number

################################

operator_name

##############################

op_type

OPERATOR##########################NONE####### ##########################
#########

FUNCTION##############B-Tree##########################B-tree#########B-tree
#########GiST#SP-GiST#GIN#BRIN##
################ ########################B-tree############FUNCTION#
#*op_type*#############

sort_family_name

##########################btree##################

FOR SEARCH#FOR ORDER BY###########FOR SEARCH########

support_number

####################################

function_name

##

argument_type

###############

storage_type

###################### ###################### ##############################
######GiST#GIN#SP-GiST#BRIN## ##############################STORAGE#####
####### #*data_type*#anyarray##########*storage_type*#anyelement######
###

OPERATOR#FUNCTION#STORAGE#############

##

#################################### ####################PUBLIC##############
################################

####SQL############# SQL##
##########################

PostgreSQL 8.4#######OPERATOR##RECHECK################# ####################
####################################
#################

# #

##########_int4#####int4#####GiST################### ############intarray#### ##########

```
CREATE OPERATOR CLASS gist__int_ops
    DEFAULT FOR TYPE _int4 USING gist AS
        OPERATOR        3       &&,
        OPERATOR        6       = (anyarray, anyarray),
        OPERATOR        7       @>,
        OPERATOR        8       <@,
        OPERATOR        20      @@ (_int4, query_int),
        FUNCTION        1       g_int_consistent (internal, _int4,
 smallint, oid, internal),
        FUNCTION        2       g_int_union (internal, internal),
        FUNCTION        3       g_int_compress (internal),
        FUNCTION        4       g_int_decompress (internal),
        FUNCTION        5       g_int_penalty (internal, internal,
 internal),
        FUNCTION        6       g_int_picksplit (internal, internal),
        FUNCTION        7       g_int_same (_int4, _int4, internal);
```

###

CREATE OPERATOR CLASS#PostgreSQL###### ##SQL##CREATE OPERATOR CLASS#### ####

####

ALTER OPERATOR CLASS, DROP OPERATOR CLASS, CREATE OPERATOR FAMILY, ALTER OPERATOR FAMILY

CREATE OPERATOR FAMILY

CREATE OPERATOR FAMILY — ###########

##

```
CREATE OPERATOR FAMILY name USING index_method
```

##

CREATE OPERATOR FAMILY############## ################################## ## ############################## ################################### ## ####################

################ ####################CREATE OPERATOR CLASS############### ### ##############################ALTER OPERATOR FAMILY##########

################################## ##################### ##################### ###############2##################

######################## ########################### ########################### ########################

###36.16#########

#####

name

: ############# ###############

index_method

: #########################

###

CREATE OPERATOR FAMILY#PostgreSQL###### ##SQL##CREATE OPERATOR FAMILY## ######

####

ALTER OPERATOR FAMILY, DROP OPERATOR FAMILY, CREATE OPERATOR CLASS, ALTER OPERATOR CLASS, DROP OPERATOR CLASS

CREATE POLICY

CREATE POLICY — ##########################

##

```
CREATE POLICY name ON table_name
    [ AS { PERMISSIVE | RESTRICTIVE } ]
    [ FOR { ALL | SELECT | INSERT | UPDATE | DELETE } ]
    [ TO { role_name | PUBLIC | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER } [, ...] ]
    [ USING ( using_expression ) ]
    [ WITH CHECK ( check_expression ) ]
```

##

CREATE POLICY############################ ################ALTER TABLE ... ENABLE ROW LEVEL SECURITY##

#####################select/insert/update/delete######### ##########USING######### ###### INSERT###UPDATE#############WITH CHECK################ ###### #USING##true#######################false###null############### #####WITH CHECK# #true#######################false###null################

INSERT##UPDATE####MERGE###BEFORE##############################WITH CHECK######## ####BEFORE ROW### ### WITH CHECK####################

################## ####1########################## ########################### #####

################################### ### ######### ##################### ############### # 297########################### ####################

USING##WITH CHECK#################ALL#UPDATE#####WITH CHECK############ ################USING#########################WITH CHECK#######USING####### ##

###

#####

name

##############################

table_name

############################

PERMISSIVE

###########(permissive)############### ############################OR######## #### ### ####################

RESTRICTIVE

###########(restrictive)############### ##########################AND#######
##
#####

###1#################
#################################### ##########################1########
##

command

################ #########ALL#SELECT#INSERT#UPDATE#DELETE### ######ALL#
############################

role_name

############### ######PUBLIC#######################

using_expression

###SQL#######boolean#### ########################### ###################
################### ####true############ ####false###null###########SELECT###
############UPDATE####DELETE################ #########################

check_expression

###SQL#######boolean#### ########################### ##################
#############INSERT###UPDATE############# #######true############ #######
#########################false###null#################### *check_expression*##
##

###########

ALL

#####ALL## ALL############
##################ALL############################## ###ALL##############
############### #####USING#####################USING#######

###UPDATE#########ALL#######UPDATE################USING########
##UPDATE##############################WITH CHECK####################
#USING################### INSERT###UPDATE#####ALL#WITH CHECK#########
##########################

SELECT

#####SELECT############SELECT################################SELECT###
###################### #####SELECT#######SELECT################### #
##UPDATE##SELECT#############SELECT######################## SELECT####
################################WITH CHECK############

INSERT

#####INSERT#############INSERT#####INSERT######MERGE################
##################################INSERT############# INSERT#######
##########################USING############

ON CONFLICT DO UPDATE###INSERT###INSERT#####WITH CHECK#####INSERT#
##

UPDATE

#####UPDATE#############UPDATE#####SELECT FOR UPDATE#####SELECT FOR SHARE###########INSERT#####ON CONFLICT DO UPDATE################ UPDATE#####MERGE############ UPDATE#################################### ##########UPDATE#####USING##WITH CHECK########### USING##UPDATE###### ################################## ###WITH CHECK######################### ######

######WITH CHECK############################## USING################ #USING#WITH CHECK#############

####UPDATE#####################################WHERE####RETURNING##### #SET######### ################SELECT########UPDATE###########SELECT## #ALL########### ########UPDATE###ALL########################## ##SELECT###ALL##############################

INSERT########ON CONFLICT DO UPDATE#####UPDATE########### ################UPDATE#####USING####################WITH CHECK####### ## #######UPDATE##############USING####################UPDATE########### ################

DELETE

#####DELETE############DELETE#################### ###########DELETE#### ####### SELECT########################### ######DELETE#####USING####### #####

########DELETE#####################################WHERE####RETURNING### ############SELECT########DELETE###########SELECT######ALL######### ## ########DELETE###ALL############################SELECT###ALL####### ########################

DELETE##WITH CHECK######## ####

#297

####	SELECT/ALL ####	INSERT/ALL ####	UPDATE/ALL####		DELETE/ALL ####
	USING#	WITH CHECK #	USING#	WITH CHECK #	USING#
SELECT	####	—	—	—	—
SELECT FOR UPDATE/ SHARE	####	—	####	—	—
INSERT / MERGE ... THEN INSERT	—	####	—	—	—
INSERT ... RETURNING	#### [a]	####	—	—	—
UPDATE / MERGE ... THEN UPDATE	######### [a]	—	####	####	—
DELETE	#### [a]	—	—	—	####

####	**SELECT/ALL ####**	**INSERT/ALL ####**	**UPDATE/ALL####**		**DELETE/ALL ####**
	USING#	**WITH CHECK #**	**USING#**	**WITH CHECK #**	**USING#**
ON CONFLICT DO UPDATE	#########	—	####	####	—

[a] ####################################WHERE####RETURNING#########

###########

##UPDATE#########SELECT#UPDATE#########
##############
#############AND#############

###1##PERMISSIVE#########
###########RESTRICTIVE############### ########PERMISSIVE######OR###########
##RESTRICTIVE######AND##############AND########### PERMISSIVE##############
#######

######################ALL###

####SELECT#UPDATE############UPDATE###
######

```
expression from RESTRICTIVE SELECT/ALL policy 1
AND
expression from RESTRICTIVE SELECT/ALL policy 2
AND
...
AND
(
  expression from PERMISSIVE SELECT/ALL policy 1
  OR
  expression from PERMISSIVE SELECT/ALL policy 2
  OR
  ...
)
AND
expression from RESTRICTIVE UPDATE/ALL policy 1
AND
expression from RESTRICTIVE UPDATE/ALL policy 2
AND
...
AND
(
  expression from PERMISSIVE UPDATE/ALL policy 1
  OR
  expression from PERMISSIVE UPDATE/ALL policy 2
  OR
  ...
)
```

##

#####################################

###
####1#####################

################ ############################## ########################
########################### ##
######### ############################## ################################
##
#####################################

########################
############################## ######################LEAKPROOF###############
##

##################
####################################
######################### ########################### #######################
###
#security_invoker#####(CREATE VIEW###)##################

MERGE############### ######SELECT#INSERT#UPDATE###DELETE#################
#####MERGE############

################5.9##########

###

CREATE POLICY#PostgreSQL######

####

ALTER POLICY, DROP POLICY, ALTER TABLE

CREATE PROCEDURE

CREATE PROCEDURE — #############

##

```
CREATE [ OR REPLACE ] PROCEDURE
    name ( [ [ argmode ] [ argname ] argtype [ { DEFAULT |
 = } default_expr ] [, ...] ] )
  { LANGUAGE lang_name
    | TRANSFORM { FOR TYPE type_name } [, ... ]
    | [ EXTERNAL ] SECURITY INVOKER | [ EXTERNAL ] SECURITY DEFINER
    | SET configuration_parameter { TO value | = value | FROM CURRENT }
    | AS 'definition'
    | AS 'obj_file', 'link_symbol'
    | sql_body
  } ...
```

##

CREATE PROCEDURE################ CREATE OR REPLACE PROCEDURE########## ################ ####################USAGE########

##########cstring######################## ############################## SQL###################

################ *table_name.column_name*%TYPE. ######################### ################

default_expr

################################## ########################### ########### ##

lang_name

#################### #########sql#c#internal################plpgsql## ######### *sql_body*###############sql### ################################ ###

TRANSFORM { FOR TYPE *type_name* } [, ...] }

############################ ###SQL######################CREATE TRANSFORM##### ## #### ## ###

[EXTERNAL] SECURITY INVOKER
[EXTERNAL] SECURITY DEFINER

SECURITY INVOKER## ########### SECURITY DEFINER######################################

EXTERNAL#######SQL################## ####SQL############################ ############################

SECURITY DEFINER############################COMMIT#ROLLBACK######## ##

configuration_parameter
value

SET## SET FROM CURRENT##CREATE PROCEDURE#####################################

#######SET#######################SET LOCAL################################ ################################### ########LOCAL####SET#####SET######## # ########SET LOCAL############### ## ###################

#######SET##COMMIT#ROLLBACK##### #####

####################SET##19###########

definition

################### ################### ##########################SQL###### ##########################

###4.1.2.4##################### ######## ##

obj_file, link_symbol

#####AS###########C############C##################SQL################## ##### *obj_file*##############C#######################LOAD###############

###*link_symbol*#####################C#####################
############SQL#######################

################CREATE PROCEDURE#####################################
##

sql_body

LANGUAGE SQL########### #######################

```
BEGIN ATOMIC
  statement;
  statement;
  ...
  statement;
END
```

##############################(###*definition*#########)############# ####
#LANGUAGE SQL################################ #########################
############# ## #############
#####################################DROP ... CASCADE###################
############################ ###########SQL###SQL##############

##

#########################CREATE FUNCTION##########

############CALL######

#

```
CREATE PROCEDURE insert_data(a integer, b integer)
LANGUAGE SQL
AS $$
INSERT INTO tbl VALUES (a);
INSERT INTO tbl VALUES (b);
$$;
```

###

```
CREATE PROCEDURE insert_data(a integer, b integer)
LANGUAGE SQL
BEGIN ATOMIC
  INSERT INTO tbl VALUES (a);
  INSERT INTO tbl VALUES (b);
END;
```

#################

```
CALL insert_data(1, 2);
```

###

CREATE PROCEDURE#######SQL######### PostgreSQL############################
####CREATE FUNCTION#########

####

ALTER PROCEDURE, DROP PROCEDURE, CALL, CREATE FUNCTION

CREATE PUBLICATION

CREATE PUBLICATION — ###############

##

```
CREATE PUBLICATION name
    [ FOR ALL TABLES
      | FOR publication_object [, ... ] ]
    [ WITH ( publication_parameter [= value] [, ... ] ) ]
```

```
###publication_object###########

    TABLE [ ONLY ] table_name [ * ] [ ( column_name [, ... ] ) ] [ WHERE
 ( expression ) ] [, ... ]
    TABLES IN SCHEMA { schema_name | CURRENT_SCHEMA } [, ... ]
```

##

CREATE PUBLICATION########################### ############################## ###########################

##29.1##########

#####

name

################

FOR TABLE

########################## ########ONLY################################## ## ONLY## ###############*####### ######################### ############################## ################## ##

######WHERE#################### *expression*#####NULL################### ######################## TRUNCATE############

######################### ## ########### TRUNCATE############ ########29.5##########

######################################

#########FOR TABLES IN SCHEMA##################################

##

FOR ALL TABLES

###

FOR TABLES IN SCHEMA

###

###

#################
##

###
################################### ####################################
####################

WITH (*publication_parameter* [= *value*] [, ...])

############################## ###################

publish (string)

#######################DML########################### #################
#######insert#update#delete#truncate###
################'insert, update, delete, truncate'###

#######DML########## #####################(29.8.1###)#################
###################

publish_via_partition_root (boolean)

################################(############)##########################
###
########## ##
###########

1#################################### publish_via_partition_root
= true###(#####
######)####################################ID####################

###############

##########################TRUNCATE##########

boolean##############= *value*#############TRUE###########

##

FOR TABLE#FOR ALL TABLES#FOR TABLES IN SCHEMA#########################
############## ######################

################################ ##

#####################################CREATE################ ##################
#########

FOR ALL TABLES##FOR
TABLES IN SCHEMA##############################

UPDATE###DELETE###########################REPLICA IDENTITY##############
#####################################

UPDATE###DELETE#####################REPLICA IDENTITY############### #####
####INSERT##############################

```
######(#####WHERE#)###UPDATE###DELETE##############REPLICA IDENTITY######
############## INSERT###########WHERE############# ######################
##########################################################

FOR TABLES IN SCHEMA##############################################

##################################################
#publish_via_partition_root################################(#####)########
########### #########29.4######### ##########################################
###########publish_via_partition_root####################################
###############

INSERT ... ON CONFLICT########################################### #######
#INSERT####UPDATE######################################

MERGE####################################INSERT#UPDATE####DELETE###########

publish_via_partition_root#true########################################
#####ATTACH######################

COPY ... FROM#####INSERT################

DDL###############

WHERE##############################
```

#

```
2####################################


CREATE PUBLICATION mypublication FOR TABLE users, departments;

#####(active departments)################################


CREATE PUBLICATION active_departments FOR TABLE departments WHERE (active
 IS TRUE);

#######################################


CREATE PUBLICATION alltables FOR ALL TABLES;

########INSERT###########################


CREATE PUBLICATION insert_only FOR TABLE mydata
    WITH (publish = 'insert');

####users#departments##############production###########################
##############


CREATE PUBLICATION production_publication FOR TABLE users, departments,
 TABLES IN SCHEMA production;

####marketing###sales##############################################


CREATE PUBLICATION sales_publication FOR TABLES IN SCHEMA marketing,
 sales;

####users####################user_id#firstname####################
```

```
CREATE PUBLICATION users_filtered FOR TABLE users (user_id, firstname);
```

###

`CREATE PUBLICATION`#PostgreSQL######

####

ALTER PUBLICATION, DROP PUBLICATION, CREATE SUBSCRIPTION, ALTER SUBSCRIPTION

CREATE ROLE

CREATE ROLE — ################

##

```
CREATE ROLE name [ [ WITH ] option [ ... ] ]
```

```
###option#########

      SUPERUSER | NOSUPERUSER
    | CREATEDB | NOCREATEDB
    | CREATEROLE | NOCREATEROLE
    | INHERIT | NOINHERIT
    | LOGIN | NOLOGIN
    | REPLICATION | NOREPLICATION
    | BYPASSRLS | NOBYPASSRLS
    | CONNECTION LIMIT connlimit
    | [ ENCRYPTED ] PASSWORD 'password' | PASSWORD NULL
    | VALID UNTIL 'timestamp'
    | IN ROLE role_name [, ...]
    | ROLE role_name [, ...]
    | ADMIN role_name [, ...]
    | SYSID uid
```

##

CREATE ROLE##PostgreSQL###################### ################################ ###################### ## ####### ##############21###20########## #############CREATEROLE#################### ##########

###

###
####### ###########################IN ROLE#ROLE#ADMIN######## GRANT#######
###

#####

name

###########

SUPERUSER
NOSUPERUSER

##################################
################################
################NOSUPERUSER###

CREATEDB
NOCREATEDB

################################ CREATEDB################################
####### NOCREATEDB################################## ################
#NOCREATEDB### #############CREATEDB########CREATEDB#######

CREATEROLE
NOCREATEROLE

##
###NOCREATEROLE###

INHERIT
NOINHERIT

##
#############IN ROLE#################ROLE############ ###GRANT#######
###INHERIT########

16####PostgreSQL###

LOGIN
NOLOGIN

##
LOGIN###
############## ################NOLOGIN####CREATE ROLE#######CREATE
USER##############

REPLICATION
NOREPLICATION

##
REPLICATION###
##############NOREPLICATION### ############REPLICATION#######
#REPLICATION#######

BYPASSRLS
NOBYPASSRLS

#########################(RLS)#################### NOBYPASSRLS######### ####
#########BYPASSRLS#########BYPASSRLS#######

pg_dump#############################row_security#######OFF#############
pg_dump##################################
####RLS######

CONNECTION LIMIT *connlimit*

##-1################## ##############
###

[ENCRYPTED] PASSWORD '*password*'
PASSWORD NULL

################ #######LOGIN###
#################################### ####################NULL##############
################ NULL############PASSWORD NULL############

##
###################NULL#######PostgreSQL######10#################### ##libpq########## ###### ################################

```
    ############################## ENCRYPTED#################################
    ## #############password_encryption########## ###############MD5###SCRAM##
    #############password_encryption#####################################
    ################################ #######################################
```

VALID UNTIL '*timestamp*'

```
    VALID UNTIL######################### ###########################
```

IN ROLE *role_name*

```
    IN    ROLE########################################  ########SET#######
    ##ADMIN########## NOINHERIT################INHERIT#############
```

ROLE *role_name*

```
    ROLE############1##########SET##########################  ##############
    ############## ####################INHERIT################INHERIT#####
    ######## ##########ADMIN#############
```

ADMIN *role_name*

```
    ADMIN##ROLE####################ADMIN##################################
    #################
```

SYSID *uid*

```
    SYSID##############################
```

##

```
############ALTER ROLE###########DROP ROLE######### CREATE ROLE########
######ALTER ROLE############

########################################GRANT#REVOKE#########

VALID UNTIL#################################### ##########################
#################

############################CREATEDB#######################INHERIT#######
############################## ############SET##################createdb####SET
ROLE####################

IN ROLE#ROLE#ADMIN##########################################

############INHERIT########## ########PostgreSQL##############################
######### ####NOINHERIT#####SQL####################

PostgreSQL###CREATE  ROLE################################createuser############
###########

CONNECTION LIMIT##################### #######1#######################2####
############################## #######################

#################################### ################### ##################
################## ####createuser#################### ###psql##\password######
##############################
```

#

```
################################
```

```
CREATE ROLE jonathan LOGIN;
```

##################

```
CREATE USER davide WITH PASSWORD 'jw8s0F4';
```

CREATE USER#LOGIN#########CREATE ROLE######

2004######################## 2005##1######################

```
CREATE ROLE miriam WITH LOGIN PASSWORD 'jw8s0F4' VALID UNTIL '2005-01-01';
```

##################################

```
CREATE ROLE admin WITH CREATEDB CREATEROLE;
```

###

CREATE ROLE####SQL######### ###SQL##################

```
CREATE ROLE name [ WITH ADMIN role_name ]
```

##############CREATE ROLE#######PostgreSQL######

##SQL#################################### ################################### ######## PostgreSQL############################## ############################## #######

NOINHERIT#########PostgreSQL#######SQL#######INHERIT#########PostgreSQL## ######SQL##############SQL##################

######USER##ROLE##################

```
USER role_name [, ...]
```

######IN GROUP##IN ROLE##################

```
IN GROUP role_name [, ...]
```

####

SET ROLE, ALTER ROLE, DROP ROLE, GRANT, REVOKE, createuser, createrole_self_grant

CREATE RULE

CREATE RULE — ###############

##

```
CREATE [ OR REPLACE ] RULE name AS ON event
    TO table_name [ WHERE condition ]
    DO [ ALSO | INSTEAD ] { NOTHING | command | ( command ; command
 ... ) }
```

###event#######1####

```
    SELECT | INSERT | UPDATE | DELETE
```

##

CREATE RULE################################### CREATE OR REPLACE RULE### ##

PostgreSQL## ### ## ######INSTEAD### ### #######SQL############## #### ## ######################### # ###################################### #################39##########

######ON SELECT################ ########"_RETURN"##########INSTEAD###### ########SELECT########################## ###################### ############ #################### ######################## ####CREATE OR REPLACE RULE "_RETURN" AS ...############CREATE OR REPLACE VIEW#########

ON INSERT#ON UPDATE#ON DELETE## ## INSERT RETURNING################### ####RETURNING############

####################################### ########################INSTEAD#### ############# #############INSTEAD##################### ############### ##################################### ##################################### DO INSTEAD NOTHING### ##INSTEAD############ ####################INSTEAD NOTHING################# ######## ############RETURNING####################

> ##
>
> ##################CREATE VIEW################################ ######## ##
>
> ##################INSTEAD OF###(CREATE TRIGGER##)########

#####

name

> ############# ################################### ########################### ##########################

event

#######SELECT#INSERT#UPDATE#DELETE####### ON CONFLICT### #INSERT##INSERT######UPDATE############################### ################ #############

table_name

##################################

condition

###SQL######boolean######## ######NEW###OLD################ ################ #####

INSTEAD

INSTEAD################################

ALSO

ALSO##############################

ALSO#INSTEAD###########ALSO#########

command

############################ ####### ##SELECT#INSERT#UPDATE#DELETE#NOTIFY###

condition#command##################################NEW#OLD####### NEW##ON INSERT#ON UPDATE############################ OLD##ON UPDATE#ON DELETE#### ########################

##

###

#####INSERT#UPDATE#DELETE######RETURNING###################### ####INSERT RETURNING#UPDATE RETURNING#DELETE RETURNING#################### ################ RETURNING####################RETURNING######## ########### #INSTEAD######RETURNING########## ########################1##RETURNING### ######### ########################RETURNING#####1########## #####RETURNING### ################RETURNING################

####################### #######2############PostgreSQL######################### ##SELECT######PostgreSQL###########

```
CREATE RULE "_RETURN" AS
    ON SELECT TO t1
    DO INSTEAD
        SELECT * FROM t2;

CREATE RULE "_RETURN" AS
    ON SELECT TO t2
    DO INSTEAD
        SELECT * FROM t1;

SELECT * FROM t1;
```

#############NOTIFY###########NOTIFY################ ######################### ###NOTIFY######## ####

```
CREATE RULE notify_me AS ON UPDATE TO mytable DO ALSO NOTIFY mytable;

UPDATE mytable SET name = 'foo' WHERE id = 42;
```

###id = 42####################UPDATE###1##NOTIFY############ ###############
###################

###

CREATE RULE#PostgreSQL#############################

####

ALTER RULE, DROP RULE

CREATE SCHEMA

CREATE SCHEMA — ############

##

```
CREATE SCHEMA schema_name [ AUTHORIZATION role_specification ]
 [ schema_element [ ... ] ]
CREATE SCHEMA AUTHORIZATION role_specification [ schema_element [ ... ] ]
CREATE SCHEMA IF NOT EXISTS schema_name [ AUTHORIZATION role_specification
 ]
CREATE SCHEMA IF NOT EXISTS AUTHORIZATION role_specification
```

```
###role_specification#########

    user_name
  | CURRENT_ROLE
  | CURRENT_USER
  | SESSION_USER
```

##

CREATE SCHEMA############################## ##############################
############

################# ## ############
################################## ##
############################## ################CREATE####################
#current_schema#########################

CREATE SCHEMA## #########
############################## ####AUTHORIZATION#########################
######################

#####

schema_name

############# #######*user_name*############### ######pg_############# ###
##########################

user_name

###################### ################################## ##################
###############SET ROLE############

schema_element

#######################SQL#### ###CREATE SCHEMA####CREATE TABLE#CREATE
VIEW#CREATE INDEX#CREATE SEQUENCE#CREATE TRIGGER####GRANT#########
####################################

IF NOT EXISTS

#################################### ###############*schema_element*####
#############

##

##################################CREATE################ ########################
###

# #

##########

```
CREATE SCHEMA myschema;
```

joe################ #########joe######

```
CREATE SCHEMA AUTHORIZATION joe;
```

test#######################joe###########test################ #joe########
###################

```
CREATE SCHEMA IF NOT EXISTS test AUTHORIZATION joe;
```

###########################

```
CREATE SCHEMA hollywood
    CREATE TABLE films (title text, release date, awards text[])
    CREATE VIEW winners AS
        SELECT title, release FROM films WHERE awards IS NOT NULL;
```

##################################

#############################

```
CREATE SCHEMA hollywood;
CREATE TABLE hollywood.films (title text, release date, awards text[]);
CREATE VIEW hollywood.winners AS
    SELECT title, release FROM hollywood.films WHERE awards IS NOT NULL;
```

###

##SQL###CREATE SCHEMA#DEFAULT CHARACTER SET######## #####PostgreSQL#####
####################

##SQL###CREATE SCHEMA#################### ###PostgreSQL#####################
########### ##

##SQL#################################### PostgreSQL#########################
################### #########################CREATE##########################
#####################

IF NOT EXISTS######PostgreSQL######

####

ALTER SCHEMA, DROP SCHEMA

CREATE SEQUENCE

CREATE SEQUENCE — ##################

##

```
CREATE [ { TEMPORARY | TEMP } | UNLOGGED ] SEQUENCE [ IF NOT EXISTS ] name
    [ AS data_type ]
    [ INCREMENT [ BY ] increment ]
    [ MINVALUE minvalue | NO MINVALUE ] [ MAXVALUE maxvalue | NO
 MAXVALUE ]
    [ START [ WITH ] start ] [ CACHE cache ] [ [ NO ] CYCLE ]
    [ OWNED BY { table_name.column_name | NONE } ]
```

##

CREATE SEQUENCE######################## ########name#########1############## ########### ####################################

########################(### ####################################)##################

###########nextval#currval#setval################## ###########9.17######## ##

#####################################

```
SELECT * FROM name;
```

#################################### #########last_value################## ########### ##################nextval################################

#####

TEMPORARY####TEMP

##

UNLOGGED

################################### ##################################### ## ################ ### ### ################################

###########ID########### ################################## ##############WAL########################

IF NOT EXISTS

############################## ############### ############################## ########################### ####################

name

#########################

data_type

######AS *data_type*################# #####smallint#integer#bigint### bigint######## ################################

increment

INCREMENT BY *increment*## ### ## ##############1###

minvalue
NO MINVALUE

MINVALUE *minvalue*################################## ################ ##NO MINVALUE################### ################1### ################ #############

maxvalue
NO MAXVALUE

MAXVALUE *maxvalue*############################ ##################NO MAXVALUE################### ############################## ############ ####-1###

start

START WITH *start*### ########### ##############*minvalue*######*maxvalue*######

cache

CACHE *cache*## ##### ####1###########1#####################################

CYCLE
NO CYCLE

CYCLE##########################*maxvalue*######*minvalue*############### ########### ##########################*minvalue*######*maxvalue*######

NO CYCLE######################nextval############### CYCLE#NO CYCLE## ##########NO CYCLE###########

OWNED BY *table_name.column_name*
OWNED BY NONE

OWNED BY## ############# ### ##### #OWNED BY NONE####################

##

###########DROP SEQUENCE#########

######bigint########## #####8########-9223372036854775808##9223372036854775807 ###############

nextval#setval###
############# ##
##

###########cache###1###
####################1################################## #######################
#####last_value######## #################cache-1#######nextval##############
############################## ##
###################

###
####cache#10######## #####A##1##10######nextval=1###### #####B########A
#nextval=2######11##20#####nextval=11###### ######cache#1########nextval###
#######################cache#1##############nextval#######################
############### ###last_value####nextval##################################
###########

#################setval###
############

# #

101#####serial###################

```
CREATE SEQUENCE serial START 101;
```

###################

```
SELECT nextval('serial');

 nextval
---------
     101
```

###################

```
SELECT nextval('serial');

 nextval
---------
     102
```

########INSERT###########

```
INSERT INTO distributors VALUES (nextval('serial'), 'nothing');
```

COPY FROM###############

```
BEGIN;
COPY distributors FROM 'input_file';
SELECT setval('serial', max(id)) FROM distributors;
END;
```

###

#########CREATE SEQUENCE###SQL######

- ##############NEXT VALUE FOR#####nextval()#########
- OWNED BY##PostgreSQL######

####

ALTER SEQUENCE, DROP SEQUENCE

CREATE SERVER

CREATE SERVER — ############

##

```
CREATE SERVER [ IF NOT EXISTS ] server_name [ TYPE 'server_type' ]
 [ VERSION 'server_version' ]
    FOREIGN DATA WRAPPER fdw_name
    [ OPTIONS ( option 'value' [, ... ] ) ]
```

##

CREATE SERVER############### ######################

##############

#########################

#########################USAGE########

#####

IF NOT EXISTS

############################ ################### ##########################
############################

server_name

##############

server_type

######(####)### ###################

server_version

########(####)### ###################

fdw_name

########################

OPTIONS (*option* '*value*' [, ...])

################### ##

##

dblink####################################dblink_connect################### ######
###################USAGE########

###

# #

#########postgres_fdw#####myserver##########

```
CREATE SERVER myserver FOREIGN DATA WRAPPER postgres_fdw OPTIONS (host
 'foo', dbname 'foodb', port '5432');
```

#######postgres_fdw##########

###

CREATE SERVER#ISO/IEC 9075-9 (SQL/MED)######

####

ALTER SERVER, DROP SERVER, CREATE FOREIGN DATA WRAPPER, CREATE FOREIGN TABLE, CREATE USER MAPPING

CREATE STATISTICS

CREATE STATISTICS — ##########

##

```
CREATE STATISTICS [ [ IF NOT EXISTS ] statistics_name ]
    ON ( expression )
    FROM table_name

CREATE STATISTICS [ [ IF NOT EXISTS ] statistics_name ]
    [ ( statistics_kind [, ... ] ) ]
    ON { column_name | ( expression ) }, { column_name | ( expression ) }
 [, ...]
    FROM table_name
```

##

CREATE STATISTICS### ### ######################################

CREATE STATISTICS######2############# 1#######1############################ ################################## ### #####2## ################### ############

###############CREATE STATISTICS myschema.mystat ...################### ######## ############################ ## ##############

#####

IF NOT EXISTS

################################## ################### ################### ############################ IF NOT EXISTS##################

statistics_name

########################## ##########PostgreSQL######################### #########

statistics_kind

######################### ############N##########ndistinct############ #dependencies############mcv### ## ## ########14.2.2###68.2######## ##

column_name

################### ###################### #####2######################### ########

expression

############## ###1## #### ##################################

table_name

##ANALYZE#######
##

##

##################################
#########

################
#################

######PostgreSQL##########
############

# #

#######2#########t1###### #####1#################################### ######
###################

```
CREATE TABLE t1 (
    a   int,
    b   int
);

INSERT INTO t1 SELECT i/100, i/500
                 FROM generate_series(1,1000000) s(i);

ANALYZE t1;


-- ####################
EXPLAIN ANALYZE SELECT * FROM t1 WHERE (a = 1) AND (b = 0);

CREATE STATISTICS s1 (dependencies) ON a, b FROM t1;

ANALYZE t1;


-- ###############
EXPLAIN ANALYZE SELECT * FROM t1 WHERE (a = 1) AND (b = 0);
```

#################2##WHERE## ######
##########WHERE###########################

(##########)#######2#########t2#####2######(MCV)#########

```
CREATE TABLE t2 (
    a   int,
    b   int
);

INSERT INTO t2 SELECT mod(i,100), mod(i,100)
                 FROM generate_series(1,1000000) s(i);

CREATE STATISTICS s2 (mcv) ON a, b FROM t2;

ANALYZE t2;
```

```
-- ########(MCV######)
EXPLAIN ANALYZE SELECT * FROM t2 WHERE (a = 1) AND (b = 1);


-- ########(MCV########)
EXPLAIN ANALYZE SELECT * FROM t2 WHERE (a = 1) AND (b = 2);

##############################################################################
############################

#########1######t3#######################  ##################################
############### ######################################################## ######
#2###########N############


CREATE TABLE t3 (
    a   timestamp
);

INSERT INTO t3 SELECT i FROM generate_series('2020-01-01'::timestamp,
                                             '2020-12-31'::timestamp,
                                             '1 minute'::interval) s(i);

ANALYZE t3;


-- ####################
EXPLAIN ANALYZE SELECT * FROM t3
  WHERE date_trunc('month', a) = '2020-01-01'::timestamp;

EXPLAIN ANALYZE SELECT * FROM t3
  WHERE date_trunc('day', a) BETWEEN '2020-01-01'::timestamp
                                 AND '2020-06-30'::timestamp;

EXPLAIN ANALYZE SELECT date_trunc('month', a), date_trunc('day', a)
   FROM t3 GROUP BY 1, 2;


-- ###########N############
-- (#################)
CREATE STATISTICS s3 (ndistinct) ON date_trunc('month', a),
 date_trunc('day', a) FROM t3;

ANALYZE t3;


-- ################
EXPLAIN ANALYZE SELECT * FROM t3
  WHERE date_trunc('month', a) = '2020-01-01'::timestamp;

EXPLAIN ANALYZE SELECT * FROM t3
  WHERE date_trunc('day', a) BETWEEN '2020-01-01'::timestamp
                                 AND '2020-06-30'::timestamp;

EXPLAIN ANALYZE SELECT date_trunc('month', a), date_trunc('day', a)
   FROM t3 GROUP BY 1, 2;

####N######################################################## #########0.5%#####
##########################(######)########  #######2#############################
##########################2##WHERE#GROUP BY################################
########################  ################################N###################
N#################### ##################################################
```

###

##SQL#`CREATE STATISTICS`##########

####

ALTER STATISTICS, DROP STATISTICS

CREATE SUBSCRIPTION

CREATE SUBSCRIPTION — ################

##

```
CREATE SUBSCRIPTION subscription_name
    CONNECTION 'conninfo'
    PUBLICATION publication_name [, ...]
    [ WITH ( subscription_parameter [= value] [, ... ] ) ]
```

##

CREATE SUBSCRIPTION############################ ######################### ############ ##

################################## ### ###################

###

#####[93.264249]##############################CREATE#####
##pg_create_subscription###########

#################################29.2####29###########

#####

subscription_name

#################

CONNECTION '*conninfo*'

######################libpq######## ###32.1.1##########

PUBLICATION *publication_name* [, ...]

##

WITH (*subscription_parameter* [= *value*] [, ...])

################################

####################################

connect (boolean)

CREATE SUBSCRIPTION################################ # #####true### ###false######create_slot#enabled####copy_data## ##false########### (connect#false########create_slot#enabled### #copy_data#true######################)

#####[77.492877]########false################################## ### #####29.2.3######### ##############false##############################

##
#################### ######29.2.3#########

create_slot (boolean)

######true###

false## ######29.2.3#########

enabled (boolean)

######true###

slot_name (string)

############################## ##################################

slot_name#NONE### ############
###enabled#create_slot####false############# ######################
############## ######29.2.3#########

######slot_name##########create_slot##################failover##
#########################failover################## #######
##failover## ##############
##############
#failover## #########
####failover#####################################

##

binary (boolean)

#####
#false### ###########(copy_data##)########## ########################
############PostgreSQL################### ################### #########
#################smallint####integer############# ##################
################################## ####################################
###################(#########CREATE TYPE########)#

###
################################ ##############binary###########

##############PostgreSQL 16##################binary = true###########
##########

copy_data (boolean)

######true#
##

#########WHERE########################### ##############

copy_data = true#origin#####################################

streaming (enum)

#######off############
##

on###
#####

parallel##################################1############## ###############
############
#########################LSN##############################

synchronous_commit (enum)

#####################################synchronous_commit############### ###
####off###

############off############ ##
#################

################################## ######################################
####################
#########################synchronous_commit#off##############
#COMMIT########################### #####synchronous_commit#local####
###################

two_phase (boolean)

##############2###################### ######false###

2###########################PREPARE TRANSACTION#####################
#####2################### ##
##################

2## #####two_phase#####
#########################2#######################pending########### ###2##
#######pg_subscription#subtwophasestate###########

disable_on_error (boolean)

###
######### ######false###

password_required (boolean)

true###
##################### ## #####
#true### #############false########

run_as_owner (boolean)

true### false####################
######################### #######################29.10########## #####
#false###

origin (string)

###
origin#none##
origin#any################################ ######any###

copy_data = true#origin#####################################

failover (boolean)

###
############################## ######false###

boolean##############= *value*##############TRUE############

##

##29.10#########

################################CREATE SUBSCRIPTION##########################
####

###
#################################### #######CREATE SUBSCRIPTION########
####### #############pg_create_logical_replication_slot######pgoutput#
##########################create_slot = false##################### #####
#29.2.3######### ################################

#####[94.514502]###############WHERE#######*expression*#####NULL##############
####################WHERE##########################(#############)###
###################### WHERE#####################WHERE#####(#############)###
######FOR ALL TABLES###FOR TABLES IN SCHEMA#########################
########### ###############PostgreSQL 15#################################### ##
##############################
#####publish##############DML############################## ######29.2.2#####
#####

###

############################### ###pg_subscription##########################
#####

###############copy_data = true#origin = NONE##############################
##################################
####################### ## #####
###

#######(##############################)##############################SQL######
########

```
# substitute <pub-names> below with your publication name(s) to be queried
# ###<pub-names>###########################
SELECT DISTINCT PT.schemaname, PT.tablename
FROM pg_publication_tables PT,
     pg_subscription_rel PS
     JOIN pg_class C ON (C.oid = PS.srrelid)
     JOIN pg_namespace N ON (N.oid = C.relnamespace)
WHERE N.nspname = PT.schemaname AND
      C.relname = PT.tablename AND
      PT.pubname IN (<pub-names>);
```

# #

########mypublication###insert_only#####################################
#################

```
CREATE SUBSCRIPTION mysub
         CONNECTION 'host=192.168.1.50 port=5432 user=foo dbname=foodb'
        PUBLICATION mypublication, insert_only;
```

########insert_only##
###

```
CREATE SUBSCRIPTION mysub
         CONNECTION 'host=192.168.1.50 port=5432 user=foo dbname=foodb'
        PUBLICATION insert_only
               WITH (enabled = false);
```

###

CREATE SUBSCRIPTION#PostgreSQL######

####

ALTER SUBSCRIPTION, DROP SUBSCRIPTION, CREATE PUBLICATION, ALTER PUBLICATION

CREATE TABLE

CREATE TABLE — ###########

##

```
CREATE [ [ GLOBAL | LOCAL ] { TEMPORARY | TEMP } | UNLOGGED ] TABLE [ IF
 NOT EXISTS ] table_name ( [
  { column_name data_type [ STORAGE { PLAIN | EXTERNAL | EXTENDED | MAIN
 | DEFAULT } ] [ COMPRESSION compression_method ] [ COLLATE collation ]
 [ column_constraint [ ... ] ]
    | table_constraint
    | LIKE source_table [ like_option ... ] }
    [, ... ]
] )
[ INHERITS ( parent_table [, ... ] ) ]
[ PARTITION BY { RANGE | LIST | HASH } ( { column_name | ( expression ) }
 [ COLLATE collation ] [ opclass ] [, ... ] ) ]
[ USING method ]
[ WITH ( storage_parameter [= value] [, ... ] ) | WITHOUT OIDS ]
[ ON COMMIT { PRESERVE ROWS | DELETE ROWS | DROP } ]
[ TABLESPACE tablespace_name ]

CREATE [ [ GLOBAL | LOCAL ] { TEMPORARY | TEMP } | UNLOGGED ] TABLE [ IF
 NOT EXISTS ] table_name
    OF type_name [ (
  { column_name [ WITH OPTIONS ] [ column_constraint [ ... ] ]
    | table_constraint }
    [, ... ]
) ]
[ PARTITION BY { RANGE | LIST | HASH } ( { column_name | ( expression ) }
 [ COLLATE collation ] [ opclass ] [, ... ] ) ]
[ USING method ]
[ WITH ( storage_parameter [= value] [, ... ] ) | WITHOUT OIDS ]
[ ON COMMIT { PRESERVE ROWS | DELETE ROWS | DROP } ]
[ TABLESPACE tablespace_name ]

CREATE [ [ GLOBAL | LOCAL ] { TEMPORARY | TEMP } | UNLOGGED ] TABLE [ IF
 NOT EXISTS ] table_name
    PARTITION OF parent_table [ (
  { column_name [ WITH OPTIONS ] [ column_constraint [ ... ] ]
    | table_constraint }
    [, ... ]
) ] { FOR VALUES partition_bound_spec | DEFAULT }
[ PARTITION BY { RANGE | LIST | HASH } ( { column_name | ( expression ) }
 [ COLLATE collation ] [ opclass ] [, ... ] ) ]
[ USING method ]
[ WITH ( storage_parameter [= value] [, ... ] ) | WITHOUT OIDS ]
[ ON COMMIT { PRESERVE ROWS | DELETE ROWS | DROP } ]
[ TABLESPACE tablespace_name ]


###column_constraint#############

[ CONSTRAINT constraint_name ]
{ NOT NULL |
  NULL |
  CHECK ( expression ) [ NO INHERIT ] |
  DEFAULT default_expr |
  GENERATED ALWAYS AS ( generation_expr ) STORED |
```

```
  GENERATED { ALWAYS | BY DEFAULT } AS IDENTITY [ ( sequence_options ) ] |
  UNIQUE [ NULLS [ NOT ] DISTINCT ] index_parameters |
  PRIMARY KEY index_parameters |
  REFERENCES reftable [ ( refcolumn ) ] [ MATCH FULL | MATCH PARTIAL |
 MATCH SIMPLE ]
    [ ON DELETE referential_action ] [ ON UPDATE referential_action ] }
[ DEFERRABLE | NOT DEFERRABLE ] [ INITIALLY DEFERRED | INITIALLY
 IMMEDIATE ]


###table_constraint#############

[ CONSTRAINT constraint_name ]
{ CHECK ( expression ) [ NO INHERIT ] |
  UNIQUE [ NULLS [ NOT ] DISTINCT ] ( column_name
 [, ... ] ) index_parameters |
  PRIMARY KEY ( column_name [, ... ] ) index_parameters |
  EXCLUDE [ USING index_method ] ( exclude_element WITH operator
 [, ... ] ) index_parameters [ WHERE ( predicate ) ] |
  FOREIGN KEY ( column_name [, ... ] ) REFERENCES reftable [ ( refcolumn
 [, ... ] ) ]
    [ MATCH FULL | MATCH PARTIAL | MATCH SIMPLE ] [ ON
 DELETE referential_action ] [ ON UPDATE referential_action ] }
[ DEFERRABLE | NOT DEFERRABLE ] [ INITIALLY DEFERRED | INITIALLY
 IMMEDIATE ]


##like_option##########

{ INCLUDING | EXCLUDING } { COMMENTS | COMPRESSION | CONSTRAINTS |
 DEFAULTS | GENERATED | IDENTITY | INDEXES | STATISTICS | STORAGE | ALL }


##partition_bound_spec##########

IN ( partition_bound_expr [, ...] ) |
FROM ( { partition_bound_expr | MINVALUE | MAXVALUE } [, ...] )
  TO ( { partition_bound_expr | MINVALUE | MAXVALUE } [, ...] ) |
WITH ( MODULUS numeric_literal, REMAINDER numeric_literal )


UNIQUE#PRIMARY KEY###EXCLUDE####index_parameters#########

[ INCLUDE ( column_name [, ... ] ) ]
[ WITH ( storage_parameter [= value] [, ... ] ) ]
[ USING INDEX TABLESPACE tablespace_name ]


EXCLUDE####exclude_element#########

{ column_name | ( expression ) } [ COLLATE collation ] [ opclass
 [ ( opclass_parameter = value [, ... ] ) ] ] [ ASC | DESC ] [ NULLS
 { FIRST | LAST } ]


FOREIGN KEY/REFERENCES####referential_action#########

{ NO ACTION | RESTRICT | CASCADE | SET NULL [ ( column_name [, ... ] ) ] |
 SET DEFAULT [ ( column_name [, ... ] ) ] }
```

##

```
CREATE TABLE########################### ##############################
```

##################CREATE TABLE myschema.mytable ...################### ### ############################## ## ######### ######################(######################################)###### ###########

####CREATE TABLE##########1################### ############################ ##############

##################SQL########

###################2######## ################### ############################ ############# ############################## #####1########################### ###

#######################OF#######USAGE##############

#####

TEMPORARY###TEMP

> ################################### ##################################### ########ON COMMIT############### ######search_path######################### ## ###################### #############
>
> ### ##########SQL##### ########################### ## ##ANALYZE############
>
> #######GLOBAL###LOCAL#TEMPORARY#TEMP############## PostgreSQL########## ####### ############

UNLOGGED

> ################################## ################################28######## ####### ###################### ######################## ################### ################## ################################## ###################### ########################
>
> #########(ID##########)##

IF NOT EXISTS

> ############################## ############## ############################## ###########

table_name

> ##########################

OF *type_name*

> #######(########)####################### ################## ######(DROP TYPE ... CASCADE#)##################
>
> ####################################CREATE TABLE############## ###CREATE TABLE########################## ##################

column_name

> ###################

data_type

######### ######################## PostgreSQL#######################8###########

COLLATE *collation*

COLLATE## ##########################
#####

STORAGE { PLAIN | EXTERNAL | EXTENDED | MAIN | DEFAULT }

################### ########################TOAST########################
############ PLAIN##integer################################ MAIN#####
################### EXTERNAL##############EXTENDED####################
DEFAULT#################################### EXTENDED##PLAIN#############
################### EXTERNAL###########text###bytea####################
############################## ####65.2##########

COMPRESSION *compression_method*

COMPRESSION##main##
#extended########### (##############ALTER TABLE#########) ##############
##
#############pglz#lz4### (lz4##PostgreSQL####--with-lz4############
####) ###*compression_method*#####################default################
######################default_toast_compression########

INHERITS (*parent_table* [, ...])

######INHERITS## ############
###################

INHERITS## ####################
##

####################
#####1############# ##
################1######### ##
######## ## ####
############

CHECK######################### ################################CHECK######
################################## ################# ##########################
######NO INHERIT################ ###########CHECK######################
###############

##STORAGE###################

##########IDENTITY################# ###########IDENTITY#############

PARTITION BY { RANGE | LIST | HASH } ({ *column_name* | (*expression*) } [*opclass*] [, ...])

######PARTITION BY############################ ##########################
####### ################################### ################################
################32########PostgreSQL#####################################
#1######################## ################B################################B
################ B#########################

#################B## ###########
################
######################2###################36.16.3#####

##CREATE TABLE############## #### ################## ######################################1############### ####################################

############EXCLUDE##

#########################5.12#########

PARTITION OF *parent_table* { FOR VALUES *partition_bound_spec* | DEFAULT }

############################## FOR VALUES#####################
##DEFAULT############################## ################################
###############

partition_bound_spec###
######################## IN#################FROM#TO################WITH#
########################

partition_bound_expr## #####
############################## ####################CURRENT_TIMESTAMP####
##############

################NULL#####################################NULL##########
########## ####1#######2############################## ###########NULL#####
#########

###############FROM###############TO#################### ####FROM######
###TO######################## ######
###########9.25.5########################## ####PARTITION BY RANGE (x,y)#
############FROM (1, 2) TO (3, 4)###x=1#y>=2########x=2#NULL#####y#
###x=3#y<4#############

###############MINVALUE###MAXVALUE#####################################
######## ####FROM (MINVALUE) TO (10)##############10##############FROM
(10) TO (MAXVALUE)##############10#############

2#####################MAXVALUE##############MINVALUE################
####FROM (0, MAXVALUE) TO (10, MAXVALUE)##################
#######1##0#######10############ ####FROM ('a', MINVALUE) TO ('b',
MINVALUE)#########################1##"a"##############

MINVALUE###MAXVALUE##########1###
####(10, MINVALUE, 0)############ (10, MINVALUE, MINVALUE)#####

timestamp##########"infinity"################################
MINVALUE#MAXVALUE###
MAXVALUE#"infinity"############################MINVALUE#"minus infinity"#####
######################## ######FROM ('infinity') TO (MAXVALUE)#######
#####1#######"infinity"########

DEFAULT### ########################
######### ##

########DEFAULT###
################################
######## ##
########

#################################### ####################################
##
######## ##

#################### ## #########8### 8#####################################16#################### ####8######### ################16###8######### ## ##############8################ ################ ## ##############

CHECK###CHECK###### ######### ################################ ########################## ########

##############################

TRUNCATE## ######################

PARTITION OF#####################################ACCESS EXCLUSIVE## ###################### ####DROP TABLE#########################ACCESS EXCLUSIVE################ ### ####ALTER TABLE ATTACH/DETACH PARTITION#######

LIKE *source_table* [*like_option* ...]

LIKE#####################################NULL##################

INHERITS################################## ############################## ##############################

INHERITS#####LIKE################################ ################LIKE## ##################

######*like_option*########################## INCLUDING######## ##EXCLUDING########## EXCLUDING######## ############################ ## ##############

INCLUDING COMMENTS

#################################### ################################## ##################

INCLUDING COMPRESSION

############## ##

INCLUDING CONSTRAINTS

CHECK########## ################### #NULL####################

INCLUDING DEFAULTS

########################### ######################################NULL ########### nextval## #########################

INCLUDING GENERATED

######################### ######################

INCLUDING IDENTITY

############################### ## ##############

INCLUDING INDEXES

################PRIMARY KEY#UNIQUE#EXCLUDE################# ###### ## ################## ##################

INCLUDING STATISTICS

#####################

INCLUDING STORAGE

############STORAGE########## ###########STORAGE################## ########################## STORAGE########65.2#########

INCLUDING ALL

INCLUDING ALL######################## ######################## #EXCLUDING##INCLUDING ALL################

##LIKE## ################ #INCLUDING INDEXES##########

CONSTRAINT *constraint_name*

############(####)### ########################## #####col must be positive(###########)## #### ########################## #####################

NOT NULL

####NULL##############

NULL

####NULL############# ##########

#########SQL######################### ############################

CHECK (*expression*) [NO INHERIT]

CHECK## TRUE###UNKNOWN############# ###########FALSE######################## ######################### ################################## ############## #######

######CHECK################################5.5.1##### #####tableoid###### #####################

NO INHERIT#####################

########CHECK###########NOT NULL#################################### #PostgreSQL#9.5###########CHECK#######################

DEFAULT *default_expr*

DEFAULT################################ ################################ ############## ############## ################################

####################NULL######

GENERATED ALWAYS AS (*generation_expr*) STORED

#################### ##################################

#####STORED##

#################################### ##############IMMUTABLE########### # ####################

GENERATED { ALWAYS | BY DEFAULT } AS IDENTITY [(*sequence_options*)]

######*IDENTITY*######### ## ##### ###########NOT NULL###

ALWAYS#BY DEFAULT####INSERT#UPDATE################################

INSERT#######ALWAYS##################INSERT##OVERRIDING SYSTEM VALUE#################### BY DEFAULT######################## ###INSERT## ######## (COPY##################################)

UPDATE###ALWAYS#########DEFAULT################## BY DEFAULT############ ######### (UPDATE#####OVERRIDING########)

######*sequence_options*############################## ####CREATE SEQUENCE##########

UNIQUE [NULLS [NOT] DISTINCT] (###)
UNIQUE [NULLS [NOT] DISTINCT] (*column_name* [, ...]) [INCLUDE (*column_name* [, ...])] (######)

UNIQUE########1## ################ ########################### ########2############1##############

########NULLS NOT DISTINCT##########NULL################

######################

###
#######################

##Btree################## ########## ####################

#####INCLUDE######################1#################################### ################### ################################## ####INCLUDE######### ########################### ######################DROP COLUMN############# ##################

PRIMARY KEY #####
PRIMARY KEY (*column_name* [, ...]) [INCLUDE (*column_name* [, ...])] (######)

PRIMARY KEY############################NULL############### ############ ############################

###########################

PRIMARY KEY#UNIQUE#NOT NULL################### ######################### ########################### ## ###################

##################PRIMARY KEY############UNIQUE#########

##Btree################## ########
##################

#####INCLUDE#######################1#####################################
################### ################################## ######################
############ #######################DROP COLUMN###########################
###

EXCLUDE [USING *index_method*] (*exclude_element* WITH *operator* [, ...]) *index_parameters* [WHERE (*predicate*)]

EXCLUDE##############2##########(###)####(###)########(###)##################
###TRUE############## ############################UNIQUE################
######### ##################################### ##########2#########(8.8##)###
#########&&#################### ################

##################################### ############################
#*index_method*##############11.10######################
#*exclude_element*################ ######################################/#
################### ########CREATE INDEX##########

#########amgettuple###############(#62###)# ########GIN################ B-tree#Hash## ######################
###################GiST####SP-GiST######

predicate############################## ######################## predicate###
####################

REFERENCES *reftable* [(*refcolumn*)] [MATCH *matchtype*] [ON DELETE *action*] [ON UPDATE *action*] #####
FOREIGN KEY (*column_name* [, ...]) REFERENCES *reftable* [(*refcolumn* [, ...])] [MATCH *matchtype*] [ON DELETE *referential_action*] [ON UPDATE *referential_action*] #####

#################### ################1#####################################
#################### *refcolumn*###########*reftable*########### ######
##*refcolumn*## #
######################
##########REFERENCES################ #################SHARE ROW EXCLUSIVE########## ##

#####3######## MATCH FULL#MATCH PARTIAL#MATCH SIMPLE############### MATCH FULL#########
NULL####################NULL############ #######NULL##################
############### MATCH SIMPLE#########NULL########### #######NULL#######
########################## MATCH PARTIAL############ #######NOT NULL####
################################

ON DELETE##################
################## ###ON UPDATE## ##
#################NO ACTION########
########## ###################

NO ACTION

#################################### ################################
########## #############

RESTRICT

################################## ##############NO ACTION######

CASCADE

##

SET NULL [(*column_name* [, ...])]

#######################NULL####### #########ON DELETE###############
##

SET DEFAULT [(*column_name* [, ...])]

##################################### #########ON DELETE############
########NULL###
###

###

DEFERRABLE
NOT DEFERRABLE

####################### ####################### ############SET
CONSTRAINTS##################################### NOT DEFERRABLE######
###UNIQUE#PRIMARY KEY#EXCLUDE#REFERENCES##########################
NOT NULL### CHECK################ ########ON CONFLICT DO UPDATE###
#INSERT##################################

INITIALLY IMMEDIATE
INITIALLY DEFERRED

##################################### ###INITIALLY IMMEDIATE############
###INITIALLY DEFERRED########################
#SET CONSTRAINTS####################

USING *method*

###TABLE################
#########61###########
########default_table_access_method#########

###

WITH (*storage_parameter* [= *value*] [, ...])

#############
###WITH############OID#######################OIDS=FALSE#########
##OIDS=TRUE#############

WITHOUT OIDS

###WITHOUT OIDS#################### WITH OIDS#######################

ON COMMIT

ON COMMIT### ###3############

PRESERVE ROWS

############################ #############

DELETE ROWS

##################################### ################TRUNCATE######
######################################

DROP

################################## ####################################
####################################

TABLESPACE *tablespace_name*

tablespace_name######################## ###########default_tablespace######
#######temp_tablespaces####### ##
##
#default_tablespace#######

USING INDEX TABLESPACE *tablespace_name*

#######UNIQUE#PRIMARY KEY####EXCLUDE################################
###########default_tablespace##############temp_tablespaces#######

#######

WITH###########UNIQUE#PRIMARY KEY####EXCLUDE############################
##################CREATE INDEX####### ###############################
######################toast########################## ################TOAST
############# (TOAST##########65.2#########) ######################toast.##
##############TOAST###################### ################################
####################################

fillfactor (integer)

############(fillfactor)#10##100################ 100################ ########
####INSERT########################## ############################## ###
##UPDATE#################################### ############################
############################ ##
######################### TOAST####################

toast_tuple_target (integer)

toast_tuple_target###########TOAST##################################TOAST#
################################ ###EXTERNAL(######)#MAIN(######)###
##EXTENDED(######)################################ ############## ########
########1###########4#######################################2040######### ###
##128############ - ##########8160######## ################################
######## ##

parallel_workers (integer)

##################################
##max_worker_processes############
##########

autovacuum_enabled, toast.autovacuum_enabled (boolean)

################################### true################24.1.6################
###########VACUUM####ANALYZE################ false###########ID##########
################### ###########24.1.5######### autovacuum######false########
#####ID## ###################
############ #################true##################false##############
###

vacuum_index_cleanup, toast.vacuum_index_cleanup (enum)

#######VACUUM################################### #######AUTO### OFF#######
################ON#####AUTO######VACUUM###################### #############

#####################################VACUUM############## ###############
###############VACUUM###
VACUUM#INDEX_CLEANUP############################

`vacuum_truncate`, `toast.vacuum_truncate` (`boolean`)

#######true### true####VACUUM###
###########ACCESS
EXCLUSIVE#################### VACUUM#TRUNCATE##########################
###

`autovacuum_vacuum_threshold`, `toast.autovacuum_vacuum_threshold` (`integer`)

autovacuum_vacuum_threshold#######################

`autovacuum_vacuum_scale_factor`, `toast.autovacuum_vacuum_scale_factor` (`floating point`)

autovacuum_vacuum_scale_factor#######################

`autovacuum_vacuum_insert_threshold`, `toast.autovacuum_vacuum_insert_threshold` (`integer`)

autovacuum_vacuum_insert_threshold####################### #######-1#############
##################

`autovacuum_vacuum_insert_scale_factor`, `toast.autovacuum_vacuum_insert_scale_factor` (`floating point`)

autovacuum_vacuum_insert_scale_factor#######################

`autovacuum_analyze_threshold` (`integer`)

autovacuum_analyze_threshold#######################

`autovacuum_analyze_scale_factor` (`floating point`)

autovacuum_analyze_scale_factor#######################

`autovacuum_vacuum_cost_delay`, `toast.autovacuum_vacuum_cost_delay` (`floating point`)

autovacuum_vacuum_cost_delay#######################

`autovacuum_vacuum_cost_limit`, `toast.autovacuum_vacuum_cost_limit` (`integer`)

autovacuum_vacuum_cost_limit#######################

`autovacuum_freeze_min_age`, `toast.autovacuum_freeze_min_age` (`integer`)

vacuum_freeze_min_age####################### ######
#autovacuum_freeze_min_age###########autovacuum_freeze_max_age###1/2###
##############################

`autovacuum_freeze_max_age`, `toast.autovacuum_freeze_max_age` (`integer`)

autovacuum_freeze_max_age####################### ######
#autovacuum_freeze_max_age##
#######################

`autovacuum_freeze_table_age`, `toast.autovacuum_freeze_table_age` (`integer`)

vacuum_freeze_table_age#######################

`autovacuum_multixact_freeze_min_age`, `toast.autovacuum_multixact_freeze_min_age` (`integer`)

vacuum_multixact_freeze_min_age####################### ######
#autovacuum_multixact_freeze_min_age###########
#autovacuum_multixact_freeze_max_age#####################################

`autovacuum_multixact_freeze_max_age`, `toast.autovacuum_multixact_freeze_max_age` (`integer`)

autovacuum_multixact_freeze_max_age####################### ######
#autovacuum_multixact_freeze_max_age################################
############################

`autovacuum_multixact_freeze_table_age`, `toast.autovacuum_multixact_freeze_table_age` (`integer`)

vacuum_multixact_freeze_table_age#######################

`log_autovacuum_min_duration`, `toast.log_autovacuum_min_duration` (`integer`)

log_autovacuum_min_duration#######################

`user_catalog_table` (`boolean`)

##################################### ####47.6.2######### ########TOAST###
###########

##

PostgreSQL## ####################
############## ########CREATE INDEX###########

######################### ###############################

#####1600############### ############################

# #

`films#####distributors##########`

```
CREATE TABLE films (
    code        char(5) CONSTRAINT firstkey PRIMARY KEY,
    title       varchar(40) NOT NULL,
    did         integer NOT NULL,
    date_prod   date,
    kind        varchar(10),
    len         interval hour to minute
);

CREATE TABLE distributors (
     did    integer PRIMARY KEY GENERATED BY DEFAULT AS IDENTITY,
     name   varchar(40) NOT NULL CHECK (name <> '')
);
```

2##################

```
CREATE TABLE array_int (
    vector  int[][]
);
```

films##### ################ ###############1##################

```
CREATE TABLE films (
    code        char(5),
    title       varchar(40),
    did         integer,
    date_prod   date,
    kind        varchar(10),
    len         interval hour to minute,
    CONSTRAINT production UNIQUE(date_prod)
);
```

############

```
CREATE TABLE distributors (
    did     integer CHECK (did > 100),
    name    varchar(40)
);
```

###############

```
CREATE TABLE distributors (
    did     integer,
    name    varchar(40),
    CONSTRAINT con1 CHECK (did > 100 AND name <> '')
);
```

films####################

```
CREATE TABLE films (
    code        char(5),
    title       varchar(40),
    did         integer,
    date_prod   date,
    kind        varchar(10),
    len         interval hour to minute,
    CONSTRAINT code_title PRIMARY KEY(code,title)
);
```

distributors################ ###2#######################################

```
CREATE TABLE distributors (
    did     integer,
    name    varchar(40),
    PRIMARY KEY(did)
);
```

```
CREATE TABLE distributors (
    did     integer PRIMARY KEY,
    name    varchar(40)
);
```

#####name##########################did#######################################
modtime#########################

```
CREATE TABLE distributors (
    name      varchar(40) DEFAULT 'Luso Films',
    did       integer DEFAULT nextval('distributors_serial'),
    modtime   timestamp DEFAULT current_timestamp
);
```

2##NOT NULL####distributors########## ####1################

```
CREATE TABLE distributors (
    did     integer CONSTRAINT no_null NOT NULL,
    name    varchar(40) NOT NULL
);
```

name################

```
CREATE TABLE distributors (
    did     integer,
    name    varchar(40) UNIQUE
);
```

######################

```
CREATE TABLE distributors (
    did     integer,
    name    varchar(40),
    UNIQUE(name)
);
```

###################70%##########################

```
CREATE TABLE distributors (
    did     integer,
    name    varchar(40),
    UNIQUE(name) WITH (fillfactor=70)
)
WITH (fillfactor=70);
```

2#################circles##########

```
CREATE TABLE circles (
    c circle,
    EXCLUDE USING gist (c WITH &&)
);
```

diskvol1#######cinemas##########

```
CREATE TABLE cinemas (
        id serial,
        name text,
        location text
) TABLESPACE diskvol1;
```

#################

```
CREATE TYPE employee_type AS (name text, salary numeric);

CREATE TABLE employees OF employee_type (
    PRIMARY KEY (name),
```

```
    salary WITH OPTIONS DEFAULT 1000
);

###################


CREATE TABLE measurement (
    logdate         date not null,
    peaktemp        int,
    unitsales       int
) PARTITION BY RANGE (logdate);

####################################


CREATE TABLE measurement_year_month (
    logdate         date not null,
    peaktemp        int,
    unitsales       int
) PARTITION BY RANGE (EXTRACT(YEAR FROM logdate), EXTRACT(MONTH FROM
 logdate));

#####################


CREATE TABLE cities (
    city_id      bigserial not null,
    name         text not null,
    population   bigint
) PARTITION BY LIST (left(lower(name), 1));

######################


CREATE TABLE orders (
    order_id     bigint not null,
    cust_id      bigint not null,
    status       text
) PARTITION BY HASH (order_id);

############################


CREATE TABLE measurement_y2016m07
    PARTITION OF measurement (
    unitsales DEFAULT 0
) FOR VALUES FROM ('2016-07-01') TO ('2016-08-01');

##################################################


CREATE TABLE measurement_ym_older
    PARTITION OF measurement_year_month
    FOR VALUES FROM (MINVALUE, MINVALUE) TO (2016, 11);

CREATE TABLE measurement_ym_y2016m11
    PARTITION OF measurement_year_month
    FOR VALUES FROM (2016, 11) TO (2016, 12);

CREATE TABLE measurement_ym_y2016m12
    PARTITION OF measurement_year_month
    FOR VALUES FROM (2016, 12) TO (2017, 01);

CREATE TABLE measurement_ym_y2017m01
    PARTITION OF measurement_year_month
```

```
    FOR VALUES FROM (2017, 01) TO (2017, 02);

###########################


CREATE TABLE cities_ab
    PARTITION OF cities (
    CONSTRAINT city_id_nonzero CHECK (city_id != 0)
) FOR VALUES IN ('a', 'b');

#################################################################


CREATE TABLE cities_ab
    PARTITION OF cities (
    CONSTRAINT city_id_nonzero CHECK (city_id != 0)
) FOR VALUES IN ('a', 'b') PARTITION BY RANGE (population);

CREATE TABLE cities_ab_10000_to_100000
    PARTITION OF cities_ab FOR VALUES FROM (10000) TO (100000);

#############################


CREATE TABLE orders_p1 PARTITION OF orders
    FOR VALUES WITH (MODULUS 4, REMAINDER 0);
CREATE TABLE orders_p2 PARTITION OF orders
    FOR VALUES WITH (MODULUS 4, REMAINDER 1);
CREATE TABLE orders_p3 PARTITION OF orders
    FOR VALUES WITH (MODULUS 4, REMAINDER 2);
CREATE TABLE orders_p4 PARTITION OF orders
    FOR VALUES WITH (MODULUS 4, REMAINDER 3);

###################


CREATE TABLE cities_partdef
    PARTITION OF cities DEFAULT;
```

###

```
CREATE TABLE################SQL######
```

######

```
CREATE  TEMPORARY  TABLE###SQL######################### ####################
#################################### PostgreSQL################################
#CREATE TEMPORARY TABLE################# ################################
########## ##################################################

############################## #####PostgreSQL##########SQL#############

####SQL################################ ##################SQL###############
########################### PostgreSQL#SQL################PostgreSQL##########
#######

###########PostgreSQL#############GLOBAL#LOCAL##############################
## PostgreSQL#################################################################

########ON COMMIT#######SQL#################### ON COMMIT#########SQL####
########ON COMMIT DELETE ROWS########### ####PostgreSQL###########ON COMMIT
PRESERVE ROWS### ###ON COMMIT DROP#SQL#######
```

########

UNIQUE###PRIMARY KEY#########PostgreSQL########################## ##SQL#### ################################ ###########1########################### ###### ################INITIALLY IMMEDIATE#####DEFERRABLE############## ######## ##################################

#####

##SQL###CHECK################################CHECK################## PostgreSQL############ ############################

EXCLUDE##

EXCLUDE########PostgreSQL######

######

########SET DEFAULT#SET NULL#############PostgreSQL######

###PostgreSQL######

NULL

NULL################SQL####PostgreSQL############################## NOT NULL################### ################################

#####

SQL### PostgreSQL############ ##################################### ########################### #UNIQUE#PRIMARY KEY####EXCLUDE## ######################################

############PostgreSQL##NULL################################### ############## #########

##

INHERITS#########PostgreSQL####### SQL:1999############################## ### ###SQL:1999######PostgreSQL#############

##########

PostgreSQL######################## #####CREATE TABLE foo();## #####SQL### ##### ##SQL################### ################################ALTER TABLE DROP COLUMN##################### ################################

###IDENTITY#

PostgreSQL#######2####IDENTITY############## ##SQL###1#########1##IDENTITY ################## ##################################### INSERT##### OVERRIDING##1############################IDENTITY######################## ###############

###

#####STORED############SQL###########SQL##################

LIKE#

LIKE####SQL#######PostgreSQL##################################PostgreSQL#####
#######

WITH#

WITH##PostgreSQL###### #################

######

PostgreSQL#################### ######TABLESPACE#USING INDEX TABLESPACE##PostgreSQL########

#######

##########SQL############# ##
PostgreSQL###################

PARTITION BY#

PARTITION BY#PostgreSQL######

PARTITION OF#

PARTITION OF##PostgreSQL######

####

ALTER TABLE, DROP TABLE, CREATE TABLE AS, CREATE TABLESPACE, CREATE TYPE

CREATE TABLE AS

CREATE TABLE AS — ######################

##

```
CREATE [ [ GLOBAL | LOCAL ] { TEMPORARY | TEMP } | UNLOGGED ] TABLE [ IF
 NOT EXISTS ] table_name
    [ (column_name [, ...] ) ]
    [ USING method ]
    [ WITH ( storage_parameter [= value] [, ... ] ) | WITHOUT OIDS ]
    [ ON COMMIT { PRESERVE ROWS | DELETE ROWS | DROP } ]
    [ TABLESPACE tablespace_name ]
    AS query
    [ WITH [ NO ] DATA ]
```

##

CREATE TABLE AS#########SELECT ############################ ###### ##SELECT##

CREATE TABLE AS########################### CREATE TABLE AS############## ################################## ## ####################SELECT########

CREATE TABLE AS####################CREATE#######

#####

GLOBAL###LOCAL

######################### ################### #######CREATE TABLE######## ##

TEMPORARY###TEMP

############################ #######CREATE TABLE##########

UNLOGGED

################################## #######CREATE TABLE##########

IF NOT EXISTS

############################## ############################

table_name

########################

column_name

################## ###########################

USING *method*

###TABLE################### ### ########61########### ### ### ########default_table_access_method##########

WITH (*storage_parameter* [= *value*] [, ...])

####################(####)####### ###CREATE TABLE################## ## ###############WITH####OID#####################OIDS=FALSE######### ##OIDS=TRUE################

WITHOUT OIDS

#########WITHOUT OIDS#####################WITH OIDS################### ###########

ON COMMIT

##########################ON COMMIT################ ###3############

PRESERVE ROWS

############################# #############

DELETE ROWS

#################################### ################TRUNCATE#######

DROP

################################

TABLESPACE *tablespace_name*

tablespace_name######################## ########default_tablespace######### ##temp_tablespaces#######

query

SELECT#TABLE#VALUES#################SELECT#TABLE###VALUES######## #EXECUTE#######

WITH [NO] DATA

############

##

########SELECT INTO############SELECT INTO################################ ####### ####CREATE TABLE AS##SELECT INTO####################

# #

films#####################films_recent#######

```
CREATE TABLE films_recent AS
  SELECT * FROM films WHERE date_prod >= '2002-01-01';
```

###############TABLE######################

```
CREATE TABLE films2 AS
  TABLE films;
```

###########films#####################films_recent####### ################# #####

```
PREPARE recentfilms(date) AS
  SELECT * FROM films WHERE date_prod > $1;
CREATE TEMP TABLE films_recent ON COMMIT DROP AS
  EXECUTE recentfilms('2002-01-01');
```

###

CREATE TABLE AS###SQL###### ###########

- ##########################PostgreSQL##############
- ####WITH [NO] DATA########PostgreSQL#########
- PostgreSQL#################### ###CREATE TABLE##########
- WITH##PostgreSQL###### #################
- PostgreSQL###################### ######TABLESPACE#######

####

CREATE MATERIALIZED VIEW, CREATE TABLE, EXECUTE, SELECT, SELECT INTO, VALUES

CREATE TABLESPACE

CREATE TABLESPACE — ##############

##

```
CREATE TABLESPACE tablespace_name
    [ OWNER { new_owner | CURRENT_ROLE | CURRENT_USER | SESSION_USER } ]
    LOCATION 'directory'
    [ WITH ( tablespace_option = value [, ... ] ) ]
```

##

CREATE TABLESPACE############################ ##############################
##################

##
##########

#############CREATE DATABASE#CREATE TABLE#CREATE INDEX#ADD CONSTRAINT#
####*tablespace_name*##

> ##
>
> ##################################### 22.6##########

#####

tablespace_name

################ pg_##

user_name

#################### ############################ ############################
################################

directory

#################### ########################(CREATE TABLESPACE ##########
####)#############PostgreSQL######################### ##########################
##

tablespace_option

############################## ##############
#seq_page_cost#random_page_cost#effective_io_concurrency#maintenance_io_concu
##
##############################
(seq_page_cost#random_page_cost#effective_io_concurrency#maintenance_io_concurrency##
#)######### ##########I/O#######################################

##

##############CREATE TABLESPACE##############

# #

```
#########/data/dbs#######dbspace##########################################
##########


mkdir /data/dbs
chown postgres:postgres /data/dbs

##PostgreSQL#####################


CREATE TABLESPACE dbspace LOCATION '/data/dbs';

##############################################


CREATE TABLESPACE indexspace OWNER genevieve LOCATION '/data/indexes';
```

###

CREATE TABLESPACE#PostgreSQL######

####

CREATE DATABASE, CREATE TABLE, CREATE INDEX, DROP TABLESPACE, ALTER TABLESPACE

CREATE TEXT SEARCH CONFIGURATION

CREATE TEXT SEARCH CONFIGURATION — ################

##

```
CREATE TEXT SEARCH CONFIGURATION name (
    PARSER = parser_name |
    COPY = source_config
)
```

##

CREATE TEXT SEARCH CONFIGURATION################## ######################
###

#####
################ALTER TEXT SEARCH CONFIGURATION#########################
############################

########################

#########################

####12###########

#####

name

################## ##################

parser_name

########################

source_config

#################

##

PARSER#COPY##################### ###############################

###

##SQL##CREATE TEXT SEARCH CONFIGURATION########

####

ALTER TEXT SEARCH CONFIGURATION, DROP TEXT SEARCH CONFIGURATION

CREATE TEXT SEARCH DICTIONARY

CREATE TEXT SEARCH DICTIONARY — ###############

##

```
CREATE TEXT SEARCH DICTIONARY name (
    TEMPLATE = template
    [, option = value [, ... ]]
)
```

##

CREATE TEXT SEARCH DICTIONARY################## ######################### ################### ##################################### ################### ####################

#######################

#########################

####12###########

#####

name

################## ##################

template

################################

option

##############################

value

######################## ################################## ###############
###

#######################

#

##

```
CREATE TEXT SEARCH DICTIONARY my_russian (
    template = snowball,
    language = russian,
    stopwords = myrussian
);
```

###

##SQL##CREATE TEXT SEARCH DICTIONARY########

####

ALTER TEXT SEARCH DICTIONARY, DROP TEXT SEARCH DICTIONARY

CREATE TEXT SEARCH PARSER

CREATE TEXT SEARCH PARSER — ################

##

```
CREATE TEXT SEARCH PARSER name (
    START = start_function ,
    GETTOKEN = gettoken_function ,
    END = end_function ,
    LEXTYPES = lextypes_function
    [, HEADLINE = headline_function ]
)
```

##

CREATE TEXT SEARCH PARSER################### ############################## ######################### ## ##############

#######################

CREATE TEXT SEARCH PARSER######################### ######################### ############################

####12###########

#####

name

################### ####################

start_function

##############

gettoken_function

######################

end_function

##############

lextypes_function

####LEXTYPE##############################

headline_function

#############################

#################### ##################################### ###################

##################################

###

##SQL##`CREATE TEXT SEARCH PARSER`########

####

ALTER TEXT SEARCH PARSER, DROP TEXT SEARCH PARSER

CREATE TEXT SEARCH TEMPLATE

CREATE TEXT SEARCH TEMPLATE — ##################

##

```
CREATE TEXT SEARCH TEMPLATE name (
    [ INIT = init_function , ]
    LEXIZE = lexize_function
)
```

##

CREATE TEXT SEARCH TEMPLATE###################### ######################### ############ ## ################## ############

######################

CREATE TEXT SEARCH TEMPLATE######################### ###################### ################################ ## ###### ## ########################

####12##########

#####

name

###################### ####################

init_function

#################

lexize_function

#################

#################### ################################### ##################### ####

##################################

###

##SQL##CREATE TEXT SEARCH TEMPLATE#######

####

ALTER TEXT SEARCH TEMPLATE, DROP TEXT SEARCH TEMPLATE

CREATE TRANSFORM

CREATE TRANSFORM — ##########

##

```
CREATE [ OR REPLACE ] TRANSFORM FOR type_name LANGUAGE lang_name (
    FROM SQL WITH FUNCTION from_sql_function_name [ (argument_type
 [, ...]) ],
    TO SQL WITH FUNCTION to_sql_function_name [ (argument_type [, ...]) ]
);
```

##

CREATE TRANSFORM############ CREATE OR REPLACE TRANSFORM############## ##############

################################ ###hstore#####PL/Python#########PL/Python #hstore###Python########################## ################################## #######################

######2##########

- #from SQL########SQL############## ############################
- #to SQL############SQL########## ############################

######################## ##################################### ################ ##

####################USAGE###########USAGE########from-SQL######to-SQL######### ############EXECUTE############

#####

type_name

#################

lang_name

###############

from_sql_function_name[(*argument_type* [, ...])]

##SQL################## internal#####1####internal######## ########### ######################## ##########1##internal#########internal###SQL### ################## ######################### ################################ ######

to_sql_function_name[(*argument_type* [, ...])]

######SQL############## internal#####1################### ############## ######### ####################################

##

#########DROP TRANSFORM######

# #

hstore####plpython3u##############################

```
CREATE TYPE hstore ...;

CREATE EXTENSION plpython3u;
```

##############

```
CREATE FUNCTION hstore_to_plpython(val internal) RETURNS internal
LANGUAGE C STRICT IMMUTABLE
AS ...;

CREATE FUNCTION plpython_to_hstore(val internal) RETURNS hstore
LANGUAGE C STRICT IMMUTABLE
AS ...;
```

################################

```
CREATE TRANSFORM FOR hstore LANGUAGE plpython3u (
    FROM SQL WITH FUNCTION hstore_to_plpython(internal),
    TO SQL WITH FUNCTION plpython_to_hstore(internal)
);
```

################################

contrib##

###

#####CREATE TRANSFORM#PostgreSQL###### ##SQL##CREATE TRANSFORM############ ############################## ######PostgreSQL##############

####

CREATE FUNCTION, CREATE LANGUAGE, CREATE TYPE, DROP TRANSFORM

CREATE TRIGGER

CREATE TRIGGER — ##########

##

```
CREATE [ OR REPLACE ] [ CONSTRAINT ] TRIGGER name { BEFORE | AFTER |
 INSTEAD OF } { event [ OR ... ] }
    ON table_name
    [ FROM referenced_table_name ]
    [ NOT DEFERRABLE | [ DEFERRABLE ] [ INITIALLY IMMEDIATE | INITIALLY
 DEFERRED ] ]
    [ REFERENCING { { OLD | NEW } TABLE [ AS ] transition_relation_name }
 [ ... ] ]
    [ FOR [ EACH ] { ROW | STATEMENT } ]
    [ WHEN ( condition ) ]
    EXECUTE { FUNCTION | PROCEDURE } function_name ( arguments )
```

###event###################

```
    INSERT
    UPDATE [ OF column_name [, ... ] ]
    DELETE
    TRUNCATE
```

##

CREATE TRIGGER############# CREATE OR REPLACE TRIGGER######## ################### ## *##function_name######*

#####################################CREATE OR REPLACE TRIGGER######### ## ################

###INSERT#UPDATE###DELETE############## ##############INSERT#UPDATE###DELETE## ########### ##INSERT#UPDATE##### ###################### ###

FOR EACH ROW#######################1########### ####10#######DELETE#### #############ON DELETE###############1######10############ ####FOR EACH STATEMENT#####################################1######## ######0############ #####FOR EACH STATEMENT###########

########INSTEAD OF###################FOR EACH ROW############# ########### ######## #######BEFORE###AFTER####FOR EACH STATEMENT#############

####FOR EACH STATEMENT#########TRUNCATE#####################

##

##	####	####	####
BEFORE	INSERT/ UPDATE/DELETE	############	################

##	####	####	####
	TRUNCATE	—	############
AFTER	INSERT/ UPDATE/DELETE	############	################
	TRUNCATE	—	############
INSTEAD OF	INSERT/ UPDATE/DELETE	###	—
	TRUNCATE	—	—

#############WHEN## ##########WHEN#####
############################## #########WHEN################################
###########################

##

CONSTRAINT############################## ####SET CONSTRAINTS###########
################################### ########################AFTER ROW######
####### ### ##############
SET CONSTRAINTS## ####################
#################

REFERENCING####################### ############SQL#######################
###########SQL###############1############################
#########AFTER################## ######UPDATE######*column_name*##########
OLD TABLE##############UPDATE###DELETE####################
####NEW TABLE##############UPDATE##
#INSERT#################### ##

SELECT##############SELECT################ SELECT#######################
##############

#################37##########

#####

name

################ ############################ ################### ##########
############ #############SET CONSTRAINTS#######################

BEFORE
AFTER
INSTEAD OF

#######AFTER#################

event

INSERT#UPDATE#DELETE#TRUNCATE########## ########################### ###
################OR#########################

UPDATE################################

UPDATE OF *column_name1* [, *column_name2* ...]

#######UPDATE#########################1######################1##UPDATE#
#########################

INSTEAD OF UPDATE################## ################################

table_name

##

referenced_table_name

############################ ##################################### #########
#############

DEFERRABLE
NOT DEFERRABLE
INITIALLY IMMEDIATE
INITIALLY DEFERRED

################# ###############CREATE TABLE########### ##############
#######

REFERENCING

################################1####2####################

OLD TABLE
NEW TABLE

###

transition_relation_name

#######################################

FOR EACH ROW
FOR EACH STATEMENT

####################################1######SQL####1############## ######
######FOR EACH STATEMENT######## ######FOR EACH ROW#############

condition

########################## WHEN########*condition*#true#################
FOR EACH ROW#####WHEN#######OLD.*column_name*#NEW.*column_name*##
#####INSERT#####OLD############
##DELETE#####NEW#############

INSTEAD OF####WHEN###########

#####WHEN####################

#######WHEN## ####################
################

function_name

#########################trigger############################

CREATE TRIGGER##########FUNCTION#PROCEDURE##########################
######################## #######PROCEDURE####################

arguments

#################################### ######################## ##############
################## ### #
##################

##

####################################TRIGGER############### #############
#EXECUTE##############

###########DROP TRIGGER#########

###
################# ############################CREATE OR REPLACE TRIGGER#
################################## ##################################

#######UPDATE OF *column_name*#########################UPDATE#####SET#####
################## BEFORE UPDATE##
################ ####UPDATE ... SET x = x ...######################x#####
###########

BEFORE#######WHEN############################ ####WHEN####################
################ #########NEW## #
#BEFORE####WHEN#####NEW#######(ctid##)#########################

AFTER#######WHEN### ###
#AFTER####WHEN## ###########
##

#######1##SQL#####2#################### ####ON CONFLICT DO UPDATE###
#INSERT## ####################
############### ####INSERT##################UPDATE##################

ON UPDATE CASCADE#ON DELETE SET NULL####################
############SQL## #############
################SQL################################### ####################
######1##SQL## ###############
#AFTER ROW############SQL##
################## ####################1###################1#################
#####################################

#################################INSTEAD OF######################### ########
##INSTEAD OF## #############
#################################### ##
################## ##############################

###
############################## ##
################### REFERENCING##
##############################
###############

#################
#INSTEAD OF###########

#######OR REPLACE##########################

####################
##

###9.29##########

# #

####accounts############check_account_update######

```
CREATE TRIGGER check_update
    BEFORE UPDATE ON accounts
    FOR EACH ROW
    EXECUTE FUNCTION check_account_update();
```

UPDATE#####balance##

```
CREATE OR REPLACE TRIGGER check_update
    BEFORE UPDATE OF balance ON accounts
    FOR EACH ROW
    EXECUTE FUNCTION check_account_update();
```

#########balance######################

```
CREATE TRIGGER check_update
    BEFORE UPDATE ON accounts
    FOR EACH ROW
    WHEN (OLD.balance IS DISTINCT FROM NEW.balance)
    EXECUTE FUNCTION check_account_update();
```

############accounts##################

```
CREATE TRIGGER log_update
    AFTER UPDATE ON accounts
    FOR EACH ROW
    WHEN (OLD.* IS DISTINCT FROM NEW.*)
    EXECUTE FUNCTION log_account_update();
```

################################view_insert_row#######

```
CREATE TRIGGER view_insert
    INSTEAD OF INSERT ON my_view
    FOR EACH ROW
    EXECUTE FUNCTION view_insert_row();
```

########check_transfer_balances_to_zero######transfer##################
##

```
CREATE TRIGGER transfer_insert
    AFTER INSERT ON transfer
    REFERENCING NEW TABLE AS inserted
    FOR EACH STATEMENT
    EXECUTE FUNCTION check_transfer_balances_to_zero();
```

########check_matching_pairs##

```
CREATE TRIGGER paired_items_update
    AFTER UPDATE ON paired_items
    REFERENCING NEW TABLE AS newtab OLD TABLE AS oldtab
    FOR EACH ROW
    EXECUTE FUNCTION check_matching_pairs();
```

37.4###C#######################

###

PostgreSQL####CREATE TRIGGER####SQL############## ####PostgreSQL############
##

- AFTER###########REFERENCING######SQL###########FOR EACH ROW#########
  ###REFERENCING############## ##
  ### ######REFERENCING##OLD ROW AS OLD NEW ROW AS NEW################

- ##SQL#######UPDATE##
  ##### #########PostgreSQL###########

- PostgreSQL############################## ##########SQL################## ####
  #######CREATE TABLE############# ##
  ######

SQL################################ PostgreSQL#########################

SQL#############BEFORE DELETE######DELETE#################### PostgreSQL#
##BEFORE DELETE################################ #################### #######
############################BEFORE###################################### ####
##

OR##########################SQL####PostgreSQL######

TRUNCATE####################################SQL####PostgreSQL######

CREATE CONSTRAINT TRIGGER###SQL####PostgreSQL###### OR REPLACE##########

####

ALTER TRIGGER, DROP TRIGGER, CREATE FUNCTION, SET CONSTRAINTS

CREATE TYPE

CREATE TYPE — ############

##

```
CREATE TYPE name AS
    ( [ attribute_name data_type [ COLLATE collation ] [, ... ] ] )

CREATE TYPE name AS ENUM
    ( [ 'label' [, ... ] ] )

CREATE TYPE name AS RANGE (
    SUBTYPE = subtype
    [ , SUBTYPE_OPCLASS = subtype_operator_class ]
    [ , COLLATION = collation ]
    [ , CANONICAL = canonical_function ]
    [ , SUBTYPE_DIFF = subtype_diff_function ]
    [ , MULTIRANGE_TYPE_NAME = multirange_type_name ]
)

CREATE TYPE name (
    INPUT = input_function,
    OUTPUT = output_function
    [ , RECEIVE = receive_function ]
    [ , SEND = send_function ]
    [ , TYPMOD_IN = type_modifier_input_function ]
    [ , TYPMOD_OUT = type_modifier_output_function ]
    [ , ANALYZE = analyze_function ]
    [ , SUBSCRIPT = subscript_function ]
    [ , INTERNALLENGTH = { internallength | VARIABLE } ]
    [ , PASSEDBYVALUE ]
    [ , ALIGNMENT = alignment ]
    [ , STORAGE = storage ]
    [ , LIKE = like_type ]
    [ , CATEGORY = category ]
    [ , PREFERRED = preferred ]
    [ , DEFAULT = default ]
    [ , ELEMENT = element ]
    [ , DELIMITER = delimiter ]
    [ , COLLATABLE = collatable ]
)

CREATE TYPE name
```

##

CREATE TYPE############################## ####################

################################## ############################## ############## ############################ ## #########

############CREATE TYPE############ ###########, ###################### #### ###################### ##CREATE TYPE# ############# ##

###

CREATE TYPE######################### ############################## ############## ######################### ##################### ########################CREATE TYPE############################### ##################################

#######################USAGE##############

###

CREATE TYPE#2###########8.7########enum####### ######################## ## ##NAMEDATALEN#PostgreSQL########64######################### (############### ##ALTER TYPE############1############################)

###

CREATE TYPE########8.17#################

####*subtype*#######################b-tree####################### ############## b-tree####################### ######################*subtype_opclass######### ###### ##*collation*############### ########

canonical##(####)################################## ######################### ##### #######8.17.8######### *canonical*## ############## ### ########### #####CREATE TYPE *name*############## ################################### ### ######################## ############################

subtype_diff##(####)##*subtype*############################double precision############## ################################GiST################ ######## #######8.17.8#########

multirange_type_name#####(####)#################### ######################## ###### ###########range################################range#multirange#### ########### ########################_multirange##############

###

CREATE TYPE################################## ############################## ## ##

#########CREATE FUNCTION# ####2#################### ########*input_function#output_function##### receive_function###send_function###type_modifier_input_function# ##type_modifier_output_function###analyze_function##### #subscript_function####### ##########C##########################

input_function## *output_function*############## #####1##cstring####### ##cstring##oid##integer####3################ #######C############2######## OID###############OID######3##############typmod##### #######-1####### ##### ######################## #######STRICT############### #######NULL############ ##NULL###################### ########################NULL############ ###### ##NULL######################## #### ############1################## ######cstring############# #####NULL####### ###########

receive_function## ################### ################ ## #############

####################
################### #####internal####1#####internal##oid#integer##3#####
########### ####################StringInfo################################
#######STRICT############### #######NULL########
#####NULL##################### ######################NULL############ ####
##NULL######################## ##
##*send_function*################################### ######################
############ ################1################### #####bytea############# ##
###NULL#############

###
########### ######################### ############################ #######
#CREATE TYPE *name*######################## ####C##########################
###########CREATE TYPE##
######

type_modifier_input_function#type_modifier_output_function#########
#################### #####char(5)#numeric(30,2)####################
PostgreSQL#########1############################ ######################0##
###1################### *type_modifier_input_function*###cstring########
########### ##################################### ###########typmod########
##0###integer##1############ ##*type_modifier_input_function*############
type_modifier_output_function#######typmod##################
##############cstring############# ####numeric######(30,2)########## #####
###########typmod#####################*type_modifier_output_function*####
########

######*analyze_function*############################## ##########B-tree######
#####ANALYZE## ##################
#######internal####1########
#boolean################# ######API#####src/include/commands/vacuum.h###
###

######*subscript_function*##SQL############################## ############
#####################################ARRAY[]################ ##############
##*subscript_function*########## ####
#internal####1################(##)############internal###################
#######API#src/include/nodes/subscripting.h###### src/backend/utils/
adt/arraysubs.c########contrib/hstore/hstore_subs.c##################
#################

##############################I/O##################################PostgreSQL#
######################## ############*internallength*### ######
##*internallength*##########################*internallength*#VARIABLE#######
############ #########typlen#-1############### ######################4#######
############## ##############65.2###
#######

######PASSEDBYVALUE################################## ###################
########Datum######4##########8######################

alignment############################## ########1#2#4#8########### ##########4
######################### ########int4###############

storage## #######plain##########
plain################################## extended#########################
#################################### external##############################
################ main## ################
############################extended#external########################
###

plain#######*storage*###############65.2###36.13.1##########*toast*###############
###############TOAST#####################TOAST##########
#ALTER TABLE SET STORAGE###############

like_type###
internallength#passedbyvalue#alignment#storage################ #######
########LIKE################################## ##############################
#####################

category#preferred###
#########ASCII#####################################preferred######## #########
###
###10########## ##
##################################1#######################1####2##########
############################ ####################################*category*##
########## ################################## #######################ASCII######
######

##################NULL########################### #######DEFAULT##########
##################################DEFAULT##################

#######################ELEMENT######################### ####4######int4#######
#####ELEMENT = int4######### ####################

################################*delimiter*################## ###############
#','#### ##

####*collatable*#####(####)######COLLATE################################### #
####################################
####

###

#############PostgreSQL################### ##########################
#NAMEDATALEN######################### ##
########### ############################array_in#array_out###### ########
###############ARRAY[]############### ##
#################

##########################ELEMENT############################## ELEMENT####
################################## ##
####point###2
###################point[0]###point[1]################# ##################
#################################### ##
##########0###############1#######

SUBSCRIPT## #########
#########SUBSCRIPT#####raw_array_subscript_handler################
#SUBSCRIPT#####ELEMENT#########################

#####SUBSCRIPT###########SUBSCRIPT####################typelem#########
##ELEMENT################# ELEMENT###
###

#####

name

#########################

attribute_name

##############

data_type

######################

collation

################################

label

####1##################################

subtype

###########################

subtype_operator_class

#####b-tree###########

canonical_function

##############

subtype_diff_function

#################

multirange_type_name

##############

input_function

#######################################

output_function

#######################################

receive_function

#######################################

send_function

#######################################

type_modifier_input_function

###########################

type_modifier_output_function

##############################

analyze_function

########################

subscript_function

##############################

internallength

########################## ####################

alignment

############# ###############char#int2#int4#double############### ##### #int4###

storage

########### ###############plain#external#extended#main############## ###### plain###

like_type

##########################
internallength#passedbyvalue#alignment#storage######CREATE TYPE### ##################################

category

###############ASCII###### ################'U'### ############ 51.65###### # ###############ASCII##############

preferred

################################ ########## ################################ ######################

default

############### #############NULL###

element

##########################

delimiter

################################

collatable

#################################### ##########

##

###PUBLIC############### ### ############################## ## ################

PostgreSQL#####8.3##_############## ################1############### ## ## ############################## ### ##pg_type.typarray##########################

################################ ## ##### ##

PostgreSQL#####8.2#################CREATE TYPE *name*######### ######### ###################### #######PostgreSQL############################## ######## ################################### ## ## ##C################ ############

PostgreSQL#####16################ereport()##############
#errsave()/ereturn()#################################### ########src/backend/utils/fmgr/README#########

##

```
##########################


CREATE TYPE compfoo AS (f1 int, f2 text);

CREATE FUNCTION getfoo() RETURNS SETOF compfoo AS $$
    SELECT fooid, fooname FROM foo
$$ LANGUAGE SQL;

#############################


CREATE TYPE bug_status AS ENUM ('new', 'open', 'closed');

CREATE TABLE bug (
    id serial,
    description text,
    status bug_status
);

###############


CREATE TYPE float8_range AS RANGE (subtype = float8, subtype_diff =
 float8mi);

############box#########################


CREATE TYPE box;

CREATE FUNCTION my_box_in_function(cstring) RETURNS box AS ... ;
CREATE FUNCTION my_box_out_function(box) RETURNS cstring AS ... ;

CREATE TYPE box (
    INTERNALLENGTH = 16,
    INPUT = my_box_in_function,
    OUTPUT = my_box_out_function
);

CREATE TABLE myboxes (
    id integer,
    description box
);
```

box#######float4##4#########################

```
CREATE TYPE box (
    INTERNALLENGTH = 16,
    INPUT = my_box_in_function,
    OUTPUT = my_box_out_function,
```

```
    ELEMENT = float4
);
```

#########box################### ###################

```
#####################################
```

```
CREATE TYPE bigobj (
    INPUT = lo_filein, OUTPUT = lo_fileout,
    INTERNALLENGTH = VARIABLE
);
CREATE TABLE big_objs (
    id integer,
    obj bigobj
);
```

#######36.13####################################

###

############CREATE TYPE##########SQL###### #####PostgreSQL###### ##SQL#####
#CREATE TYPE############PostgreSQL############

###########################PostgreSQL######CREATE TABLE##########

####

ALTER TYPE, CREATE DOMAIN, CREATE FUNCTION, DROP TYPE

CREATE USER

CREATE USER — ################

##

```
CREATE USER name [ [ WITH ] option [ ... ] ]
```

```
###option#########

      SUPERUSER | NOSUPERUSER
    | CREATEDB | NOCREATEDB
    | CREATEROLE | NOCREATEROLE
    | INHERIT | NOINHERIT
    | LOGIN | NOLOGIN
    | REPLICATION | NOREPLICATION
    | BYPASSRLS | NOBYPASSRLS
    | CONNECTION LIMIT connlimit
    | [ ENCRYPTED ] PASSWORD 'password' | PASSWORD NULL
    | VALID UNTIL 'timestamp'
    | IN ROLE role_name [, ...]
    | IN GROUP role_name [, ...]
    | ROLE role_name [, ...]
    | ADMIN role_name [, ...]
    | USER role_name [, ...]
    | SYSID uid
```

##

CREATE USER#CREATE ROLE######### #######CREATE USER#################### ###LOGIN####CREATE ROLE#######################NOLOGIN#######

###

CREATE USER###PostgreSQL###### ##SQL####################

####

CREATE ROLE

CREATE USER MAPPING

CREATE USER MAPPING — ##################

##

```
CREATE USER MAPPING [ IF NOT EXISTS ] FOR { user_name | USER |
 CURRENT_ROLE | CURRENT_USER | PUBLIC }
    SERVER server_name
    [ OPTIONS ( option 'value' [ , ... ] ) ]
```

##

CREATE USER MAPPING#################### ################################## ##

########USAGE##################

#####

IF NOT EXISTS

   ### ################### ############ ##

user_name

   ###################### CURRENT_ROLE#CURRENT_USER#USER################ PUBLIC###

server_name

   ########################

OPTIONS (*option* '*value*' [, ...])

   ###################### ################################## ################## ## ####################################

#

###bob####foo##############

```
CREATE USER MAPPING FOR bob SERVER foo OPTIONS (user 'bob', password
 'secret');
```

###

CREATE USER MAPPING#ISO/IEC 9075-9 (SQL/MED)######

####

ALTER USER MAPPING, DROP USER MAPPING, CREATE FOREIGN DATA WRAPPER, CREATE SERVER

CREATE VIEW

CREATE VIEW — ##########

##

```
CREATE [ OR REPLACE ] [ TEMP | TEMPORARY ] [ RECURSIVE ] VIEW name
 [ ( column_name [, ...] ) ]
    [ WITH ( view_option_name [= view_option_value] [, ... ] ) ]
    AS query
    [ WITH [ CASCADED | LOCAL ] CHECK OPTION ]
```

##

CREATE VIEW################## ######################### ################### ####################

CREATE OR REPLACE VIEW########### ###################################### ## ##############################(###################)############## ########### ############## ############################

##################CREATE VIEW myschema.myview ...###################### ############################## ################### ######################### ######### ######################(######################################)###### ###########

#####

TEMPORARY###TEMP

- ############################## ############################## ############## ##

- ##############################TEMPORARY##############################

RECURSIVE

- ############

```
CREATE RECURSIVE VIEW [ schema . ] view_name (column_names) AS
 SELECT ...;
```

- ######

```
CREATE VIEW [ schema . ] view_name AS WITH RECURSIVE view_name
 (column_names) AS (SELECT ...) SELECT column_names FROM view_name;
```

- ###### ############################

name

- #######################

column_name

- ##

WITH (*view_option_name* [= *view_option_value*] [, ...])

##

check_option (enum)

#######local#cascaded######WITH [CASCADED | LOCAL] CHECK OPTION##########(#####)#

security_barrier (boolean)

##39.5##########

security_invoker (boolean)

##

#############ALTER VIEW##################

query

############SELECT###VALUES#######

WITH [CASCADED | LOCAL] CHECK OPTION

#####[82.794118]###INSERT###UPDATE###)# ##################### CHECK OPTION################INSERT#####UPDATE################################## ###check option########## ##INSERT#UPDATE#MERGE##

##

##########DROP VIEW##########

################################## #################

```
CREATE VIEW vista AS SELECT 'Hello World';
```

##############?column?############## ###############text#################
####### ##

```
CREATE VIEW vista AS SELECT text 'Hello World' AS hello;
```

####################
######################## ####################################39.5##########

####security_invoker###true##
###################### ##
#########

##############
######security_invoker###
####

##
#######
#security_invoker#true###
########################

##############################
#################### ###########SECURITY INVOKER###SECURITY DEFINER####
##############CURRENT_USER#
################################ ######security_invoker########### ####
##security_invoker#false###########SECURITY DEFINER####################
#######

##USAGE#########
##########################
###USAGE###USAGE###########

CREATE OR REPLACE VIEW#########################SELECT####WITH (...)##
####CHECK OPTION######### #######SELECT########################### ########
#####################################

#######

#####[85.106383]################### ############INSERT#UPDATE#DELETE#######
##################### ################################ #######################
####INSERT#UPDATE#DELETE#MERGE############################### ##############
##############

- ####FROM###
- ################WITH#DISTINCT#GROUP BY#HAVING#LIMIT#OFFSET##########
- ######################UNION#INTERSECT#EXCEPT###########
- ##

```
#####[87.435897]######################################### #######################
########## ###############INSERT####UPDATE########################### ########
#################################### ########################################
## #################INSERT#UPDATE####MERGE######################

#####[86.015831]##########################INSERT#UPDATE###DELETE###########
################ ON CONFLICT UPDATE####INSERT############# ##############
###########INSERT#UPDATE#DELETE####MERGE############################## ON
CONFLICT UPDATE####INSERT###############

#####[83.365759]##########WHERE#######   ####################UPDATE#DELETE#
###################   ###UPDATE#########WHERE############################
#########   ###INSERT#####WHERE##########################################
########ON         CONFLICT         UPDATE##############################
####    CHECK         OPTION#INSERT#UPDATE##############################
## ################WHERE#########################UPDATE#DELETE#MERGE######
####################  ####UPDATE###MERGE##WHERE############################
############### ####INSERT###MERGE######WHERE############################
###############ON CONFLICT UPDATE################################### CHECK
OPTION##INSERT#UPDATE#MERGE#####################################

#########security_barrier###############WHERE##(###LEAKPROOF###########
##)############################## ###39.5######### #########(#####WHERE#######
###)################################ EXPLAIN#######################(##########
###)######################

#####[70.884354]###################################### #######################
### ######INSTEAD   OF############################## ####################
################################ ######CREATE TRIGGER######### #########
##CREATE RULE############################################## #################
############################ ##########INSERT#UPDATE#DELETE#MERGE########
######INSTEAD OF################################ ######################### #
################# ######### ###CREATE TRIGGER######### ####################
##CREATE RULE############################### ###MERGE######################
#############

################################################################ ###########
#############################################################################
#######39.5#### #######security_invoker#true##############################
##################################
```

# #

```
##########Comedy films##############


CREATE VIEW comedies AS
    SELECT *
    FROM films
    WHERE kind = 'Comedy';

##############film##################### ###########*##########################
###########

LOCAL CHECK OPTION#############


CREATE VIEW universal_comedies AS
    SELECT *
    FROM comedies
```

```
    WHERE classification = 'U'
    WITH LOCAL CHECK OPTION;
```

###comedies##############kind = 'Comedy'##classification = 'U'#######
#########INSERT#UPDATE##classification = 'U'###############kind###
######

CASCADED CHECK OPTION###########

```
CREATE VIEW pg_comedies AS
    SELECT *
    FROM comedies
    WHERE classification = 'PG'
    WITH CASCADED CHECK OPTION;
```

###########kind#classification#################

############################

```
CREATE VIEW comedies AS
    SELECT f.*,
           country_code_to_name(f.country_code) AS country,
           (SELECT avg(r.rating)
            FROM user_ratings r
            WHERE r.film_id = f.id) AS avg_rating
    FROM films f
    WHERE f.kind = 'Comedy';
```

######INSERT#UPDATE#DELETE######## films#########################
#country#avg_rating########

1##100####################

```
CREATE RECURSIVE VIEW public.nums_1_100 (n) AS
    VALUES (1)
UNION ALL
    SELECT n+1 FROM nums_1_100 WHERE n < 100;
```

###CREATE###
#####CTE##################

###

CREATE OR REPLACE VIEW#PostgreSQL####### ################# WITH (...)#
####################################

####

ALTER VIEW, DROP VIEW, CREATE MATERIALIZED VIEW

DEALLOCATE

DEALLOCATE — ################

##

```
DEALLOCATE [ PREPARE ] { name | ALL }
```

##

DEALLOCATE##############SQL############# ################################### ###########

################PREPARE##########

#####

PREPARE

###############

name

######################

ALL

###################

###

DEALLOCATE####SQL############SQL##################

####

EXECUTE, PREPARE

DECLARE

DECLARE — ########

##

```
DECLARE name [ BINARY ] [ ASENSITIVE | INSENSITIVE ] [ [ NO ] SCROLL ]
    CURSOR [ { WITH | WITHOUT } HOLD ] FOR query
```

##

DECLARE###################### ################################ ######### ##FETCH#############

> ##
>
> ###########SQL########################### PL/pgSQL#################### #########41.7##########

#####

name

##

BINARY

####################################

ASENSITIVE
INSENSITIVE

###
####### INSENSITIVE#############ASENSITIVE###################### #3#### ###SENSITIVE#############################PostgreSQL########## PostgreSQL### ########insensitive################################SQL####################

INSENSITIVE#FOR UPDATE###FOR SHARE###################

SCROLL
NO SCROLL

SCROLL### ################ ##SCROLL############################## NO SCROLL######################### ############ ######################### ###SCROLL############## ############ ###

WITH HOLD
WITHOUT HOLD

WITH HOLD###
WITHOUT HOLD### WITH HOLD#WITHOUT HOLD##########WITHOUT HOLD###########

query

##################SELECT###VALUES######

ASENSITIVE#BINARY#INSENSITIVE#SCROLL#######################

##

#########SELECT######################## BINARY############################## ##### ## # ##################1##################1## ###4######(###############)############

############################## psql## ###############

##
##################################FETCH################################## ############################## ######################## ##############/###### ##

WITH HOLD###WITH HOLD###DECLARE############################## ######################## ## #####PostgreSQL## ################## ##BEGIN#COMMIT####ROLLBACK###########

WITH HOLD###WITH HOLD#################### #######CLOSE############################## ##

######FOR UPDATE###FOR SHARE######WITH HOLD##############

#####################SCROLL############## #####SQL########### ############### ###########PostgreSQL## SCROLL##### ################## ####SCROLL## NO SCROLL##################################

#########FOR UPDATE###FOR SHARE###################### ########SCROLL###### ########

##
##################36.7############################## ##################################### ################NO SCROLL######## #### ##################1########WITH HOLD################################## ##1##########

###########FOR UPDATE###FOR SHARE##################SELECT################## ############ ############################

##
#####UPDATE ... WHERE CURRENT OF###DELETE ... WHERE CURRENT OF### #########FOR UPDATE########### FOR UPDATE##############################

```
################## FOR UPDATE##########################WHERE CURRENT
OF##############

FOR    UPDATE#################################SQL################
#1###########################ORDER    BY#######################WHERE
CURRENT OF############### ###########################################
####### ########################################### FOR UPDATE#######
#########################

FOR UPDATE#WHERE CURRENT OF#######################################
############################################## ######################
```

##SQL######SQL################## PostgreSQL#########OPEN########## ######
################## ####PostgreSQL#####SQL#########ECPG###DECLARE#OPEN####
#####SQL#################

################################ ##
###################### DECLARE################################

pg_cursors##

# #

##########

```
DECLARE liahona CURSOR FOR SELECT * FROM films;
```

##############FETCH#########

###

##SQL###########SQL################ PostgreSQL####################

##SQL#####UPDATE ... WHERE CURRENT OF#DELETE ... WHERE CURRENT OF####insensitive######################## PostgreSQL##########################
###insensitive##################

#########PostgreSQL######

####

CLOSE, FETCH, MOVE

DELETE

DELETE — ###########

##

```
[ WITH [ RECURSIVE ] with_query [, ...] ]
DELETE FROM [ ONLY ] table_name [ * ] [ [ AS ] alias ]
    [ USING from_item [, ...] ]
    [ WHERE condition | WHERE CURRENT OF cursor_name ]
    [ RETURNING { * | output_expression [ [ AS ] output_name ] } [, ...] ]
```

##

DELETE###########WHERE############ WHERE################################ ## ##############################

> ###
>
> TRUNCATE################################

##SELECT#USING###############2##### ## ####################

RETURNING########DELETE################################ #########USING###### ##################################### RETURNING#######SELECT###########

################DELETE######## ###USING##########SELECT#######*condition*### ###########################SELECT########

#####

with_query

WITH####DELETE#################################### ####7.8#SELECT###### ####

table_name

########################## ########ONLY################################ ONLY## ########## #####*#####################################

alias

########### ############################## #####DELETE FROM foo AS f#### ####DELETE###############foo####f###############

from_item

WHERE################################## ####SELECT##FROM############ ################## ############*from_item*############################ ##*from_item*############################

condition

boolean######### ####true#############

cursor_name

WHERE CURRENT OF################# ################################ ####
##DELETE################################### WHERE CURRENT OF############
########### WHERE CURRENT OF####################DECLARE##########

output_expression

#########DELETE################ ######*table_name*###########USING#######
#################### ###########*##########

output_name

###############

##

##########DELETE######################

DELETE *count*

count######### #####BEFORE DELETE#################*condition*##############
############### *count*#0############################## #################

DELETE#####RETURNING############RETURNING################################
######SELECT################

##

PostgreSQL###USING################WHERE###################### ##############
################################

```
DELETE FROM films USING producers
  WHERE producer_id = producers.id AND producers.name = 'foo';
```

#####films#producers#######films####################### ##################
################

```
DELETE FROM films
  WHERE producer_id IN (SELECT id FROM producers WHERE name = 'foo');
```

#SELECT##################################

#

####################

```
DELETE FROM films WHERE kind <> 'Musical';
```

films###########

```
DELETE FROM films;
```

#######status#DONE#######################

```
DELETE FROM tasks WHERE status = 'DONE' RETURNING *;
```

tasks####c_tasks####################

```
DELETE FROM tasks WHERE CURRENT OF c_tasks;
```

######DELETE##LIMIT#########UPDATE##################################

```
WITH delete_batch AS (
  SELECT l.ctid FROM user_logs AS l
    WHERE l.status = 'archived'
    ORDER BY l.creation_date
    FOR UPDATE
    LIMIT 10000
)
DELETE FROM user_logs AS dl
  USING delete_batch AS del
  WHERE dl.ctid = del.ctid;
```

###

#########SQL######### ####USING##RETURNING##PostgreSQL###### DELETE#WITH#### ###############

####

TRUNCATE

DISCARD

DISCARD — ############

##

```
DISCARD { ALL | PLANS | SEQUENCES | TEMPORARY | TEMP }
```

##

DISCARD############################ ######################################
################################## DISCARD ALL###########################

#####

PLANS

###

SEQUENCES

############################# #####currval()/lastval()###############
###########nextval()################ (####################CREATE SEQUENCE#########)

TEMPORARY###TEMP

#############################

ALL

################################
##

```
CLOSE ALL;
SET SESSION AUTHORIZATION DEFAULT;
RESET ALL;
DEALLOCATE ALL;
UNLISTEN *;
SELECT pg_advisory_unlock_all();
DISCARD PLANS;
DISCARD TEMP;
DISCARD SEQUENCES;
```

##

DISCARD ALL##########################

###

DISCARD#PostgreSQL######

DO

DO — ###############

##

```
DO [ LANGUAGE lang_name ] code
```

##

DO#####################################

#####################void################## ################

LANGUAGE##########################

#####

code

################ ####CREATE FUNCTION############################## ####### #############

lang_name

#################### #########plpgsql###

##

###########CREATE EXTENSION################################ plpgsql###### ############################

############USAGE############## ################################ ############ ###############

DO## DO#################### ####################

#

####public#####################webuser#######

```
DO $$DECLARE r record;
BEGIN
    FOR r IN SELECT table_schema, table_name FROM
 information_schema.tables
             WHERE table_type = 'VIEW' AND table_schema = 'public'
    LOOP
        EXECUTE 'GRANT ALL ON ' || quote_ident(r.table_schema) || '.' ||
 quote_ident(r.table_name) || ' TO webuser';
    END LOOP;
END$$;
```

###

##SQL##DO#######

####

CREATE LANGUAGE

DROP ACCESS METHOD

DROP ACCESS METHOD — ############

##

```
DROP ACCESS METHOD [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP ACCESS METHOD################## #########################

#####

IF EXISTS

########################### ###############

name

################

CASCADE

###
#########5.15####

RESTRICT

####################1################ ###########

#

#######heptree###############

```
DROP ACCESS METHOD heptree;
```

###

DROP ACCESS METHOD#PostgreSQL######

####

CREATE ACCESS METHOD

DROP AGGREGATE

DROP AGGREGATE — ########

##

```
DROP AGGREGATE [ IF EXISTS ] name ( aggregate_signature ) [, ...]
 [ CASCADE | RESTRICT ]
```

```
###aggregate_signature#########

* |
[ argmode ] [ argname ] argtype [ , ... ] |
[ [ argmode ] [ argname ] argtype [ , ... ] ] ORDER BY [ argmode ]
 [ argname ] argtype [ , ... ]
```

##

DROP AGGREGATE########################### ################################
########

#####

IF EXISTS

  ##################### ##################

name

  ########################

argmode

  ########IN###VARIADIC### ############IN###

argname

  ######## DROP AGGREGATE########################### ########################
  ################

argtype

  ##################### ############################*########## ###############
  ################ORDER BY##########

CASCADE

  ##
  ####5.15####

RESTRICT

  ################################### ############

##

##########################ALTER AGGREGATE##########

# #

integer##myavg##########

```
DROP AGGREGATE myavg(integer);
```

##################################myrank#######

```
DROP AGGREGATE myrank(VARIADIC "any" ORDER BY VARIADIC "any");
```

########1#############

```
DROP AGGREGATE myavg(integer), myavg(bigint);
```

###

##SQL###DROP AGGREGATE########

####

ALTER AGGREGATE, CREATE AGGREGATE

DROP CAST

DROP CAST — ########

##

```
DROP CAST [ IF EXISTS ] (source_type AS target_type) [ CASCADE |
 RESTRICT ]
```

##

DROP CAST###################

##########################

#####

IF EXISTS

##################

source_type

##################

target_type

##################

CASCADE
RESTRICT

DROP COLLATION

DROP COLLATION — ########

##

```
DROP COLLATION [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP COLLATION################### ##################################

#####

IF EXISTS

######################## ###################

name

##########################

CASCADE

###5.15####

RESTRICT

################################ ###########

#

german################

```
DROP COLLATION german;
```

###

PostgreSQL######IF EXISTS#########DROP COLLATION#######SQL######

####

ALTER COLLATION, CREATE COLLATION

DROP CONVERSION

DROP CONVERSION — ############

##

```
DROP CONVERSION [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP CONVERSION## ################
############

#####

IF EXISTS

  ###################### ##################

name

  #######################

CASCADE
RESTRICT

  #####################################

#

myname##############

```
DROP CONVERSION myname;
```

###

##SQL##DROP CONVERSION####### ####PostgreSQL#CREATE CONVERSION###CREATE TRANSLATION#######DROP TRANSLATION#######

####

ALTER CONVERSION, CREATE CONVERSION

DROP DATABASE

DROP DATABASE — ##########

##

```
DROP DATABASE [ IF EXISTS ] name [ [ WITH ] ( option [, ...] ) ]
```

###option#########

FORCE

##

DROP DATABASE############## ## ############################## ########################## ############ ##postgres################### ####################################FORCE##### ####################

DROP DATABASE############# ##############

#####

IF EXISTS

######################### ##################

name

################

FORCE

################################ ## ############################

##########################pg_terminate_backend#######################
#9.28.2######## ####################

##

DROP DATABASE#########################

####################dropdb############

###

##SQL##DROP DATABASE#######

####

CREATE DATABASE

DROP DOMAIN

DROP DOMAIN — ########

##

```
DROP DOMAIN [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP DOMAIN########### ########################

#####

IF EXISTS

######################## ##################

name

########################

CASCADE

###5.15####

RESTRICT

################################### ###########

#

####box#######

```
DROP DOMAIN box;
```

###

########PostgreSQL######IF EXISTS###########SQL#########

####

CREATE DOMAIN, ALTER DOMAIN

DROP EVENT TRIGGER

DROP EVENT TRIGGER — ###########

##

```
DROP EVENT TRIGGER [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP EVENT TRIGGER################# ######################################
#####

#####

IF EXISTS

########################### #############

name

#################

CASCADE

###5.15####

RESTRICT

############################### ###########

#

```
###snitch#######
```

```
DROP EVENT TRIGGER snitch;
```

###

DROP EVENT TRIGGER####SQL#######

####

CREATE EVENT TRIGGER, ALTER EVENT TRIGGER

DROP EXTENSION

DROP EXTENSION — #######

##

```
DROP EXTENSION [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP EXTENSION################# ##################################(ALTER ROUTINE#DEPENDS ON EXTENSION *extension_name*###########)########

DROP EXTENSION###########################

#####

IF EXISTS

#################### ##################

name

##################

CASCADE

###5.15####

RESTRICT

###
######################### ###########

#

##########hstore########

```
DROP EXTENSION hstore;
```

###########hstore################hstore############################ ######
####################CASCADE########

###

DROP EXTENSION#PostgreSQL######

####

CREATE EXTENSION, ALTER EXTENSION

DROP FOREIGN DATA WRAPPER

DROP FOREIGN DATA WRAPPER — ##############

##

```
DROP FOREIGN DATA WRAPPER [ IF EXISTS ] name [, ...] [ CASCADE |
 RESTRICT ]
```

##

DROP FOREIGN DATA WRAPPER################### ############################## ################

#####

IF EXISTS

############################ ###############

name

##################

CASCADE

###5.15####

RESTRICT

#################################### ###########

#

#########dbi#######

```
DROP FOREIGN DATA WRAPPER dbi;
```

###

DROP FOREIGN DATA WRAPPER#ISO/IEC 9075-9 (SQL/MED)###### IF EXISTS## PostgreSQL######

####

CREATE FOREIGN DATA WRAPPER, ALTER FOREIGN DATA WRAPPER

DROP FOREIGN TABLE

DROP FOREIGN TABLE — ##########

##

```
DROP FOREIGN TABLE [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP FOREIGN TABLE############# ########################

#####

IF EXISTS

########################## ##################

name

##########################

CASCADE

###
#5.15####

RESTRICT

################################## ############

#

films###distributors##################

```
DROP FOREIGN TABLE films, distributors;
```

###

########ISO/IEC 9075-9 (SQL/MED)##################################PostgreSQL#####IF EXISTS##################

####

ALTER FOREIGN TABLE, CREATE FOREIGN TABLE

DROP FUNCTION

DROP FUNCTION — ######

##

```
DROP FUNCTION [ IF EXISTS ] name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] [, ...]
    [ CASCADE | RESTRICT ]
```

##

DROP FUNCTION############## ########################## ####################
################################

#####

IF EXISTS

  ###################### ##################

name

  ###################### ####################################

argmode

  #######IN#OUT#INOUT#VARIADIC####### #############IN### ############
  ##############DROP FUNCTION#OUT################## #####IN#INOUT###
  #VARIADIC##############

argname

  ####### ##############################DROP FUNCTION####################
  ##

argtype

  ##############################

CASCADE

  ##5.15####

RESTRICT

  ############################### ###########

#

##################

```
DROP FUNCTION sqrt(integer);
```

######1############

```
DROP FUNCTION sqrt(integer), sqrt(bigint);
```

###

```
DROP FUNCTION update_employee_salaries;
```

######################

```
DROP FUNCTION update_employee_salaries();
```

##

###

#########SQL###########PostgreSQL######

- ##SQL##1##########1###############
- `IF EXISTS`#####
- ################

####

CREATE FUNCTION, ALTER FUNCTION, DROP PROCEDURE, DROP ROUTINE

DROP GROUP

DROP GROUP — ##############

##

```
DROP GROUP [ IF EXISTS ] name [, ...]
```

##

DROP GROUP#DROP ROLE##########

###

##SQL##DROP GROUP########

####

DROP ROLE

DROP INDEX

DROP INDEX — ##########

##

```
DROP INDEX [ CONCURRENTLY ] [ IF EXISTS ] name [, ...] [ CASCADE |
 RESTRICT ]
```

##

DROP INDEX########################### ####################################

#####

CONCURRENTLY

###DROP INDEX#### ######ACCESS EXCLUSIVE#################################### ############

############################## ######################CASCADE############ ### ######UNIQUE###PRIMARY KEY################################ ##### #DROP INDEX#########################DROP INDEX CONCURRENTLY####### ## ##

###########DROP INDEX### ######

IF EXISTS

########################## ##################

name

##########################

CASCADE

###5.15####

RESTRICT

################################### ############

#

############title_idx#######

```
DROP INDEX title_idx;
```

###

DROP INDEX#PostgreSQL####### ##SQL####################

####

CREATE INDEX

DROP LANGUAGE

DROP LANGUAGE — #########

##

```
DROP [ PROCEDURAL ] LANGUAGE [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP LANGUAGE######################### DROP LANGUAGE#########################
#######

> ##
>
> PostgreSQL 9.1############################DROP LANGUAGE####DROP EXTENSION############

#####

IF EXISTS

#################### ##################

name

##############

CASCADE

###
#5.15####

RESTRICT

############################### ############

#

#######plsample##############

```
DROP LANGUAGE plsample;
```

###

##SQL##DROP LANGUAGE#######

####

ALTER LANGUAGE, CREATE LANGUAGE

DROP MATERIALIZED VIEW

DROP MATERIALIZED VIEW — ###############

##

```
DROP MATERIALIZED VIEW [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP MATERIALIZED VIEW#################### ############################## #########

#####

IF EXISTS

  ############################### #############

name

  ##############################

CASCADE

  ##
  ############5.15####

RESTRICT

  ##################################### ###########

#

########order_summary######################

```
DROP MATERIALIZED VIEW order_summary;
```

###

DROP MATERIALIZED VIEW#PostgreSQL######

####

CREATE MATERIALIZED VIEW, ALTER MATERIALIZED VIEW, REFRESH MATERIALIZED VIEW

DROP OPERATOR

DROP OPERATOR — ########

##

```
DROP OPERATOR [ IF EXISTS ] name ( { left_type | NONE } , right_type )
 [, ...] [ CASCADE | RESTRICT ]
```

##

DROP OPERATOR########################### ##################################

#####

IF EXISTS

######################## ##################

name

######################

left_type

################## ################NONE#######

right_type

##################

CASCADE

###
#5.15####

RESTRICT

############################ ###########

#

integer###########a^n#######

```
DROP OPERATOR ^ (integer, integer);
```

bit##################~b#######

```
DROP OPERATOR ~ (none, bit);
```

#######1#############

```
DROP OPERATOR ~ (none, bit), ^ (integer, integer);
```

###

##SQL##DROP OPERATOR########

####

CREATE OPERATOR, ALTER OPERATOR

DROP OPERATOR CLASS

DROP OPERATOR CLASS — ##########

##

```
DROP OPERATOR CLASS [ IF EXISTS ] name USING index_method [ CASCADE |
 RESTRICT ]
```

##

DROP OPERATOR CLASS################ ##################################

DROP OPERATOR CLASS############################ ########################### #######CASCADE############

#####

IF EXISTS

  ######################### ##################

name

  ########################

index_method

  ################################

CASCADE

  ### #5.15####

RESTRICT

  ##################################### ############

##

DROP OPERATOR CLASS#################### ############################CREATE OPERATOR CLASS################# ##########################DROP OPERATOR FAMILY###################### #######DROP OPERATOR FAMILY#######

#

widget_ops######B-tree############

```
DROP OPERATOR CLASS widget_ops USING btree;
```

############################CASCADE##

###

##SQL##DROP OPERATOR CLASS#######

####

ALTER OPERATOR CLASS, CREATE OPERATOR CLASS, DROP OPERATOR FAMILY

DROP OPERATOR FAMILY

DROP OPERATOR FAMILY — ########

##

```
DROP OPERATOR FAMILY [ IF EXISTS ] name USING index_method [ CASCADE |
 RESTRICT ]
```

##

DROP OPERATOR FAMILY############## ##################################

DROP OPERATOR FAMILY################################## ################## ############# ##CASCADE########### ##

#####

IF EXISTS

  ###################### ##################

name

  ######################

index_method

  ############################

CASCADE

  ##5.15####

RESTRICT

  ################################## ###########

#

B-tree####float_ops######

```
DROP OPERATOR FAMILY float_ops USING btree;
```

#CASCADE########

###

##SQL##DROP OPERATOR FAMILY#######

####

ALTER OPERATOR FAMILY, CREATE OPERATOR FAMILY, ALTER OPERATOR CLASS, CREATE OPERATOR CLASS, DROP OPERATOR CLASS

DROP OWNED

DROP OWNED — ##################################

##

```
DROP OWNED BY { name | CURRENT_ROLE | CURRENT_USER | SESSION_USER }
 [, ...] [ CASCADE | RESTRICT ]
```

##

DROP OWNED## ######################### ##

#####

name

##################################

CASCADE

###5.15####

RESTRICT

###########

##

DROP OWNED######################### DROP OWNED############################## ##

CASCADE###

####REASSIGN OWNED#######1## ###REASSIGN OWNED######################

##################################

####21.4##########

###

DROP OWNED#####PostgreSQL######

####

REASSIGN OWNED, DROP ROLE

DROP POLICY

DROP POLICY — ########################

##

```
DROP POLICY [ IF EXISTS ] name ON table_name [ CASCADE | RESTRICT ]
```

##

DROP POLICY#################### ###########################ALTER TABLE# ## ####################ALTER TABLE ... DISABLE ROW LEVEL SECURITY##############################

#####

IF EXISTS

######################## ##############

name

#############

table_name

##############################

CASCADE
RESTRICT

#################### #################

#

my_table###########p1#######################

```
DROP POLICY p1 ON my_table;
```

###

DROP POLICY#PostgreSQL######

####

CREATE POLICY, ALTER POLICY

DROP PROCEDURE

DROP PROCEDURE — ##########

##

```
DROP PROCEDURE [ IF EXISTS ] name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] [, ...]
    [ CASCADE | RESTRICT ]
```

##

DROP PROCEDURE#1################### ############################### ####### ################### ####################################

#####

IF EXISTS

######################### ######(NOTICE)############

name

#####################

argmode

#######IN#OUT#INOUT#VARIADIC####### ############IN##############

argname

######## ####################################DROP PROCEDURE############### ########

argtype

################################## ###############

CASCADE

##5.15####

RESTRICT

################################### ##########

##

##############1#################### ################

PostgreSQL########INOUT############## ## ### ###DROP#####OUT########################## #########OUT################ ### ########CREATE################################

SQL##########*argmode*##################OUT#################### ############ #####OUT########################## ############*argmode*#####################

###################### DROP################2########################## ###### #############IN#########################PostgreSQL###################

############ALTER PROCEDURE#COMMENT ON PROCEDURE#######################
########

# #

do_db_maintenance#######1#####################

```
DROP PROCEDURE do_db_maintenance;
```

##################

```
CREATE PROCEDURE do_db_maintenance(IN target_schema text, OUT results
 text) ...
```

#####################################

```
DROP PROCEDURE do_db_maintenance(IN target_schema text, OUT results text);
DROP PROCEDURE do_db_maintenance(IN text, OUT text);
DROP PROCEDURE do_db_maintenance(IN text);
DROP PROCEDURE do_db_maintenance(text);
DROP PROCEDURE do_db_maintenance(text, text);  -- potentially ambiguous
```

##########################

```
CREATE PROCEDURE do_db_maintenance(IN target_schema text, IN options
 text) ...
```

###

#######SQL#############PostgreSQL#########

- ############################
- IF EXISTS##########
- ##

####

CREATE PROCEDURE, ALTER PROCEDURE, DROP FUNCTION, DROP ROUTINE

DROP PUBLICATION

DROP PUBLICATION — ############

##

```
DROP PUBLICATION [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP PUBLICATION##########################

##

#####

IF EXISTS

########################### ####################

name

################

CASCADE
RESTRICT

##

#

##############

```
DROP PUBLICATION mypublication;
```

###

DROP PUBLICATION#PostgreSQL######

####

CREATE PUBLICATION, ALTER PUBLICATION

DROP ROLE

DROP ROLE — ##############

##

```
DROP ROLE [ IF EXISTS ] name [, ...]
```

##

DROP ROLE################### ##################################### ##########
###########CREATEROLE###############ADMIN OPTION################

################
##################################### #####################################
#########REASSIGN OWNED###DROP OWNED########## ####21.4#########

############################## DROP ROLE################################## #####
##########################

#####

IF EXISTS

  ###################### ##################

name

  ##############

##

PostgreSQL######################dropuser###### ############################## ####
####################

#

##########

```
DROP ROLE jonathan;
```

###

##SQL##DROP ROLE#############1################### ###PostgreSQL###############
#######

####

CREATE ROLE, ALTER ROLE, SET ROLE

DROP ROUTINE

DROP ROUTINE — ########

##

```
DROP ROUTINE [ IF EXISTS ] name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] [, ...]
    [ CASCADE | RESTRICT ]
```

##

DROP ROUTINE#1################# ################################ ############ ######DROP AGGREGATE#DROP FUNCTION#DROP PROCEDURE#########

##

DROP ROUTINE##################DROP PROCEDURE###### ##DROP ROUTINE##*argmode*###########OUT###################SQL############## ################### (DROP AGGREGATE#DROP FUNCTION##############)

##################################(DROP FUNCTION#)#########DROP ROUTINE## ###################### ##

ALTER ROUTINE#COMMENT ON ROUTINE##

#

integer########foo######

```
DROP ROUTINE foo(integer);
```

#######foo############################

###

#######SQL##############PostgreSQL########

- ##########################
- IF EXISTS##########
- ##
- #################

####

DROP AGGREGATE, DROP FUNCTION, DROP PROCEDURE, ALTER ROUTINE

CREATE ROUTINE##################

DROP RULE

DROP RULE — ###########

##

```
DROP RULE [ IF EXISTS ] name ON table_name [ CASCADE | RESTRICT ]
```

##

DROP RULE##############

#####

IF EXISTS

###################### ##################

name

############

table_name

#####################################

CASCADE

##5.15####

RESTRICT

########################### ###########

#

newrule################

```
DROP RULE newrule ON mytable;
```

###

DROP RULE#PostgreSQL############################

####

CREATE RULE, ALTER RULE

DROP SCHEMA

DROP SCHEMA — ########

##

```
DROP SCHEMA [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP SCHEMA###################

############################## ##
####################

#####

IF EXISTS

######################## ##################

name

##########

CASCADE

###5.15#
###

RESTRICT

################################### ############

##

CASCADE##

#

########mystuff##############################

```
DROP SCHEMA mystuff CASCADE;
```

###

##SQL#############1#########################PostgreSQL######IF EXISTS##
##DROP SCHEMA####SQL############

####

ALTER SCHEMA, CREATE SCHEMA

DROP SEQUENCE

DROP SEQUENCE — ##########

##

```
DROP SEQUENCE [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP SEQUENCE#################### ################################

#####

IF EXISTS

######################### ####################

name

######################

CASCADE

###5.15####

RESTRICT

############################## ############

#

serial#################

```
DROP SEQUENCE serial;
```

###

####1#####1#######################PostgreSQL###### IF EXISTS#########DROP SEQUENCE###SQL#######

####

CREATE SEQUENCE, ALTER SEQUENCE

DROP SERVER

DROP SERVER — ##############

##

```
DROP SERVER [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP SERVER################## ##

#####

IF EXISTS

######################## ###############

name

############

CASCADE

###5.15####

RESTRICT

############################### ###########

#

```
###foo##############

DROP SERVER IF EXISTS foo;
```

###

DROP SERVER#ISO/IEC 9075-9 (SQL/MED)###### IF EXISTS##PostgreSQL######

####

CREATE SERVER, ALTER SERVER

DROP STATISTICS

DROP STATISTICS — ########

##

```
DROP STATISTICS [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP STATISTICS###################### ################################### ############

#####

IF EXISTS

########################## ##################

name

################################

CASCADE
RESTRICT

################################

#

#########2#######################################

```
DROP STATISTICS IF EXISTS
    accounting.users_uid_creation,
    public.grants_user_role;
```

###

##SQL##DROP STATISTICS##########

####

ALTER STATISTICS, CREATE STATISTICS

DROP SUBSCRIPTION

DROP SUBSCRIPTION — ##############

##

```
DROP SUBSCRIPTION [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP SUBSCRIPTION############################

###

#####[77.431907]##################################DROP SUBSCRIPTION#########
################### ###############ALTER SUBSCRIPTION############ ###########
##DROP SUBSCRIPTION###############
#######ALTER SUBSCRIPTION########

#####

name

###################

CASCADE
RESTRICT

###

##

#####[87.707881]###DROP
SUBSCRIPTION##################################(##############)########## ###
################################
##DROP SUBSCRIPTION########## #####
##############ALTER SUBSCRIPTION ... DISABLE#######################ALTER
SUBSCRIPTION ... SET (slot_name = NONE)############################
######### #####DROP SUBSCRIPTION########################## ###############
############## (##############) ######################## ########WAL#########
###################### 29.2.1######### ######################################
#################DROP SUBSCRIPTION######################################
##################### ## ########
###DROP SUBSCRIPTION#
######### ################ALTER SUBSCRIPTION ... DISABLE#####ALTER
SUBSCRIPTION ... SET (slot_name = NONE)####### ####DROP SUBSCRIPTION##
########################## ##
######### #######WAL############################## 29.2.1#########

##################################DROP SUBSCRIPTION#########################
####

#

################

```
DROP SUBSCRIPTION mysub;
```

###

`DROP SUBSCRIPTION`#PostgreSQL######

####

CREATE SUBSCRIPTION, ALTER SUBSCRIPTION

DROP TABLE

DROP TABLE — ########

##

```
DROP TABLE [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP TABLE################### ### ##### #######################DELETE###TRUNCATE##########

DROP TABLE## ##################### #####################CASCADE############ #CASCADE########################### ##

#####

IF EXISTS

########################## ####################

name

#########################

CASCADE

###5.15# ###

RESTRICT

################################## ############

#

2#######films#distributors#######

```
DROP TABLE films, distributors;
```

###

####1#####1#####################PostgreSQL######IF EXISTS##################SQL ######

####

ALTER TABLE, CREATE TABLE

DROP TABLESPACE

DROP TABLESPACE — ##########

##

```
DROP TABLESPACE [ IF EXISTS ] name
```

##

DROP TABLESPACE###################

################################### ## ## ## ####### ##########temp_tablespaces##DROP#### #########

#####

IF EXISTS

  ######################### ##################

name

  ###########

##

#############DROP TABLESPACE#############

#

######mystuff############

```
DROP TABLESPACE mystuff;
```

###

DROP TABLESPACE#PostgreSQL######

####

CREATE TABLESPACE, ALTER TABLESPACE

DROP TEXT SEARCH CONFIGURATION

DROP TEXT SEARCH CONFIGURATION — ############

##

```
DROP TEXT SEARCH CONFIGURATION [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP TEXT SEARCH CONFIGURATION################## ######################### #########

#####

IF EXISTS

########################## ##############

name

#########################

CASCADE

##5.15####

RESTRICT

################################## ###########

#

########my_english#######

```
DROP TEXT SEARCH CONFIGURATION my_english;
```

to_tsvector## ######################### #######CASCADE########

###

##SQL##DROP TEXT SEARCH CONFIGURATION#######

####

ALTER TEXT SEARCH CONFIGURATION, CREATE TEXT SEARCH CONFIGURATION

DROP TEXT SEARCH DICTIONARY

DROP TEXT SEARCH DICTIONARY — ############

##

```
DROP TEXT SEARCH DICTIONARY [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP TEXT SEARCH DICTIONARY################## ########################### #######

#####

IF EXISTS

########################## ##############

name

#########################

CASCADE

##5.15####

RESTRICT

################################## ###########

#

#######english#######

```
DROP TEXT SEARCH DICTIONARY english;
```

#################################### ####################CASCADE#########

###

##SQL##DROP TEXT SEARCH DICTIONARY#######

####

ALTER TEXT SEARCH DICTIONARY, CREATE TEXT SEARCH DICTIONARY

DROP TEXT SEARCH PARSER

DROP TEXT SEARCH PARSER — ##############

##

```
DROP TEXT SEARCH PARSER [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP TEXT SEARCH PARSER################### ################################ ##

#####

IF EXISTS

############################ ###############

name

##########################

CASCADE

###5.15####

RESTRICT

#################################### ###########

#

```
########my_parser#######

DROP TEXT SEARCH PARSER my_parser;
```

######################CASCADE#########

###

##SQL##DROP TEXT SEARCH PARSER#######

####

ALTER TEXT SEARCH PARSER, CREATE TEXT SEARCH PARSER

DROP TEXT SEARCH TEMPLATE

DROP TEXT SEARCH TEMPLATE — ################

##

```
DROP TEXT SEARCH TEMPLATE [ IF EXISTS ] name [ CASCADE | RESTRICT ]
```

##

DROP TEXT SEARCH TEMPLATE###################### ########################### #######

#####

IF EXISTS

################################ ###############

name

#############################

CASCADE

###5.15####

RESTRICT

###########

#

###########thesaurus#######

```
DROP TEXT SEARCH TEMPLATE thesaurus;
```

##########################CASCADE###

###

##SQL##DROP TEXT SEARCH TEMPLATE#######

####

ALTER TEXT SEARCH TEMPLATE, CREATE TEXT SEARCH TEMPLATE

DROP TRANSFORM

DROP TRANSFORM — #######

##

```
DROP TRANSFORM [ IF EXISTS ] FOR type_name LANGUAGE lang_name [ CASCADE |
 RESTRICT ]
```

##

DROP TRANSFORM##################

############################## ###########################

#####

IF EXISTS

######################### ###############

type_name

#############

lang_name

###########

CASCADE

###5.15####

RESTRICT

############################## ###########

#

hstore####plpython3u##################

```
DROP TRANSFORM FOR hstore LANGUAGE plpython3u;
```

###

#####DROP TRANSFORM#PostgreSQL###### ####CREATE TRANSFORM##########

####

CREATE TRANSFORM

DROP TRIGGER

DROP TRIGGER — #######

##

```
DROP TRIGGER [ IF EXISTS ] name ON table_name [ CASCADE | RESTRICT ]
```

##

DROP TRIGGER############### ##################################

#####

IF EXISTS

##

name

############

table_name

#############################

CASCADE

###5.15####

RESTRICT

############################### ############

#

```
####films######if_dist_exists#######

DROP TRIGGER if_dist_exists ON films;
```

###

PostgreSQL#DROP TRIGGER#####SQL########### ##SQL#######################
##DROP TRIGGER *name*###############

####

CREATE TRIGGER

DROP TYPE

DROP TYPE — ########

##

```
DROP TYPE [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP TYPE################# ######################

#####

IF EXISTS

####################### ####################

name

#########################

CASCADE

###
###5.15####

RESTRICT

################################# ############

#

box###########

```
DROP TYPE box;
```

###

########PostgreSQL######IF EXISTS###########SQL################# ####PostgreSQL
#CREATE TYPE###################SQL################

####

ALTER TYPE, CREATE TYPE

DROP USER

DROP USER — ##############

##

```
DROP USER [ IF EXISTS ] name [, ...]
```

##

DROP USER#DROP ROLE###########

###

DROP USER##PostgreSQL###### ##SQL#####################

####

DROP ROLE

DROP USER MAPPING

DROP USER MAPPING — ##################

##

```
DROP USER MAPPING [ IF EXISTS ] FOR { user_name | USER | CURRENT_ROLE |
 CURRENT_USER | PUBLIC } SERVER server_name
```

##

DROP USER MAPPING######################

########USAGE##################
##################################

#####

IF EXISTS

  ######################### ##############

user_name

  ############# CURRENT_ROLE#CURRENT_USER#USER################ PUBLIC##
  ##################################

server_name

  #############

#

##########foo########bob#######

```
DROP USER MAPPING IF EXISTS FOR bob SERVER foo;
```

###

DROP USER MAPPING#ISO/IEC 9075-9 (SQL/MED)###### IF EXISTS##PostgreSQL######

####

CREATE USER MAPPING, ALTER USER MAPPING

DROP VIEW

DROP VIEW — ########

##

```
DROP VIEW [ IF EXISTS ] name [, ...] [ CASCADE | RESTRICT ]
```

##

DROP VIEW############## ##########################

#####

IF EXISTS

########################## ###################

name

#######################

CASCADE

###
#5.15####

RESTRICT

############################## ############

#

#######kinds###############

```
DROP VIEW kinds;
```

###

####1#####1#######################PostgreSQL######IF EXISTS#################
SQL########

####

ALTER VIEW, CREATE VIEW

END

END — ################

##

```
END [ WORK | TRANSACTION ] [ AND [ NO ] CHAIN ]
```

##

END#################### ## ######## PostgreSQL#####COMMIT######

#####

WORK
TRANSACTION

############ ###########

AND CHAIN

AND CHAIN##(SET TRANSACTION# ########)########### ###########################

##

##############ROLLBACK##########

############END##############################

#

##############################

END;

###

END#PostgreSQL##################SQL######COMMIT######

####

BEGIN, COMMIT, ROLLBACK

EXECUTE

EXECUTE — ##########

##

```
EXECUTE name [ ( parameter [, ...] ) ]
```

##

EXECUTE############################ ################################## #PREPARE#####################

######PREPARE################################EXECUTE############ ####### ####### ## ##########1##### ######################

#######################PREPARE#########

#####

name

###################

parameter

########################## ## #########

##

EXECUTE############EXECUTE######################

#

##PREPARE####Examples######

###

##SQL##EXECUTE#################SQL######### ##########EXECUTE########### ########

####

DEALLOCATE, PREPARE

EXPLAIN

EXPLAIN — ###############

##

```
EXPLAIN [ ( option [, ...] ) ] statement
```

```
###option###################

    ANALYZE [ boolean ]
    VERBOSE [ boolean ]
    COSTS [ boolean ]
    SETTINGS [ boolean ]
    GENERIC_PLAN [ boolean ]
    BUFFERS [ boolean ]
    SERIALIZE [ { NONE | TEXT | BINARY } ]
    WAL [ boolean ]
    TIMING [ boolean ]
    SUMMARY [ boolean ]
    MEMORY [ boolean ]
    FORMAT { TEXT | XML | JSON | YAML }
```

##

##########PostgreSQL################### ######################################
##
#######

################################### ##
#################### ######2########### 1##########################1###########
############ ################################EXISTS############################
##1################ ###LIMIT####
##

ANALYZE################################## ####################################
########################### ##

> ##
>
> ANALYZE############################## EXPLAIN#SELECT##############
> #################### INSERT#UPDATE#DELETE#MERGE#CREATE TABLE
> AS#EXECUTE###################EXPLAIN ANALYZE#######################
>
> ```
> BEGIN;
> EXPLAIN ANALYZE ...;
> ROLLBACK;
> ```

#####

ANALYZE

> ################################ ##############FALSE###

VERBOSE

################## ## ####
#########################
#compute_query_id######### #############FALSE###

COSTS

#############TRUE###

SETTINGS

################## ## #############
#FALSE###

GENERIC_PLAN

##$1### ################
##############PREPARE######### #######ANALYZE######## ######FALSE###

BUFFERS

#####[88.933440]##################### ####################################
###track_io_timing###
##(#####)######## ################
################################## ################################## ##
#################################### ####################################
############### ## #######
############
#################################### ####################### #############
#FALSE### ######################### ######################################
###track_io_timing#######################
##############################
################## ##
######################Materialize############################## #############
################################### ######################################
######################### ##
####### ######################### #############FALSE###

SERIALIZE

################################
####### ##################TOAST#######################################
########################### EXPLAIN###########SERIALIZE NONE#########
SERIALIZE TEXT###SERIALIZE BINARY##############################
##########TIMING OFF############ BUFFERS##############################
########### ####EXPLAIN########################### #######################
####### ####ANALYZE################ #######SERIALIZE#####TEXT#######

WAL

WAL################# #####################(fpi)#######WAL###############
#######ANALYZE#####################
#FALSE###

TIMING

################################## ##
################# ##FALSE############
########ANALYZE#
########################### ######TRUE###

SUMMARY

################################## #####ANALYZE##########################
################################ EXPLAIN EXECUTE######################
##############################

MEMORY

#################################### ####################################
##################### #############FALSE###

FORMAT

########### TEXT#XML#JSON#YAML####### TEXT######TEXT##############
##################### #############TEXT###

boolean

############################ ##############TRUE#ON#1########## ######
#FALSE#OFF# 0########## *boolean*######## ####TRUE#######

statement

##########
##SELECT#INSERT#UPDATE#DELETE#MERGE#VALUES#EXECUTE#DECLARE#CREATE
TABLE AS#CREATE MATERIALIZED VIEW AS#########

##

########*statement*##################### ######################### 14.1#######
###########

##

PostgreSQL##
#pg_statistic################### ######################### ###############
##ANALYZE################

############################EXPLAIN ANALYZE################################
################EXPLAIN ANALYZE###################################
################################## ##
##############################

# #

integer##1####10000##

```
EXPLAIN SELECT * FROM foo;

                       QUERY PLAN
-------------------------------------------------------
 Seq Scan on foo  (cost=0.00..155.00 rows=10000 width=4)
(1 row)
```

##########JSON#############

```
EXPLAIN (FORMAT JSON) SELECT * FROM foo;
           QUERY PLAN
```

```
--------------------------------
 [                              +
   {                            +
     "Plan": {                  +
       "Node Type": "Seq Scan",+
       "Relation Name": "foo", +
       "Alias": "foo",         +
       "Startup Cost": 0.00,   +
       "Total Cost": 155.00,   +
       "Plan Rows": 10000,     +
       "Plan Width": 4         +
     }                          +
   }                            +
 ]
(1 row)

################WHERE#################EXPLAIN###################


EXPLAIN SELECT * FROM foo WHERE i = 4;

                          QUERY PLAN
-------------------------------------------------------------
 Index Scan using fi on foo  (cost=0.00..5.98 rows=1 width=4)
   Index Cond: (i = 4)
(2 rows)

###########YAML###########


EXPLAIN (FORMAT YAML) SELECT * FROM foo WHERE i='4';
          QUERY PLAN
--------------------------------
 - Plan:                       +
     Node Type: "Index Scan"  +
     Scan Direction: "Forward"+
     Index Name: "fi"         +
     Relation Name: "foo"     +
     Alias: "foo"             +
     Startup Cost: 0.00       +
     Total Cost: 5.98         +
     Plan Rows: 1             +
     Plan Width: 4            +
     Index Cond: "(i = 4)"
(1 row)

#########XML##############

########################


EXPLAIN (COSTS FALSE) SELECT * FROM foo WHERE i = 4;

        QUERY PLAN
----------------------------
 Index Scan using fi on foo
   Index Cond: (i = 4)
(2 rows)

####################################


EXPLAIN SELECT sum(i) FROM foo WHERE i < 10;

                                QUERY PLAN
```

```
-----------------------------------------------------------------
 Aggregate  (cost=23.93..23.93 rows=1 width=4)
   ->  Index Scan using fi on foo  (cost=0.00..23.92 rows=6 width=4)
         Index Cond: (i < 10)
(3 rows)
```

####EXPLAIN EXECUTE#########################

```
PREPARE query(int, int) AS SELECT sum(bar) FROM test
    WHERE id > $1 AND id < $2
    GROUP BY foo;

EXPLAIN ANALYZE EXECUTE query(100, 200);

                                                        QUERY PLAN
-----------------------------------------------------------------
-------------------------------------------------------
 HashAggregate  (cost=10.77..10.87 rows=10 width=12) (actual
 time=0.043..0.044 rows=10 loops=1)
   Group Key: foo
   Batches: 1  Memory Usage: 24kB
   ->  Index Scan using test_pkey on test  (cost=0.29..10.27 rows=99
 width=8) (actual time=0.009..0.025 rows=99 loops=1)
         Index Cond: ((id > 100) AND (id < 200))
 Planning Time: 0.244 ms
 Execution Time: 0.073 ms
(7 rows)
```

##############################
##PostgreSQL############################ ####ANALYZE#########################
##################################ANALYZE#####################

######EXECUTE### ##################
##################GENERIC_PLAN######

```
EXPLAIN (GENERIC_PLAN)
  SELECT sum(bar) FROM test
    WHERE id > $1 AND id < $2
    GROUP BY foo;

                                  QUERY PLAN
-----------------------------------------------------------------
------------
 HashAggregate  (cost=26.79..26.89 rows=10 width=12)
   Group Key: foo
   ->  Index Scan using test_pkey on test  (cost=0.29..24.29 rows=500
 width=8)
         Index Cond: ((id > $1) AND (id < $2))
(4 rows)
```

##########$1#$2#id#######################PREPARE#########################
##

```
EXPLAIN (GENERIC_PLAN)
  SELECT sum(bar) FROM test
    WHERE id > $1::integer AND id < $2::integer
    GROUP BY foo;
```

###

##SQL##EXPLAIN###########

######PostgreSQL#####9.0##################################

```
EXPLAIN [ ANALYZE ] [ VERBOSE ] statement
```

##

####

ANALYZE

FETCH

FETCH — ####################

##

```
FETCH [ direction ] [ FROM | IN ] cursor_name
```

###direction#################

```
    NEXT
    PRIOR
    FIRST
    LAST
    ABSOLUTE count
    RELATIVE count
    count
    ALL
    FORWARD
    FORWARD count
    FORWARD ALL
    BACKWARD
    BACKWARD count
    BACKWARD ALL
```

##

FETCH##########################

###################FETCH######### ### ########## ########################## ############################## FETCH# ## FETCH ALL####FETCH BACKWARD ALL############################

NEXT#PRIOR#FIRST#LAST#ABSOLUTE#RELATIVE####################1####### #### ##

FORWARD###BACKWARD###
#*count*##################/#######

RELATIVE 0#FORWARD 0####BACKWARD 0########################## ########## ################## ################################### ######################
##

> ##
>
> ###########SQL############################## PL/pgSQL################## ######41.7.3#########

#####

direction

direction#################### ###############

NEXT

########## ####*direction*###############

PRIOR

1###########

FIRST

############### #ABSOLUTE 1######

LAST

############### #ABSOLUTE -1######

ABSOLUTE *count*

#####*count*########### *count*##########abs(*count*)########### *count*#
############################# ###ABSOLUTE 0################

RELATIVE *count*

##########*count*########## *count*###########abs(*count*)############
RELATIVE 0##########################

count

##*count*####### #FORWARD *count*######

ALL

############## #FORWARD ALL######

FORWARD

1########## #NEXT######

FORWARD *count*

##*count*########### FORWARD 0###################

FORWARD ALL

###############

BACKWARD

1########## #PRIOR######

BACKWARD *count*

##*count*#################### BACKWARD 0###################

BACKWARD ALL

###########################

count

count################################## FORWARD#BACKWARD#####*count*####
#######FORWARD#BACKWARD###############

cursor_name

##################

##

#########FETCH######################

FETCH *count*

count##########0########## psql################################ psql##########
###########

##

FETCH#######FETCH NEXT#####count#FETCH FORWARD##############SCROLL#####
############### ##########PostgreSQL#######SCROLL###########################
############################ #####NO SCROLL################################

ABSOLUTE#################################### ########################### ##
######################### ## ##
###FETCH ABSOLUTE 0######################

DECLARE############### #######################MOVE##########

#

############################

BEGIN WORK;

-- ###########
DECLARE liahona SCROLL CURSOR FOR SELECT * FROM films;

-- ####liahona#####5#########
FETCH FORWARD 5 FROM liahona;

```
 code  |          title          | did | date_prod  |   kind   |  len
-------+-------------------------+-----+------------+----------+-------
 BL101 | The Third Man           | 101 | 1949-12-23 | Drama    | 01:44
 BL102 | The African Queen       | 101 | 1951-08-11 | Romantic | 01:43
 JL201 | Une Femme est une Femme | 102 | 1961-03-12 | Romantic | 01:25
 P_301 | Vertigo                 | 103 | 1958-11-14 | Action   | 02:08
 P_302 | Becket                  | 103 | 1964-02-03 | Drama    | 02:28
```

-- 1############
FETCH PRIOR FROM liahona;

```
 code  |  title  | did | date_prod  |  kind  |  len
-------+---------+-----+------------+--------+-------
 P_301 | Vertigo | 103 | 1958-11-14 | Action | 02:08
```

-- #######################
CLOSE liahona;
COMMIT WORK;

###

##SQL#######SQL####FETCH########### ##########FETCH###############SELECT## ################# #######FETCH######SQL###########

FORWARD#BACKWARD###FETCH########FORWARD###FETCH *count*#FETCH ALL# PostgreSQL######

##SQL#################FROM##### IN#########################PostgreSQL######

####

CLOSE, DECLARE, MOVE

GRANT

GRANT — ###########

##

```
GRANT { { SELECT | INSERT | UPDATE | DELETE | TRUNCATE | REFERENCES |
 TRIGGER | MAINTAIN }
    [, ...] | ALL [ PRIVILEGES ] }
    ON { [ TABLE ] table_name [, ...]
         | ALL TABLES IN SCHEMA schema_name [, ...] }
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { { SELECT | INSERT | UPDATE | REFERENCES } ( column_name [, ...] )
    [, ...] | ALL [ PRIVILEGES ] ( column_name [, ...] ) }
    ON [ TABLE ] table_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { { USAGE | SELECT | UPDATE }
    [, ...] | ALL [ PRIVILEGES ] }
    ON { SEQUENCE sequence_name [, ...]
         | ALL SEQUENCES IN SCHEMA schema_name [, ...] }
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { { CREATE | CONNECT | TEMPORARY | TEMP } [, ...] | ALL
 [ PRIVILEGES ] }
    ON DATABASE database_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { USAGE | ALL [ PRIVILEGES ] }
    ON DOMAIN domain_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { USAGE | ALL [ PRIVILEGES ] }
    ON FOREIGN DATA WRAPPER fdw_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { USAGE | ALL [ PRIVILEGES ] }
    ON FOREIGN SERVER server_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { EXECUTE | ALL [ PRIVILEGES ] }
    ON { { FUNCTION | PROCEDURE | ROUTINE } routine_name [ ( [ [ argmode ]
 [ arg_name ] arg_type [, ...] ] ) ] [, ...]
         | ALL { FUNCTIONS | PROCEDURES | ROUTINES } IN SCHEMA schema_name
 [, ...] }
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { USAGE | ALL [ PRIVILEGES ] }
    ON LANGUAGE lang_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]
```

```
GRANT { { SELECT | UPDATE } [, ...] | ALL [ PRIVILEGES ] }
    ON LARGE OBJECT loid [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { { SET | ALTER SYSTEM } [, ... ] | ALL [ PRIVILEGES ] }
    ON PARAMETER configuration_parameter [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { { CREATE | USAGE } [, ...] | ALL [ PRIVILEGES ] }
    ON SCHEMA schema_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { CREATE | ALL [ PRIVILEGES ] }
    ON TABLESPACE tablespace_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT { USAGE | ALL [ PRIVILEGES ] }
    ON TYPE type_name [, ...]
    TO role_specification [, ...] [ WITH GRANT OPTION ]
    [ GRANTED BY role_specification ]

GRANT role_name [, ...] TO role_specification [, ...]
    [ WITH { ADMIN | INHERIT | SET } { OPTION | TRUE | FALSE } ]
    [ GRANTED BY role_specification ]


###role_specification#########

    [ GROUP ] role_name
  | PUBLIC
  | CURRENT_ROLE
  | CURRENT_USER
  | SESSION_USER
```

##

GRANT######2########## 1###1############# ##### ##

################GRANT

#####GRANT#######################1############## ############################## #####

PUBLIC################################### PUBLIC########################### ########### ##PUBLIC########### ############

WITH GRANT OPTION#################################### ###################### ########### ##########PUBLIC##############

GRANTED BY################################### ####SQL###################### ##

################################

######################
################# ## #####
######################################

###############

SELECT
INSERT
UPDATE
DELETE
TRUNCATE
REFERENCES
TRIGGER
CREATE
CONNECT
TEMPORARY
EXECUTE
USAGE
SET
ALTER SYSTEM
MAINTAIN

##########5.8#########

TEMP

TEMPORARY########

ALL PRIVILEGES

########################## PRIVILEGES######PostgreSQL############SQL####
###

FUNCTION##PROCEDURE###
######################################ROUTINE########

1## ##############################
######### ALL TABLES#######GRANT######################### ALL FUNCTIONS##
##GRANT######### ###########ALL
ROUTINES########

#######GRANT

#####GRANT################1#######################SET#INHERIT#ADMIN#####
######21.3######### ##
######################## ## ##########
######################################

##########TRUE###FALSE############ #####OPTION#TRUE#################WITH
ADMIN OPTION#WITH ADMIN TRUE###### ####################################

ADMIN## #########
################################## ###########WITH ADMIN OPTION######
######### ### #############
#FALSE###

INHERIT#################################21.3######### TRUE################
############ FALSE################### ####################################
#######

SET#####TRUE##########SET ROLE############################## ##############
###########SET TRUE####################SET ROLE################## ########
#####TRUE###

###SET ROLE#######
####### ########ALTER ... OWNER TO#CREATE DATABASE ... OWNER#############
#############################SET ROLE#######################################
##########################(##################################)# ###########
########SET ROLE#######################SQL##################

GRANTED BY################################## ##############################
########## ##########################ADMIN OPTION###########################
############# ##ADMIN OPTION####
######################ADMIN OPTION##############################

####################PUBLIC############# #############*role_specification*##
###GROUP######################

#############REVOKE###########

PostgreSQL 8.1## #########################
######GROUP#################### ##########GROUP######################

#############################SELECT#INSERT############## ##################
########################### ###############################

##GRANT#####################
########
#########GRANT ALL PRIVILEGES############### ############################
############################## ##
##

##########
###Unix#######root######## root##

########GRANT#REVOKE### #
############################
####################

GRANT###REVOKE#################################### ########################
#######WITH GRANT OPTION##################### ###########################
###WITH GRANT OPTION########################## ########t1####g1##########
#u1####g1########### #####u1#t1######u2####### ##########g1###############
#####g1######################

GRANT## ############################
######## #######SET ROLE######GRANT######################

##############SERIAL###
########################

##################################5.8#########

####films######################

```
GRANT INSERT ON films TO PUBLIC;
```

###kinds##################manuel######

```
GRANT ALL PRIVILEGES ON kinds TO manuel;
```

###############kinds###
####################

###admins##########joe######

```
GRANT admins TO joe;
```

###

##SQL###ALL PRIVILEGES##PRIVILEGES########## ##SQL###1##################
#################

PostgreSQL################################## ##############
#INSERT#UPDATE#DELETE#TRUNCATE##
######SQL######## PostgreSQL##
################## ##SQL####################_SYSTEM##################### ###
##_SYSTEM#########################

##SQL################PUBLIC############## PostgreSQL#######################
###########

##SQL###GRANTED BY############CURRENT_USER#CURRENT_ROLE##### ###### PostgreSQL######

##SQL#####################################USAGE##############

##SQL########USAGE######### ###PostgreSQL####nextval######NEXT VALUE FOR############# #########SELECT###UPDATE###PostgreSQL###### #########USAGE##
#currval##########PostgreSQL#####(###########)#

####################################PostgreSQL######

####

REVOKE, ALTER DEFAULT PRIVILEGES

IMPORT FOREIGN SCHEMA

IMPORT FOREIGN SCHEMA — ####################

##

```
IMPORT FOREIGN SCHEMA remote_schema
    [ { LIMIT TO | EXCEPT } ( table_name [, ...] ) ]
    FROM SERVER server_name
    INTO local_schema
    [ OPTIONS ( option 'value' [, ... ] ) ]
```

##

IMPORT FOREIGN SCHEMA############################## ######################
######################################

############################
######################## ##

IMPORT FOREIGN SCHEMA#############USAGE##################CREATE#######

#####

remote_schema

#################### ####################################

LIMIT TO (*table_name* [, ...])

##########1######################### ########################

EXCEPT (*table_name* [, ...])

######################## ##

server_name

################

local_schema

##########################

OPTIONS (*option* '*value*' [, ...])

################### ################################

#

###film_server###########foreign_films#########################films####
###########

```
IMPORT FOREIGN SCHEMA foreign_films
    FROM SERVER film_server INTO films;
```

#######2######actors#directors#######################

```
IMPORT FOREIGN SCHEMA foreign_films LIMIT TO (actors, directors)
    FROM SERVER film_server INTO films;
```

###

IMPORT FOREIGN SCHEMA######OPTIONS##PostgreSQL################SQL#########

####

CREATE FOREIGN TABLE, CREATE SERVER

INSERT

INSERT — ##############

##

```
[ WITH [ RECURSIVE ] with_query [, ...] ]
INSERT INTO table_name [ AS alias ] [ ( column_name [, ...] ) ]
    [ OVERRIDING { SYSTEM | USER } VALUE ]
    { DEFAULT VALUES | VALUES ( { expression | DEFAULT } [, ...] ) [, ...]
 | query }
    [ ON CONFLICT [ conflict_target ] conflict_action ]
    [ RETURNING { * | output_expression [ [ AS ] output_name ] } [, ...] ]


###conflict_target###########

    ( { index_column_name | ( index_expression ) } [ COLLATE collation ]
 [ opclass ] [, ...] ) [ WHERE index_predicate ]
    ON CONSTRAINT constraint_name


##conflict_action###########

    DO NOTHING
    DO UPDATE SET { column_name = { expression | DEFAULT } |
                    ( column_name [, ...] ) = [ ROW ] ( { expression |
 DEFAULT } [, ...] ) |
                    ( column_name [, ...] ) = ( sub-SELECT )
                  } [, ...]
              [ WHERE condition ]
```

##

INSERT################ ####################################0################

#################### ## #
##VALUES##query#N################N################### VALUES##query#######
################################

###NULL#########

################################

################INSERT########################### ###########################
###62.5######## ON CONFLICT#
####ON CONFLICT#########
###

RETURNING########INSERT##############ON CONFLICT DO UPDATE##############
################### ### ################
################# RETURNING#######SELECT############ ##################### ##
############ON CONFLICT DO UPDATE ... WHERE## condition###############
############

#######################INSERT################ ON CONFLICT DO UPDATE#######
#####UPDATE########

######################INSERT######### ####ON CONFLICT DO UPDATE###### ########################UPDATE####### ####ON CONFLICT DO UPDATE####ON CONFLICT DO UPDATE#####*condition*###############SELECT#######

RETURNING########RETURNING#############SELECT####### *query*############ ##SELECT###############

#####

##

#################################### ON CONFLICT########################### ##

with_query

WITH#####INSERT## ### #7.8#SELECT#########

query#SELECT######WITH############ ######*with_query*#######*query*#### ######### ################################

table_name

#######################

alias

table_name###### alias######################### ####excluded######### #ON CONFLICT DO UPDATE## #################

column_name

table_name################ ################################## ######### ####################NULL###### ON CONFLICT DO UPDATE################ ############### ####INSERT INTO table_name ... ON CONFLICT DO UPDATE SET table_name.col = 1########UPDATE#############

OVERRIDING SYSTEM VALUE

###########IDENTITY##################################

GENERATED ALWAYS#######IDENTITY######OVERRIDING SYSTEM VALUE#OVERRIDING USER VALUE#####(DEFAULT###)################ (GENERATED BY DEFAULT#######IDENTITY######OVERRIDING SYSTEM VALUE# ##########################PostgreSQL###########)

OVERRIDING USER VALUE

###########IDENTITY##

########################## INSERT INTO tbl2 OVERRIDING USER VALUE SELECT * FROM tbl1####tbl1###tbl2#IDENTITY##################tbl2# IDENTITY#####tbl2#####################

DEFAULT VALUES

######DEFAULT## #OVERRIDING######## ########

expression

#####################

DEFAULT

################## IDENTITY############################### ###############
##

query

###############SELECT######## #########SELECT##########

output_expression

##############INSERT############### #####*table_name*################### ####### ######################*#########

output_name

###############

ON CONFLICT#

######ON CONFLICT### ################
################*conflict_target*############################
#*conflict_action*############### ON CONFLICT DO NOTHING################## ######## ON CONFLICT DO UPDATE####################################

conflict_target######################## ############1### #*index_column_name*#####*index_expression*################## #*index_predicate*######## *table_name*#########*conflict_target*########### ## *index_predicate*############### ################################## ## ########################ON CONFLICT######################## ################ ###

ON CONFLICT DO UPDATE#INSERT###UPDATE############# ##################### ################2#######1###### ###*UPSERT*#####UPDATE or INSERT############

conflict_target

ON CONFLICT################################### ########################### ############### ON CONFLICT DO NOTHING##*conflict_target*########## #### ## ON CONFLICT DO UPDATE# #*conflict_target*############

conflict_action

conflict_action##ON CONFLICT############ ###DO NOTHING####DO UPDATE# ################################UPDATE############### ON CONFLICT DO UPDATE#SET##WHERE## #excluded################ excluded#########################SELECT###### ##

#########BEFORE INSERT#######excluded################## ############### ######################

index_column_name

table_name######## ###################### CREATE INDEX######### *index_column_name*#SELECT######

index_expression

index_column_name####################*table_name*#####################
CREATE INDEX######## *index_expression*#########SELECT#######

collation

#################*index_column_name*####*index_expression*##############
############### ##################################### CREATE INDEX#####
####

opclass

#################*index_column_name*####*index_expression*##############
################# ##
################################### CREATE INDEX########

index_predicate

#################### ##
CREATE INDEX######## *index_predicate*############SELECT#######

constraint_name

##

condition

boolean######### ####true############# ####ON CONFLICT DO UPDATE######
################### *condition*##

#####ON CONFLICT DO UPDATE########################## ##########NOT DEFERRABLE################################

ON CONFLICT DO UPDATE####INSERT########### ######################2#######
##################### ################################### ###################
##################################

################INSERT#ON CONFLICT DO UPDATE############################
##

###
ON CONFLICT ON CONSTRAINT *constraint_name*#################### ################### ## ##################### CREATE UNIQUE INDEX ... CONCURRENTLY#######

##

########INSERT##################

INSERT *oid count*

count############### *oid*###0##(*count*####1###########WITH OIDS############## ####OID########0##############WITH OIDS##########################)#

INSERT#####RETURNING############RETURNING##############################
###########SELECT################

##

```
##################################################  ##########################
############################
```

MERGE############INSERT#UPDATE#DELETE############# MERGE#########

# #

```
films#####1########
```

```
INSERT INTO films VALUES
    ('UA502', 'Bananas', 105, '1971-07-13', 'Comedy', '82 minutes');
```

```
######len######### ####################
```

```
INSERT INTO films (code, title, did, date_prod, kind)
    VALUES ('T_601', 'Yojimbo', 106, '1961-06-16', 'Drama');
```

```
######################DEFAULT#######
```

```
INSERT INTO films VALUES
    ('UA502', 'Bananas', 105, DEFAULT, 'Comedy', '82 minutes');
INSERT INTO films (code, title, did, date_prod, kind)
    VALUES ('T_601', 'Yojimbo', 106, DEFAULT, 'Drama');
```

```
###################
```

```
INSERT INTO films DEFAULT VALUES;
```

```
####VALUES################
```

```
INSERT INTO films (code, title, did, date_prod, kind) VALUES
    ('B6717', 'Tampopo', 110, '1985-02-10', 'Comedy'),
    ('HG120', 'The Dinner Game', 140, DEFAULT, 'Comedy');
```

```
######films###############tmp_films######films################
```

```
INSERT INTO films SELECT * FROM tmp_films WHERE date_prod < '2004-05-07';
```

```
#################
```

```
-- ######3×3##############
INSERT INTO tictactoe (game, board[1:3][1:3])
    VALUES (1, '{{" "," "," "},{" "," "," "},{" "," "," "}}');
```

```
--###################
INSERT INTO tictactoe (game, board)
    VALUES (2, '{{X," "," "},{" ",O," "},{" ",X," "}}');
```

```
distributors#############DEFAULT####################
```

```
INSERT INTO distributors (did, dname) VALUES (DEFAULT, 'XYZ Widgets')
   RETURNING did;
```

Acme##

```
WITH upd AS (
  UPDATE employees SET sales_count = sales_count + 1 WHERE id =
    (SELECT sales_person FROM accounts WHERE name = 'Acme Corporation')
    RETURNING *
)
INSERT INTO employees_log SELECT *, current_timestamp FROM upd;
```

######(distributors)############## did################################## ##########
###############excluded#####################

```
INSERT INTO distributors (did, dname)
    VALUES (5, 'Gizmo Transglobal'), (6, 'Associated Computing, Inc')
    ON CONFLICT (did) DO UPDATE SET dname = EXCLUDED.dname;
```

################################before insert##################################
###did##############################

```
INSERT INTO distributors (did, dname) VALUES (7, 'Redline GmbH')
    ON CONFLICT (did) DO NOTHING;
```

#################### ###did############################## ###############
#WHERE####################################

```
-- #########################
INSERT INTO distributors AS d (did, dname) VALUES (8, 'Anvil
 Distribution')
    ON CONFLICT (did) DO UPDATE
    SET dname = EXCLUDED.dname || ' (formerly ' || d.dname || ')'
    WHERE d.zipcode <> '21201';
```

```
-- ##############DO NOTHING#############
-- #################
INSERT INTO distributors (did, dname) VALUES (9, 'Antwerp Design')
    ON CONFLICT ON CONSTRAINT distributors_pkey DO NOTHING;
```

##########################DO NOTHING##### ######is_active########true#####
######did##########################

```
-- ####"WHERE is_active"##################
-- ##########"did"###################
INSERT INTO distributors (did, dname) VALUES (10, 'Conrad International')
    ON CONFLICT (did) WHERE is_active DO NOTHING;
```

###

INSERT###SQL###### ####RETURNING##INSERT#WITH########ON CONFLICT######
#######PostgreSQL###### #####SQL################VALUES####*query*############
######## ON CONFLICT#####SQL################MERGE#########

##SQL##########IDENTITY##########OVERRIDING SYSTEM VALUE############
PostgreSQL#####################################

query#########SELECT##########

LISTEN

LISTEN — #######

##

```
LISTEN channel
```

##

LISTEN################channel############ ################################
#############

##################################NOTIFY channel##########################
############### ####################################

UNLISTEN################################## ################################
######

##################################PostgreSQL########################### libpq#
###################LISTEN####SQL##############PQnotifies################
######################## libpgtcl### #
##libpgtcl#####################LISTEN#UNLISTEN################ ##############
####################

#####

channel

###################

##

LISTEN####################### LISTEN###UNLISTEN###########################
########################

LISTEN##################################

############################## ##
##################### ## #
####LISTEN##################
############(##########)##
################################## ##
##################

NOTIFY###LISTEN###NOTIFY#################

#

psql#####/################

```
LISTEN virtual;
NOTIFY virtual;
```

```
Asynchronous notification "virtual" received from server process with PID
 8448.
```

###

##SQL#LISTEN#######

####

NOTIFY, UNLISTEN, max_notify_queue_pages

LOAD

LOAD — ##################

##

```
LOAD 'filename'
```

##

LOAD#################PostgreSQL################# ############################## ### C####################1########################## ###########LOAD######## ##

##################################dynamic_library_path######################### #### ###################### ## ###### ########36.10.1##########

#######$libdir/plugins/##############LOAD########## ########*filename*##### ################ ##

###

LOAD#PostgreSQL######

####

CREATE FUNCTION

LOCK

LOCK — #########

##

```
LOCK [ TABLE ] [ ONLY ] name [ * ] [, ...] [ IN lockmode MODE ] [ NOWAIT ]
```

lockmode###############

```
    ACCESS SHARE | ROW SHARE | ROW EXCLUSIVE | SHARE UPDATE EXCLUSIVE
    | SHARE | SHARE ROW EXCLUSIVE | EXCLUSIVE | ACCESS EXCLUSIVE
```

##

LOCK TABLE## NOWAIT#########LOCK TABLE## ################ ################### #UNLOCK TABLE############## ############################

###

################################PostgreSQL######################### LOCK TABLE# ############################ ########READ COMMITTED######################## ### ####################SHARE########### ## ## ####SHARE## ##############ROW EXCLUSIVE##########LOCK TABLE *name* IN SHARE MODE# ##ROW EXCLUSIVE### ##

REPEATABLE READ###SERIALIZABLE################################SELECT# ################LOCK TABLE############# REPEATABLE READ## #SERIALIZABLE#########################SELECT######################### #### ########LOCK TABLE## ###########

##############################SHARE######SHARE ROW EXCLUSIVE############ ###### ## SHARE ROW EXCLUSIVE######### ################# 2##############SHARE#####################ROW EXCLUSIVE## ###################### ######################################SHARE########### #ROW EXCLUSIVE############ ##############SHARE############ROW EXCLUSIVE# ############# ## ###1###

############################13.3#########

#####

name

```
    ############################ #######ONLY######################### ONLY####
    ################################### ###############*#####################
    ###############
```

```
    LOCK a, b;#######LOCK TABLE a; LOCK TABLE b;###### #####1####LOCK##
    ###############
```

lockmode

    ############################### ############13.3#######

    #####################ACCESS EXCLUSIVE#######

NOWAIT

    LOCK TABLE########################### ####################################

##

#####[87.014925]#####################*lockmode*################################ ################# ############UPDATE#DELETE###TRUNCATE############ #*lockmode*######## ############INSERT##########ROW EXCLUSIVE MODE(## #13.3##################)######## ############SELECT##########ACCESS SHARE MODE######## #####################*lockmode*############## ########## #MAINTAIN#UPDATE#DELETE####TRUNCATE##########*lockmode*########## ######## ####INSERT########ROW EXCLUSIVE MODE####13.3######################### ###### ######SELECT########ACCESS SHARE MODE########

#security_invoker#true#########(CREATE VIEW########)###################

LOCK TABLE########################## ##################### #####PostgreSQL #LOCK################################## ####################BEGIN###COMMIT## ##ROLLBACK##########

LOCK#################### ##########ROW################ ######################## ############################ ###ROW EXCLUSIVE################## LOCK TABLE# ## ########## #############SELECT####13.3.2################

# #

##########################SHARE#########

```
BEGIN WORK;
LOCK TABLE films IN SHARE MODE;
SELECT id FROM films
    WHERE name = 'Star Wars: Episode I - The Phantom Menace';

-- #########ROLLBACK#######
INSERT INTO films_user_comments VALUES
    (_id_, 'GREAT! I was waiting for it for so long!');
COMMIT WORK;
```

################SHARE ROW EXCLUSIVE#########

```
BEGIN WORK;
LOCK TABLE films IN SHARE ROW EXCLUSIVE MODE;
DELETE FROM films_user_comments WHERE id IN
    (SELECT id FROM films WHERE rating < 5);
DELETE FROM films WHERE rating < 5;
COMMIT WORK;
```

###

##SQL##LOCK TABLE###### ##########################SET TRANSACTION######## PostgreSQL#####################SET TRANSACTION##########

ACCESS SHARE#ACCESS EXCLUSIVE#SHARE UPDATE EXCLUSIVE##########PostgreSQL ########LOCK TABLE###Oracle#############

MERGE

MERGE — ############INSERT#UPDATE#DELETE##

##

```
[ WITH with_query [, ...] ]
MERGE INTO [ ONLY ] target_table_name [ * ] [ [ AS ] target_alias ]
USING data_source ON join_condition
when_clause [...]
[ RETURNING { * | output_expression [ [ AS ] output_name ] } [, ...] ]

where data_source is:

{ [ ONLY ] source_table_name [ * ] | ( source_query ) }
 [ [ AS ] source_alias ]

and when_clause is:

{ WHEN MATCHED [ AND condition ] THEN { merge_update | merge_delete | DO
 NOTHING } |
  WHEN NOT MATCHED BY SOURCE [ AND condition ] THEN { merge_update
 | merge_delete | DO NOTHING } |
  WHEN NOT MATCHED [ BY TARGET ] [ AND condition ] THEN { merge_insert |
 DO NOTHING } }

and merge_insert is:

INSERT [( column_name [, ...] )]
[ OVERRIDING { SYSTEM | USER } VALUE ]
{ VALUES ( { expression | DEFAULT } [, ...] ) | DEFAULT VALUES }

and merge_update is:

UPDATE SET { column_name = { expression | DEFAULT } |
             ( column_name [, ...] ) = [ ROW ] ( { expression | DEFAULT }
 [, ...] ) |
             ( column_name [, ...] ) = ( sub-SELECT )
           } [, ...]

and merge_delete is:

DELETE
```

##

MERGE##*data_source*#####*target_table_name*############################
MERGE########INSERT#UPDATE###DELETE########SQL####### ################
#######

#########MERGE#####*data_source*################################### #########
##MATCHED#NOT MATCHED BY SOURCE####NOT MATCHED [BY TARGET]############
############WHEN######### ############################### ###############WHEN##
1######

MERGE##############UPDATE#INSERT###DELETE############## ##############
##WHERE#################### ##
##

DO NOTHING######################### #####################DO NOTHING######## ###########################

###########RETURNING################################MERGE####### ######### ###############merge_action()############### INSERT###UPDATE############ ###################### DELETE########################## RETURNING###### ##SELECT###########

###MERGE######## ##############SET##################UPDATE####### ###### ##################INSERT####### ########################DELETE####### DO NOTHING#########################1#######SELECT####### ###*data_source*##### #condition#join_condition#######expression####################SELECT### ##### ####################WHEN####################

###MERGE###########

#####

with_query

WITH########MERGE#############1################# ###7.8#SELECT######### WITH RECURSIVE#MERGE######################

target_table_name

#####[88.925081]####################(########)# ########ONLY################ ################### ONLY## ##### ##########*################################## ONLY########*################ #### ######################## ######################## ########### ###### ##ONLY#################################### ONLY########################### ######################## #######*### ONLY######*######################################

######*target_table_name*########INSTEAD OF#####################WHEN#### #############INSERT#UPDATE#DELETE#####INSTEAD OF############## ######### ############

target_alias

############# ############################### #####MERGE INTO foo AS f### #####MERGE###############foo####f##############

source_table_name

#########################(########)# ########ONLY############################ #### ONLY## ###############*####### ###########################

source_query

#########################SELECT####VALUES##### #######SELECT####VALUES## #########

source_alias

############# ##

join_condition

join_condition#boolean##########WHERE##############*data_source*####### ###################

##

join_condition###*data_source*########################## ######## ######*join_condition*##

######WHEN NOT MATCHED BY SOURCE#WHEN NOT MATCHED [BY TARGET]#############MERGE#####*data_source*############FULL# ####### ###############1##*join_condition*######################### ####################################

when_clause

#####1##WHEN######

######WHEN##WHEN MATCHED#WHEN NOT MATCHED BY SOURCE####WHEN NOT MATCHED [BY TARGET]#######SQL#####WHEN MATCHED#WHEN NOT MATCHED#### ######################################WHEN NOT MATCHED BY SOURCE#WHEN NOT MATCHED#BY TARGET#############SQL#######################

######WHEN##WHEN MATCHED##########*data_source*################## ##WHEN##*condition*#######true###############

######WHEN##WHEN NOT MATCHED BY SOURCE##########*data_source*######## ############WHEN##*condition*#######true###############

######WHEN##WHEN NOT MATCHED [BY TARGET]##########*data_source*###### ####################WHEN##*condition*#######true###############

condition

boolean#######WHEN####################################

#####[61.710037]#WHEN MATCHED#######################################WHEN NOT MATCHED######################### ######################## ######## ################### ######WHEN MATCHED#################################### WHEN NOT MATCHED BY SOURCE### WHEN NOT MATCHED [BY TARGET]### #### ##########################

merge_insert

#######1########INSERT######## ################## ######################### ##########################

###NULL########

##############################

################## INSERT#######SELECT############

VALUES##1########## VALUES#################### ######################

merge_update

################UPDATE######## ###2##########

######WHERE#########

merge_delete

###############DELETE########### DELETE################################
##

column_name

######### ################################ ############################
NULL###### ########################

OVERRIDING SYSTEM VALUE

###########GENERATED ALWAYS#######ID#########(DEFAULT##)############
################

OVERRIDING USER VALUE

##########GENERATED BY DEFAULT#######ID##############################
######

DEFAULT VALUES

##################(########OVERRIDING########)#

expression

############## WHEN MATCHED##########################*data_source*#####
####### WHEN NOT MATCHED BY SOURCE####################################
WHEN NOT MATCHED [BY TARGET]##########*data_source*############

DEFAULT

#####################################NULL######

sub-SELECT

##SELECT###### ########## 1 #
############### ### 1 ################################ ###########NULL
#################### WHEN MATCHED################################
#*data_source*########### WHEN NOT MATCHED BY SOURCE################
##########################

output_expression

#########################MERGE################### ######################
#######merge_action()################################

######*## ##########################
###################### ######################################*######

output_name

#####################

##

#########MERGE######################

MERGE *total_count*

total_count############(###############)# *total_count*#0##################

######MERGE#####RETURNING#########RETURNING##############SELECT#######
##########################

##

########MERGE##########

1. WHEN################################BEFORE STATEMENT#########

2. ########################### ################################ ###########

   a. ########MATCHED#SOURCE####################[BY TARGET]############
      ##

   b. #########WHEN################

   c. ########################

      i. ####################BEFORE ROW#########

      ii. ################################

      iii. ####################AFTER ROW#########
      If the target relation is a view with `INSTEAD OF ROW` triggers for the action's event type, they are used to perform the action instead.

3. ####################################AFTER STATEMENT######### #########
   #UPDATE##########

#######################INSERT################# ########### #################
####################### #####MERGE######UPDATE##############
##UPDATE#INSERT#######################

##############1######################### ######################### ######
######1################# #################### ##########################
#MERGE########################## ###########INSERT################UPDATE##
#DELETE######################## #######SQL######### ####PostgreSQL#UPDATE##
#DELETE#################### #######2####################

#####[84.920635]#WHEN##AND#########################(MATCHED###NOT MATCHED)
######WHEN################################
################################ #####WHEN##AND##########################
#MATCHED#NOT MATCHED BY SOURCE####NOT MATCHED [BY TARGET]## ######WHEN#
################################## ##
###########

############################## *source_query*################################
#####################################

MERGE## ##############13.2#####
###INSERT ... ON CONFLICT################### ######INSERT#######
#UPDATE############## 2####################################

# #

##recent_transactions######customer_accounts#############

```
MERGE INTO customer_account ca
USING recent_transactions t
```

```
ON t.customer_id = ca.customer_id
WHEN MATCHED THEN
  UPDATE SET balance = balance + transaction_value
WHEN NOT MATCHED THEN
  INSERT (customer_id, balance)
  VALUES (t.customer_id, t.transaction_value);
```

MATCHED##

```
MERGE INTO customer_account ca
USING (SELECT customer_id, transaction_value FROM recent_transactions) AS
 t
ON t.customer_id = ca.customer_id
WHEN MATCHED THEN
  UPDATE SET balance = balance + transaction_value
WHEN NOT MATCHED THEN
  INSERT (customer_id, balance)
  VALUES (t.customer_id, t.transaction_value);
```

#####[80.176211]########################### ############################## ####0#### ######### ############################### ########################### ########## ###### ####################

```
MERGE INTO wines w
USING wine_stock_changes s
ON s.winename = w.winename
WHEN NOT MATCHED AND s.stock_delta > 0 THEN
  INSERT VALUES(s.winename, s.stock_delta)
WHEN MATCHED AND w.stock + s.stock_delta > 0 THEN
  UPDATE SET stock = w.stock + s.stock_delta
WHEN MATCHED THEN
  DELETE
RETURNING merge_action(), w.*;
```

wine_stock_changes#################################

###################wines###

```
MERGE INTO wines w
USING new_wine_list s
ON s.winename = w.winename
WHEN NOT MATCHED BY TARGET THEN
  INSERT VALUES(s.winename, s.stock)
WHEN MATCHED AND w.stock != s.stock THEN
  UPDATE SET stock = s.stock
WHEN NOT MATCHED BY SOURCE THEN
  DELETE;
```

###

##########SQL#########

######WHEN NOT MATCHED#DO NOTHING######RETURNING#####WITH##BY SOURCE## #BY TARGET#####SQL########

MOVE

MOVE — ##########

##

```
MOVE [ direction ] [ FROM | IN ] cursor_name
```

```
###direction#################

    NEXT
    PRIOR
    FIRST
    LAST
    ABSOLUTE count
    RELATIVE count
    count
    ALL
    FORWARD
    FORWARD count
    FORWARD ALL
    BACKWARD
    BACKWARD count
    BACKWARD ALL
```

##

MOVE######################### MOVE#FETCH#################################### #######

MOVE##########FETCH######### ##############FETCH#########

##

```
#########MOVE##################
```

```
MOVE count
```

count###########FETCH############# #####0#########

# #

```
BEGIN WORK;
DECLARE liahona CURSOR FOR SELECT * FROM films;
```

```
-- ###5##########
MOVE FORWARD 5 IN liahona;
MOVE 5
```

```
-- liahona######6#########
FETCH 1 FROM liahona;
 code  | title  | did | date_prod  |  kind  |  len
-------+--------+-----+------------+--------+-------
```

```
 P_303 | 48 Hrs | 103 | 1982-10-22 | Action | 01:37
(1 row)


-- ####liahona###################
CLOSE liahona;
COMMIT WORK;
```

###

##SQL##MOVE#######

####

CLOSE, DECLARE, FETCH

NOTIFY

NOTIFY — ######

##

```
NOTIFY channel [ , payload ]
```

##

NOTIFY##############################LISTEN *channel*#########################
#######(####)############## ################

NOTIFY####PostgreSQL## ##############
########### ##########################1#######################################
#################

##PID#########(##############)#
######

######################
################ ### ##
##NOTIFY#####LISTEN########################## ##############1###############
###################################### ######################################

######################NOTIFY#######NOTIFY############################## ###
###

NOTIFY#SQL############################## ###NOTIFY##########################
########################## ###################NOTIFY##########################
##################### ### ##############
###
######### ##
#################################### ##
######NOTIFY##

##1##################### ##############
##################### ##################1##################### ##############
##############NOTIFY#################################### ####################
##############################

NOTIFY### ##############################
################ ## ####################
################PID(###############
##)#######PID(libpq######)############################## PID###################
###########################

#####

channel

#############################

payload

######################## ############################## ###########8000######
########## (##
##)

##

###############NOTIFY#########
############## ############(#############8#####)########################### ###
######NOTIFY### ###################
#################################### ##
#####################

##pg_notification_queue_usage################################   #########
#9.27#########

NOTIFY##################################

pg_notify

###########pg_notify(text,text)############ ######1###########2###########
##############################NOTIFY########################

#

psql####/################

```
LISTEN virtual;
NOTIFY virtual;
Asynchronous notification "virtual" received from server process with PID
 8448.
NOTIFY virtual, 'This is the payload';
Asynchronous notification "virtual" with payload "This is the payload"
 received from server process with PID 8448.

LISTEN foo;
SELECT pg_notify('fo' || 'o', 'pay' || 'load');
Asynchronous notification "foo" with payload "payload" received from
 server process with PID 14728.
```

###

##SQL##NOTIFY######

####

LISTEN, UNLISTEN, max_notify_queue_pages

PREPARE

PREPARE — #########

##

```
PREPARE name [ ( data_type [, ...] ) ] AS statement
```

##

PREPARE############# ################################## PREPARE########### ###################### ####EXECUTE################################## ######### ##

##############$1#$2################ ########## ################################## ####################unknown## ## ########EXECUTE################### #### ###EXECUTE#########

################################## ######################### ################# ######### #######1## ######## ######################### ##################DEALLOCATE##########

##

#####

name

##################### ######1## ######

data_type

##################### #######################unknown################### ############ ########################$1#$2########

statement

###SELECT#INSERT#UPDATE#DELETE#MERGE#VALUES####

##

###################################### ###################################### ########################### ## ######################### (##)

#####(#####plan_cache_mode#auto#########)################################### ################# ###############5#################################### ### ## ################################### ############################

plan_cache_mode#force_generic_plan###force_custom_plan############### ## ############################### ##

##########PostgreSQL#####################EXPLAIN####

```
EXPLAIN EXECUTE name(parameter_values);
```

########## #################$n##

#######################PostgreSQL###############ANALYZE################

###PostgreSQL#########################
#####################DDL## ###
#######search_path###############search_path############ ########PostgreSQL
9.3########### ##
###############
###################search_path##
############### ##

pg_prepared_statements##

# #

INSERT####################

```
PREPARE fooplan (int, text, bool, numeric) AS
    INSERT INTO foo VALUES($1, $2, $3, $4);
EXECUTE fooplan(1, 'Hunter Valley', 't', 200.00);
```

SELECT####################

```
PREPARE usrrptplan (int) AS
    SELECT * FROM users u, logs l WHERE u.usrid=$1 AND u.usrid=l.usrid
    AND l.date = $2;
EXECUTE usrrptplan(1, current_date);
```

######2#################### ####$2#####################

###

##SQL##PREPARE###############SQL############# #####SQL#PREPARE##############
####

####

DEALLOCATE, EXECUTE

PREPARE TRANSACTION

PREPARE TRANSACTION — ######################

##

```
PREPARE TRANSACTION transaction_id
```

##

PREPARE TRANSACTION########################### ############################ ######## ## ########

#################COMMIT PREPARED###########ROLLBACK PREPARED########### ####### ##

###################PREPARE TRANSACTION#ROLLBACK################ ######### ## ################################

#######PREPARE TRANSACTION###########ROLLBACK####### ##################### ##

#####

transaction_id

##COMMIT PREPARED#ROLLBACK PREPARED######################## ######## #################200############### ##################################### ###########

##

PREPARE TRANSACTION################################ ##################### ## ########### ###################PREPARE TRANSACTION##############

################################## ##################BEGIN#########

###WITH HOLD########### ##LISTEN#UNLISTEN###NOTIFY#############PREPARE########### ############### ##

######################LOCAL########SET###############PREPARE TRANSACTION####### ######COMMIT PREPARED#ROLLBACK PREPARED######### ## ##########PREPARE TRANSACTION #ROLLBACK##COMMIT#############

###########################pg_prepared_xacts##############

##
################################## ###########VACUUM################# ########24.1.5########################### ############################## ############## ## ##

##
##max_prepared_transactions#################################### ###########
##

# #

########################## ##############foobar#######

```
PREPARE TRANSACTION 'foobar';
```

###

PREPARE TRANSACTION#PostgreSQL###### ################################## ##### #############X/Open XA#######################SQL#############

####

COMMIT PREPARED, ROLLBACK PREPARED

REASSIGN OWNED

REASSIGN OWNED — ####################################

##

```
REASSIGN OWNED BY { old_role | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER } [, ...]
               TO { new_role | CURRENT_ROLE | CURRENT_USER |
 SESSION_USER }
```

##

REASSIGN OWNED##*old_roles*#########################*new_role*##############
####

#####

old_role

####### ##
#*new_role*#######

new_role

###########################

##

REASSIGN OWNED##1#################### REASSIGN OWNED####################
##

REASSIGN OWNED#########################

####DROP OWNED#########1##

REASSIGN OWNED######*old_roles*####################*old_roles*##############
####ALTER DEFAULT PRIVILEGES#################### ##############DROP
OWNED#########

####21.4##########

###

REASSIGN OWNED#####PostgreSQL######

####

DROP OWNED, DROP ROLE, ALTER DATABASE

REFRESH MATERIALIZED VIEW

REFRESH MATERIALIZED VIEW — ##################

##

```
REFRESH MATERIALIZED VIEW [ CONCURRENTLY ] name
    [ WITH [ NO ] DATA ]
```

##

#####[87.985213]#REFRESH MATERIALIZED VIEW######################### ######## ################################ ############ WITH DATA#########(#########)### ### WITH NO DATA############ ##################################### ######REFRESH MATERIALIZED VIEW####### ################### ################### (Materialized View) ####MAINTAIN####### # ########### WITH DATA### ######## ###################### WITH NO DATA### #####

CONCURRENTLY#WITH NO DATA################

#####

CONCURRENTLY

#########################SELECT######################REFRESH#### ######## ##################REFRESH## ################ ################################

#####################################UNIQUE############1############ ##### ###############WHERE#############

#####[72.727273]## ################ ################################

###############1#######################REFRESH########

name

#############################

##

######################ORDER BY#####################################REFRESH MATERIALIZED VIEW####################

######REFRESH MATERIALIZED VIEW#####search_path#####pg_catalog, pg_temp## ######

#

##################################order_summary############################## ######

```
REFRESH MATERIALIZED VIEW order_summary;
```

```
##################annual_statistics_basis##########################

REFRESH MATERIALIZED VIEW annual_statistics_basis WITH NO DATA;
```

###

```
REFRESH MATERIALIZED VIEW#PostgreSQL######
```

####

CREATE MATERIALIZED VIEW, ALTER MATERIALIZED VIEW, DROP MATERIALIZED VIEW

REINDEX

REINDEX — ###########

##

```
REINDEX [ ( option [, ...] ) ] { INDEX | TABLE | SCHEMA }
 [ CONCURRENTLY ] name
REINDEX [ ( option [, ...] ) ] { DATABASE | SYSTEM } [ CONCURRENTLY ]
 [ name ]
```

###option#########

```
    CONCURRENTLY [ boolean ]
    TABLESPACE new_tablespace
    VERBOSE [ boolean ]
```

##

REINDEX## ###REINDEX# ############

- ################################ ## ################## REINDEX#############
- ## ######PostgreSQL#B-tree## ################################## REINDEX################################ ########################### ###24.2##########
- ##
- CONCURRENTLY## ################ ###REINDEX#################### REINDEX INDEX############################ #######

#####

INDEX

  ################## #####REINDEX## ##

TABLE

  ######################## #####2####TOAST############################### ### ##REINDEX##

SCHEMA

  ########################### ################TOAST######################### ###### ######################## #####REINDEX#####################

DATABASE

  ## ######################## #### #REINDEX######################

SYSTEM

###################################### ###################### ############ ######### #####REINDEX#####################

name

################################## ######################## #####REINDEX DATABASE#REINDEX SYSTEM############################## ############### ####################

CONCURRENTLY

#############PostgreSQL## ######################################(###########)# ##################### #######—#######################

###########REINDEX## #####

TABLESPACE

##############################

VERBOSE

##############################

boolean

########################### ############TRUE#ON###1########FALSE#OFF## #0########## *boolean*################TRUE#######

new_tablespace

####################

##

############################REINDEX INDEX##########################REINDEX TABLE##################################

###################################### ## ####### ## #### ################################-P#########################

#######1############################-P##################PostgreSQL####### ## ###############REINDEX DATABASE#REINDEX SYSTEM#REINDEX TABLE### ##REINDEX INDEX########## #########REINDEX SYSTEM###################### ############## ## ##################### ###########postgres################

###########-P########################### ######################## ####libpq#### ##########################PGOPTIONS#-P############ ######################### ##

REINDEX###########1###################################### ############### ## REINDEX## ################ACCESS EXCLUSIVE#################################### ############################ #######ACCESS SHARE###############REINDEX############################

################################## ###DROP INDEX#############ACCESS EXCLUSIVE############################ ######CREATE INDEX################## ######## ################################### ################################ ################################

######REINDEX#####search_path#####pg_catalog, pg_temp#######

##MAINTAIN####### ################# #########REINDEX####################MAINTAIN################## ######## ################################## ## ###########pg_maintain########### ## ###################### ########################MAINTAIN###########REINDEX DATABASE#REINDEX SCHEMA####REINDEX SYSTEM######################

##REINDEX INDEX#REINDEX TABLE###### ##### ## ################ ##

#########################TABLESPACE#####REINDEX############################ ###### ##ALTER TABLE ONLY############ ## ######### ##

SCHEMA#DATABASE#SYSTEM#TABLESPACE#########################WARNING#1### ## TOAST#####################################

##############

################################## ###PostgreSQL################################ ######################### ## ################# ### ################ ## ###################

PostgreSQL################################## REINDEX#CONCURRENTLY########### ########## ###########PostgreSQL######################2#################### ### ######################### ## # ### #################CPU ##

############################ ######################### ########################### ###########################

1. ####pg_index######################## ########################### ########### ################################SHARE UPDATE EXCLUSIVE################# ############

2. #########1########################## ################################## #pg_index.indisready##true## ##### ##################################

3. ####1##################2######### ##################################

4. ## #### ##pg_index.indisvalid###############true##################false######### ##

5. ## #pg_index.indisready##false##########

6. ############### ##############SHARE UPDATE EXCLUSIVE###############

##REINDEX############################
################ ##
psql \d################INVALID#######

```
postgres=# \d tab
       Table "public.tab"
 Column |  Type   | Modifiers
--------+---------+-----------
 col    | integer |
Indexes:
    "idx" btree (col)
    "idx_ccnew" btree (col) INVALID
```

INVALID################ccnew##
#DROP INDEX#############REINDEX CONCURRENTLY######## #################
#ccold################################ ####################################
#######

###1#######1############ ####
##################################### ############REINDEX   TABLE#REINDEX
INDEX############################REINDEX CONCURRENTLY#########

##############################REINDEX###################VACUUM##############
#######

############################REINDEX SYSTEM#CONCURRENTLY#########

#################################### ## ##
############
#CONCURRENTLY########################

REINDEX###################pg_stat_progress_create_index#########   ##
#27.4.4#########

#

#################

```
REINDEX INDEX my_index;
```

####my_table###################

```
REINDEX TABLE my_table;
```

###

```
$ export PGOPTIONS="-P"
$ psql broken_db
...
broken_db=> REINDEX DATABASE broken_db;
broken_db=> \q
```

##

```
REINDEX TABLE CONCURRENTLY my_broken_table;
```

###

##SQL##REINDEX#######

####

CREATE INDEX, DROP INDEX, reindexdb, 27.4.4

RELEASE SAVEPOINT

RELEASE SAVEPOINT — ################

##

```
RELEASE [ SAVEPOINT ] savepoint_name
```

##

RELEASE SAVEPOINT##RELEASE SAVEPOINT##

#####

savepoint_name

################

##

##############################

###ROLLBACK TO SAVEPOINT#########################

###

#

#####################

```
BEGIN;
    INSERT INTO table1 VALUES (3);
    SAVEPOINT my_savepoint;
    INSERT INTO table1 VALUES (4);
    RELEASE SAVEPOINT my_savepoint;
COMMIT;
```

##############3#4###########

##############################

```
BEGIN;
    INSERT INTO table1 VALUES (1);
    SAVEPOINT sp1;
    INSERT INTO table1 VALUES (2);
    SAVEPOINT sp2;
    INSERT INTO table1 VALUES (3);
    RELEASE SAVEPOINT sp2;

    INSERT INTO table1 VALUES (4))); -- ######
```

#####################sp2###########3######### ######################sp1#### #### #4###2#4##########3############### ####### ################################2########1###########

```
ROLLBACK;
ROLLBACK TO SAVEPOINT sp1;
```

ROLLBACK#######1#####################ROLLBACK TO SAVEPOINT sp1#######1# ########################

###

#########SQL######## SQL:2003#####SAVEPOINT############### PostgreSQL# #SAVEPOINT#################

####

BEGIN, COMMIT, ROLLBACK, ROLLBACK TO SAVEPOINT, SAVEPOINT

RESET

RESET — ##################

##

```
RESET configuration_parameter
RESET ALL
```

##

RESET##################### RESET################

```
SET configuration_parameter TO DEFAULT
```

###SET##########

#################SET################################## ######################
######################
############### ### ####19
##########

RESET##############SET###### ################################

#####

configuration_parameter

##################### ################19######## ###SET##################

ALL

############################

#

timezone#################

```
RESET timezone;
```

###

RESET#PostgreSQL######

####

SET, SHOW

REVOKE

REVOKE — ###########

##

```
REVOKE [ GRANT OPTION FOR ]
    { { SELECT | INSERT | UPDATE | DELETE | TRUNCATE | REFERENCES |
 TRIGGER | MAINTAIN }
    [, ...] | ALL [ PRIVILEGES ] }
    ON { [ TABLE ] table_name [, ...]
         | ALL TABLES IN SCHEMA schema_name [, ...] }
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { SELECT | INSERT | UPDATE | REFERENCES } ( column_name [, ...] )
    [, ...] | ALL [ PRIVILEGES ] ( column_name [, ...] ) }
    ON [ TABLE ] table_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { USAGE | SELECT | UPDATE }
    [, ...] | ALL [ PRIVILEGES ] }
    ON { SEQUENCE sequence_name [, ...]
         | ALL SEQUENCES IN SCHEMA schema_name [, ...] }
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { CREATE | CONNECT | TEMPORARY | TEMP } [, ...] | ALL
 [ PRIVILEGES ] }
    ON DATABASE database_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { USAGE | ALL [ PRIVILEGES ] }
    ON DOMAIN domain_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { USAGE | ALL [ PRIVILEGES ] }
    ON FOREIGN DATA WRAPPER fdw_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { USAGE | ALL [ PRIVILEGES ] }
    ON FOREIGN SERVER server_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
```

```
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { EXECUTE | ALL [ PRIVILEGES ] }
    ON { { FUNCTION | PROCEDURE | ROUTINE } function_name [ ( [ [ argmode
 ] [ arg_name ] arg_type [, ...] ] ) ] [, ...]
         | ALL { FUNCTIONS | PROCEDURES | ROUTINES } IN SCHEMA schema_name
 [, ...] }
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { USAGE | ALL [ PRIVILEGES ] }
    ON LANGUAGE lang_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { SELECT | UPDATE } [, ...] | ALL [ PRIVILEGES ] }
    ON LARGE OBJECT loid [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { SET | ALTER SYSTEM } [, ...] | ALL [ PRIVILEGES ] }
    ON PARAMETER configuration_parameter [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { { CREATE | USAGE } [, ...] | ALL [ PRIVILEGES ] }
    ON SCHEMA schema_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { CREATE | ALL [ PRIVILEGES ] }
    ON TABLESPACE tablespace_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ GRANT OPTION FOR ]
    { USAGE | ALL [ PRIVILEGES ] }
    ON TYPE type_name [, ...]
    FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]

REVOKE [ { ADMIN | INHERIT | SET } OPTION FOR ]
    role_name [, ...] FROM role_specification [, ...]
    [ GRANTED BY role_specification ]
    [ CASCADE | RESTRICT ]


###role_specification#########

    [ GROUP ] role_name
  | PUBLIC
```

```
  | CURRENT_ROLE
  | CURRENT_USER
  | SESSION_USER
```

##

REVOKE######1########################### PUBLIC#############################

############GRANT################

##PUBLIC##########3################
########## #########PUBLIC##SELECT################################SELECT#
############## ######################################SELECT######### #######
#SELECT#######PUBLIC################SELECT#######SELECT###########

GRANT OPTION FOR### ################
################

########
######################################CASCADE##############################
############## ######################################REVOKE##################
####### ##REVOKE######################
###

########################
######################

##########################GRANT OPTION#ADMIN OPTION###### PostgreSQL 16####
#####################################CASCADE############################## ##
############ ##########*role_specification*#####GROUP####################
##

ADMIN OPTION####################INHERIT OPTION###SET OPTION############
###############FALSE############

##

############################## #########A################B##############B####
C################A####C################## ########A####B##########CASCADE#
#################C#################### ###########A#B########C######A#A##
#################B################### ######C######################

############################REVOKE##
REVOKE ALL
PRIVILEGES################################ ##################################
##################### ##
######################

#######GRANT#REVOKE## ####
####################GRANT## #####
###
########### ###########CASCADE#################

REVOKE#################################### ################################
#WITH GRANT OPTION##################### ##### ####################WITH
GRANT OPTION################################### ####t1#####g1#########
##u1#g1############## #####u1#g1############################## #####u1######
#g1##########################
```
```

REVOKE###
#######SET ROLE######REVOKE######################
##########################

##################################5.8##########

# #

####films####public###############

```
REVOKE INSERT ON films FROM PUBLIC;
```

###kinds######manuel################

```
REVOKE ALL PRIVILEGES ON kinds FROM manuel;
```

##################################

###joe#####admins##############

```
REVOKE admins FROM joe;
```

###

GRANT###############REVOKE############ ##########RESTRICT#CASCADE#########
####PostgreSQL########RESTRICT########

####

GRANT, ALTER DEFAULT PRIVILEGES

ROLLBACK

ROLLBACK — ################

##

```
ROLLBACK [ WORK | TRANSACTION ] [ AND [ NO ] CHAIN ]
```

##

ROLLBACK##

#####

WORK
TRANSACTION

############ ###########

AND CHAIN

AND CHAIN############(##########)################################(SET TRANSACTION########)########### ###########################

##

##################COMMIT#########

################ROLLBACK################################ ###############
#ROLLBACK AND CHAIN##############

#

#############

```
ROLLBACK;
```

###

####ROLLBACK###SQL######### ROLLBACK TRANSACTION####PostgreSQL#######

####

BEGIN, COMMIT, ROLLBACK TO SAVEPOINT

ROLLBACK PREPARED

ROLLBACK PREPARED — ##########################

##

```
ROLLBACK PREPARED transaction_id
```

##

```
ROLLBACK PREPARED##########################
```

#####

transaction_id

##############################

##

####################

############################ ##########################

#######################pg_prepared_xacts#################

#

```
###########foobar#######################

ROLLBACK PREPARED 'foobar';
```

###

ROLLBACK PREPARED#PostgreSQL###### ################################ ####### ########################(X/Open XA##)################SQL############

####

PREPARE TRANSACTION, COMMIT PREPARED

ROLLBACK TO SAVEPOINT

ROLLBACK TO SAVEPOINT — ################

##

```
ROLLBACK [ WORK | TRANSACTION ] TO [ SAVEPOINT ] savepoint_name
```

##

############
################################

ROLLBACK TO SAVEPOINT##

#####

savepoint_name

   #################

##

##RELEASE SAVEPOINT#########

################################

################################
################ ############################FETCH###MOVE##################
##################FETCH################# ####FETCH#################### ###
################################ ##
################################
#################################### #############ROLLBACK TO SAVEPOINT################
#################

#

my_savepoint#######################

ROLLBACK TO SAVEPOINT my_savepoint;

################################

```
BEGIN;

DECLARE foo CURSOR FOR SELECT 1 UNION SELECT 2;

SAVEPOINT foo;

FETCH 1 FROM foo;
 ?column?
----------
        1
```

```
ROLLBACK TO SAVEPOINT foo;

FETCH 1 FROM foo;
 ?column?
----------
        2

COMMIT;
```

###

##SQL###SAVEPOINT########## ####PostgreSQL#Oracle############## SQL###### ###WORK##### TRANSACTION#######ROLLBACK################## ###SQL##AND [NO] CHAIN#(####)##########PostgreSQL################ ##################SQL# #########

####

BEGIN, COMMIT, RELEASE SAVEPOINT, ROLLBACK, SAVEPOINT

SAVEPOINT

SAVEPOINT — ##########################

##

```
SAVEPOINT savepoint_name
```

##

SAVEPOINT##############################

##
###############

#####

savepoint_name

> ################# ##
> ###

##

##################ROLLBACK TO######### ####################################
####RELEASE SAVEPOINT#########

#################################### 1####################################

#

#####################################

```
BEGIN;
    INSERT INTO table1 VALUES (1);
    SAVEPOINT my_savepoint;
    INSERT INTO table1 VALUES (2);
    ROLLBACK TO SAVEPOINT my_savepoint;
    INSERT INTO table1 VALUES (3);
COMMIT;
```

#############1#3########2########

######################

```
BEGIN;
    INSERT INTO table1 VALUES (3);
    SAVEPOINT my_savepoint;
    INSERT INTO table1 VALUES (4);
    RELEASE SAVEPOINT my_savepoint;
COMMIT;
```

#############3#4###########

#################

```
BEGIN;
    INSERT INTO table1 VALUES (1);
    SAVEPOINT my_savepoint;
    INSERT INTO table1 VALUES (2);
    SAVEPOINT my_savepoint;
    INSERT INTO table1 VALUES (3);


    -- 2##################
    ROLLBACK TO SAVEPOINT my_savepoint;

    SELECT * FROM table1;               -- # 1 # 2 ###


    -- 2#############
    RELEASE SAVEPOINT my_savepoint;


    -- 1##################
    ROLLBACK TO SAVEPOINT my_savepoint;

    SELECT * FROM table1;               -- # 1 #####
COMMIT;
```

################ 3 ############ 2 ###########

###

SQL### PostgreSQL########### ### #RELEASE SAVEPOINT################# ######ROLLBACK TO SAVEPOINT#RELEASE SAVEPOINT######################### # ######SAVEPOINT####SQL########

####

BEGIN, COMMIT, RELEASE SAVEPOINT, ROLLBACK, ROLLBACK TO SAVEPOINT

SECURITY LABEL

SECURITY LABEL — ##############################

##

```
SECURITY LABEL [ FOR provider ] ON
{
  TABLE object_name |
  COLUMN table_name.column_name |
  AGGREGATE aggregate_name ( aggregate_signature ) |
  DATABASE object_name |
  DOMAIN object_name |
  EVENT TRIGGER object_name |
  FOREIGN TABLE object_name |
  FUNCTION function_name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] |
  LARGE OBJECT large_object_oid |
  MATERIALIZED VIEW object_name |
  [ PROCEDURAL ] LANGUAGE object_name |
  PROCEDURE procedure_name [ ( [ [ argmode ] [ argname ] argtype
 [, ...] ] ) ] |
  PUBLICATION object_name |
  ROLE object_name |
  ROUTINE routine_name [ ( [ [ argmode ] [ argname ] argtype [, ...] ] ) ]
 |
  SCHEMA object_name |
  SEQUENCE object_name |
  SUBSCRIPTION object_name |
  TABLESPACE object_name |
  TYPE object_name |
  VIEW object_name
} IS { string_literal | NULL }


###aggregate_signature#########

* |
[ argmode ] [ argname ] argtype [ , ... ] |
[ [ argmode ] [ argname ] argtype [ , ... ] ] ORDER BY [ argmode ]
 [ argname ] argtype [ , ... ]
```

##

SECURITY LABEL############################## ################################ ########################## #########register_label_provider############## #############

> ##
>
> register_label_provider#SQL######### ############C################ ##

PostgreSQL## ####### ######################## ##########SELinux##################MAC#######

#################### ##DAC#####
#########################

#####

object_name
table_name.column_name
aggregate_name
function_name
procedure_name
routine_name

#################### ###############(#########)###############

provider

#################### ##
##

argmode

######################### IN#OUT#INOUT#VARIADIC####### #############IN#
################################SECURITY LABEL#OUT##################
####IN#INOUT#VARIADIC#########

argname

######################### ##################################SECURITY LABEL
ON FUNCTION#########################

argtype

##########################

large_object_oid

##########OID###

PROCEDURAL

############

string_literal

##############################

NULL

##################NULL#######

# #

#######################################

```
SECURITY LABEL FOR selinux ON TABLE mytable IS
 'system_u:object_r:sepgsql_table_t:s0';
```

####################

```
SECURITY LABEL FOR selinux ON TABLE mytable IS NULL;
```

###

##SQL##SECURITY LABEL###########

####

sepgsql, `src/test/modules/dummy_seclabel`

SELECT

SELECT, TABLE, WITH — ##################

##

```
[ WITH [ RECURSIVE ] with_query [, ...] ]
SELECT [ ALL | DISTINCT [ ON ( expression [, ...] ) ] ]
    [ { * | expression [ [ AS ] output_name ] } [, ...] ]
    [ FROM from_item [, ...] ]
    [ WHERE condition ]
    [ GROUP BY [ ALL | DISTINCT ] grouping_element [, ...] ]
    [ HAVING condition ]
    [ WINDOW window_name AS ( window_definition ) [, ...] ]
    [ { UNION | INTERSECT | EXCEPT } [ ALL | DISTINCT ] select ]
    [ ORDER BY expression [ ASC | DESC | USING operator ] [ NULLS { FIRST
 | LAST } ] [, ...] ]
    [ LIMIT { count | ALL } ]
    [ OFFSET start [ ROW | ROWS ] ]
    [ FETCH { FIRST | NEXT } [ count ] { ROW | ROWS } { ONLY | WITH
 TIES } ]
    [ FOR { UPDATE | NO KEY UPDATE | SHARE | KEY SHARE }
 [ OF from_reference [, ...] ] [ NOWAIT | SKIP LOCKED ] [...] ]
```

```
###from_item###########

    [ ONLY ] table_name [ * ] [ [ AS ] alias [ ( column_alias
 [, ...] ) ] ]
                [ TABLESAMPLE sampling_method ( argument [, ...] )
 [ REPEATABLE ( seed ) ] ]
    [ LATERAL ] ( select ) [ [ AS ] alias [ ( column_alias [, ...] ) ] ]
    with_query_name [ [ AS ] alias [ ( column_alias [, ...] ) ] ]
    [ LATERAL ] function_name ( [ argument [, ...] ] )
                [ WITH ORDINALITY ] [ [ AS ] alias [ ( column_alias
 [, ...] ) ] ]
    [ LATERAL ] function_name ( [ argument [, ...] ] ) [ AS ] alias
 ( column_definition [, ...] )
    [ LATERAL ] function_name ( [ argument [, ...] ] ) AS
 ( column_definition [, ...] )
    [ LATERAL ] ROWS FROM( function_name ( [ argument [, ...] ] ) [ AS
 ( column_definition [, ...] ) ] [, ...] )
                [ WITH ORDINALITY ] [ [ AS ] alias [ ( column_alias
 [, ...] ) ] ]
    from_item join_type from_item { ON join_condition | USING
 ( join_column [, ...] ) [ AS join_using_alias ] }
    from_item NATURAL join_type from_item
    from_item CROSS JOIN from_item
```

```
##grouping_element###########

    ( )
    expression
    ( expression [, ...] )
    ROLLUP ( { expression | ( expression [, ...] ) } [, ...] )
    CUBE ( { expression | ( expression [, ...] ) } [, ...] )
    GROUPING SETS ( grouping_element [, ...] )
```

```
##with_query#########

    with_query_name [ ( column_name [, ...] ) ] AS [ [ NOT ]
 MATERIALIZED ] ( select | values | insert | update | delete | merge )
        [ SEARCH { BREADTH | DEPTH } FIRST BY column_name [, ...]
 SET search_seq_col_name ]
        [ CYCLE column_name [, ...] SET cycle_mark_col_name
 [ TO cycle_mark_value DEFAULT cycle_mark_default ]
 USING cycle_path_col_name ]

TABLE [ ONLY ] table_name [ * ]
```

##

SELECT#0################ SELECT###############

1. WITH#################### #########FROM########################## NOT MATERIALIZED###########FROM##2#######WITH################ ####WITH#### #######
2. FROM############### #FROM########################### FROM############### ############## ####FROM###########
3. WHERE################################ ####WHERE###########
4. GROUP BY###########################1################################## ############### HAVING################################## ####GROUP BY ##HAVING########### ###########################GROUP BY################ #########
5. ########################SELECT######################## ####SELECT######### #####
6. SELECT DISTINCT############### SELECT DISTINCT ON################### ## SELECT ALL############################## ########DISTINCT###########
7. UNION#INTERSECT#EXCEPT###########SELECT#####1################## UNION## INTERSECT################## ##### EXCEPT###############2################ ALL########################## ##### ####DISTINCT################################### SELECT###ALL###### ########DISTINCT###################### ####UNION##INTERSECT##EXCEPT## #########
8. ORDER BY############################## ORDER BY########################## ####### ####ORDER BY###########
9. LIMIT####FETCH FIRST#####OFFSET#########SELECT############### ######## #LIMIT###########
10. FOR UPDATE#FOR NO KEY UPDATE#FOR SHARE###FOR KEY SHARE#######SELECT# ######################## ########################

SELECT#################SELECT####### FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE###FOR KEY SHARE##############################1######UPDATE########

#####

WITH#

#####[88.751192]#WITH#####################1##################### ########### ############################## #######
```
```

#SELECT#TABLE#VALUES#INSERT#UPDATE#DELETE########## WITH#######
#INSERT#UPDATE#DELETE#########RETURNING########### ##########
#####RETURNING########################### RETURNING##########
########################## ######WITH############################1#######
#######
#SELECT#TABLE#VALUES#INSERT#UPDATE#DELETE####MERGE###### WITH######
#INSERT#UPDATE#DELETE#MERGE#########RETURNING####### ##################
#########################RETURNING###### RETURNING##########
##################################

##############WITH################## ######################## ###############
##########

RECURSIVE#######SELECT######################### ##########################
###

non_recursive_term UNION [ALL | DISTINCT] *recursive_term*

###########UNION############### ########1################## ################
#################SELECT############### ##7.8#########

RECURSIVE######WITH############################ ##########################
####### ############################ RECURSIVE#####WITH################WITH#
#######WITH#########################

WITH###############RECURSIVE#WITH############## ##########################
##WITH##################

#####SEARCH## ##################
################ *search_seq_col_name*########WITH################### #####
######################### ##7.8.2.1#########

#####CYCLE######################### ##################################
cycle_mark_col_name########WITH################### ###############
#cycle_mark_value##########*cycle_mark_default*####### ##################
########## *cycle_mark_value#cycle_mark_default*##########################
############################ (##SQL######################PostgreSQL#######) #
######(boolean##)TRUE#FALSE###### ####*cycle_path_col_name*########WITH###
################ ######################## ##7.8.2.2#########

SEARCH##CYCLE######WITH############### *with_query*##(######UNION###)2#
#SELECT(#########)#####UNION(###UNION ALL)########## ###########SEARCH####
#####CYCLE###############

#######WITH####################### ######WITH############RETURNING#########
#################################### 2################################

WITH## #########
###

####WITH######NOT MATERIALIZED########################### #####WITH####
#########FROM#######SELECT############################## ###########WITH#
############################## ##################WITH#######################
##NOT MATERIALIZED#################################### ##################
#################SELECT#####WITH######NOT MATERIALIZED############

###############FROM######################WITH################### #######
#################################### ####WITH######MATERIALIZED###########
################# ##################################WITH###################
PostgreSQL#####12##WITH#######
###################

#########7.8##########

FROM#

FROM###SELECT#############1######## ######################################
########WHERE#####################################

FROM###############

table_name

############################## ########ONLY#########################
ONLY## ####################*###
##################################

alias

#####FROM######## ##
####FROM foo AS f######
#####SELECT##########FROM###foo####f############## ####################
##############################

TABLESAMPLE *sampling_method* (*argument* [, ...]) [REPEATABLE (*seed*)]

table_name###TABLESAMPLE##########################*sampling_method*#
############### ########WHERE######################### PostgreSQL########
#########BERNOULLI#SYSTEM#2#################### ##############(extension)#
#########################

##########BERNOULLI#SYSTEM#####1###*argument*######################0##
100############### #####real############## ##########################
############# ###2###
BERNOULLI## SYSTEM##########
############ ####################################### ##################
##SYSTEM#BERNOULLI#######################BERNOULLI###################
######

#####REPEATABLE#########################*seed*############# #####NULL#####
################# ####*argument*#####2#####################################
REPEATABLE##########################
########################### ###################REPEATABLE##############
######################

select

FROM#####SELECT########## SELECT#########SELECT#######################
#SELECT## VALUES##############
#####

with_query_name

WITH## #####WITH#########
############################## ##
#########################

function_name

FROM################# ## SELECT######
################################ #######(###OUT###############)############
#################

###########WITH ORDINALITY########bigint################### ######## ######1############ ###########ordinality#######

############################## ### #####ordinality#########################

##########ROWS FROM(...)########1##FROM################ ############ ####################2################ ################################## #NULL####################################

###record#######################AS############*column_name* *data_type* [, ...]################## ##

ROWS FROM(...)##########1####################ROWS FROM(...)### ########################### ###1#####WITH ORDINALITY#################ROWS FROM(...)############

ORDINALITY################ROWS FROM(...)#############ROWS FROM(...)###############

join_type

##########

- [INNER] JOIN
- LEFT [OUTER] JOIN
- RIGHT [OUTER] JOIN
- FULL [OUTER] JOIN

INNER###OUTER################ON *join_condition*#USING (*join_column* [, ...])#NATURAL#####1################ #############

JOIN###2##FROM######### ########################FROM############ ###### ######################### #######JOIN############# ########JOIN######### #FROM################ JOIN########################## ########FROM#WHERE# ################

LEFT OUTER JOIN### ################################## ## #NULL######## ############JOIN######################### ###############

###RIGHT OUTER JOIN####################################NULL##########1 ######### #############LEFT OUTER JOIN########RIGHT OUTER JOIN####### ##################

FULL OUTER JOIN##########################NULL###################NULL# ##########

ON *join_condition*

join_condition######################boolean#########WHERE##########

USING (*join_column* [, ...]) [AS *join_using_alias*]

USING (a, b, ...)#######ON left_table.a = right_table.a AND left_table.b = right_table.b ...###### ##USING###################### ############

join_using_alias############################ USING#################### ####### ###*alias*################################### ######*alias*######### ############# — #######USING#############

NATURAL

NATURAL##2#####################USING######### ##########NATURAL#ON TRUE########

CROSS JOIN

CROSS JOIN#INNER JOIN ON(TRUE)##### ##################### ########### ####### ####FROM#########2#####################(#####)##############

LATERAL

LATERAL######SELECT#FROM############## ######SELECT#FROM######### #FROM################ #LATERAL#####SELECT##############FROM########### ###########

LATERAL#######FROM############# ################## ############### #FROM#################

LATERAL###FROM#######JOIN################ #####JOIN################### ###########

FROM###LATERAL######################## #############FROM############### #####FROM########## LATERAL###################### ###################### ### ################################

#########LATERAL###INNER###LEFT############### ###### LATERAL########## ######################### #####*X* RIGHT JOIN LATERAL *Y*################ ####*Y*##*X*#############

WHERE#

WHERE###########################

WHERE *condition*

condition#########boolean########## ######################### ############# ##############################

GROUP BY#

GROUP BY###########################

GROUP BY [ALL | DISTINCT] *grouping_element* [, ...]

GROUP BY##################################1############### *grouping_element*#######*expression*#############SELECT########/######### ######################## ##########GROUP BY#########################

###########GROUPING SETS#ROLLUP#CUBE###############GROUP BY############## ############## ##################GROUP BY############UNION ALL######## #### #DISTINCT####################UNION ALL#UNION DISTINCT######## ######### #########7.2.4#########

###########GROUP BY###### ####################1#################### ########################FILTER#### ################### ####4.2.7######### FILTER##################################

GROUP BY##SELECT############# ## ###### #########(#########)#######################################

##########HAVING##SELECT################################## #######CASE####################################### 4.2.14#########

####FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#GROUP BY#############

HAVING#

HAVING###########################

HAVING *condition*

condition#WHERE#############

HAVING#################################### HAVING#WHERE##########WHERE##GROUP BY#########################HAVING##GROUP BY################## *condition*###

HAVING######GROUP BY################################## GROUP BY############# ########## ##########1#######################SELECT####HAVING##HAVING#################0########

####FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#HAVING##########

WINDOW#

WINDOW############################

WINDOW *window_name* AS (*window_definition*) [, ...]

###window_name##OVER########################## *###window_definition*#####

[*existing_window_name*]
[PARTITION BY *expression* [, ...]]
[ORDER BY *expression* [ASC | DESC | USING *operator*] [NULLS { FIRST | LAST }] [, ...]]
[*frame_clause*]

existing_window_name##########WINDOW############################ ########### ###PARTITION BY############## ORDER BY############ ################ #PARTITION BY############## ##############ORDER BY########ORDER BY###### ####### ######################### ############################

PARTITION BY######GROUP BY################### ########################### ########### ##################GROUP BY############################## ######### ################################

####ORDER BY###########ORDER BY################### ######################### ###############

frame_clause## ######### ##########(##########)########### *frame_clause*##################

```
{ RANGE | ROWS | GROUPS } frame_start [ frame_exclusion ]
{ RANGE | ROWS | GROUPS } BETWEEN frame_start AND frame_end
 [ frame_exclusion ]
```

###*frame_start*#*frame_end*##################

```
UNBOUNDED PRECEDING
offset PRECEDING
CURRENT ROW
offset FOLLOWING
UNBOUNDED FOLLOWING
```

###*frame_exclusion*###################

```
EXCLUDE CURRENT ROW
EXCLUDE GROUP
EXCLUDE TIES
EXCLUDE NO OTHERS
```

frame_end##############CURRENT ROW###### *frame_start*#UNBOUNDED FOLLOWING##########*frame_end*#UNBOUNDED PRECEDING########### ##*frame_start*#*frame_end*###########*frame_end*####*frame_start*########### ################### ###RANGE BETWEEN CURRENT ROW AND *offset* PRECEDING# #######

################RANGE UNBOUNDED PRECEDING### ###RANGE BETWEEN UNBOUNDED PRECEDING AND CURRENT ROW##### ################### #######ORDER BY#############ORDER BY############################## ### ##RANGE#ROWS#GROUPS##########UNBOUNDED PRECEDING######################## #######UNBOUNDED FOLLOWING############################## ROWS#####CURRENT ROW##############################RANGE####GROUPS#########ORDER BY####### ############################## *offset* PRECEDING###*offset* FOLLOWING#### ###################### ROWS######*offset*################################## GROUPS######*offset*## ####### ########ORDER BY################ RANGE######*offset*################### #ORDER BY########### ###########*offset*######################PRECEDING## ##############FOLLOWING################## #####*offset*######################## ### ###########*offset*################################interval###### ####### ##*offset*####NULL############## ###*offset*################################## ###########

frame_exclusion## ###### EXCLUDE CURRENT ROW################ EXCLUDE GROUP################## ###### EXCLUDE TIES########################## EXCLUDE NO OTHERS######## ################################

ORDER BY######################ROWS################################## RANGE## #GROUPS#####ORDER BY## ########

WINDOW###########SELECT######ORDER BY######################### ####### #OVER#######WINDOW################ ###WINDOW#################### ######### ################## ###########OVER##########################WINDOW########### ################ ###WINDOW##

####FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#WINDOW##### ###########

##################3.5#4.2.8#7.2.5##########

SELECT###

SELECT####SELECT#####FROM##############SELECT#################### #### ##FROM################################

###########SELECT################ ###SELECT################################ ## ###SELECT## ### ############################AS *output_name*######### #######PostgreSQL### ######C##############AS############ ######################AS################ ################ #############PostgreSQL############# ###################### ########### ####################################?column?################### ##

ORDER BY##GROUP BY######################## ####WHERE#HAVING############## ################

##############################*########### ######################## ##*table_name*.*########### ########AS#################### ################## ##

##SQL##############DISTINCT#ORDER BY#LIMIT#################### DISTINCT# ################## ################DISTINCT############## #########ORDER BY#LIMIT################## #########(volatile)#################### ############ ################################### PostgreSQL#####DISTINCT#ORDER BY#GROUP BY###################(limit)################## #########SELECT f(x) FROM tab ORDER BY 1 ######f(x)#################### ################################## #####LIMIT#######################

##
PostgreSQL######9.6## ###### ###################

DISTINCT#

SELECT DISTINCT############################ ############1######### SELECT ALL################## ###########

SELECT DISTINCT ON (*expression* [, ...])############################## ##### DISTINCT ON###ORDER BY############### ###########ORDER BY########### ##################################### ##########################

```
SELECT DISTINCT ON (location) location, time, report
    FROM weather_reports
    ORDER BY location, time DESC;
```

###ORDER BY###

DISTINCT ON######ORDER BY#################### ORDER BY#######DISTINCT ON# ############################

####FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#DISTINCT## #############

UNION#

UNION################

select_statement UNION [ALL | DISTINCT] *select_statement*

select_statement###ORDER BY#LIMIT#FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE########SELECT###### #ORDER BY#LIMIT################ ####### ###########################UNION##################

UNION#####2##SELECT############### #######2##SELECT###################### ########## UNION###########2##SELECT#################################### ##########

ALL################UNION############### ALL#################### ####### #UNION ALL#UNION######### ####ALL########## ######################### #DISTINCT############

1##SELECT#####UNION################################

######UNION####UNION#######FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#############

INTERSECT#

INTERSECT################

select_statement INTERSECT [ALL | DISTINCT] *select_statement*

select_statement###ORDER BY#LIMIT#FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#########SELECT#######

INTERSECT##2##SELECT############### ############2##SELECT#################

ALL##############INTERSECT############## ALL################*m*#########*n*##### ##########min(*m*,*n*)####### ###########################DISTINCT############

1##SELECT#####INTERSECT################################ INTERSECT#UNION### ########### ####A UNION B INTERSECT C #A UNION (B INTERSECT C)#######

######INTERSECT####INTERSECT#######FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE###FOR KEY SHARE#############

EXCEPT#

EXCEPT################

select_statement EXCEPT [ALL | DISTINCT] *select_statement*

select_statement###ORDER BY#LIMIT#FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE#########SELECT#######

EXCEPT#####SELECT############SELECT#####################

ALL################EXCEPT############### ALL############*m*#########*n*####### #######max(*m*-*n*,0)####### ###########################DISTINCT############

1##SELECT#####EXCEPT################################ EXCEPT########UNION## ####

######EXCEPT####EXCEPT#######FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE## #FOR KEY SHARE#############

ORDER BY#

ORDER BY###########################

```
ORDER BY expression [ ASC | DESC | USING operator ] [ NULLS { FIRST |
 LAST } ] [, ...]
```

ORDER BY################################### ################2############## ##1################################### ####################################

expression#######SELECT##

############################## ################################ AS########### ##

###ORDER BY####SELECT########################### ###############

```
SELECT name FROM distributors ORDER BY code;
```

####UNION#INTERSECT#EXCEPT####ORDER BY################################### ########

ORDER BY##################################ORDER BY############### ######### ###GROUP BY######### #########SQL####################

ORDER BY##############ASC#####DESC###############(####)# ##############ASC## ########### ###############USING############# ###########B-Tree############## ############# ###ASC#USING <##DESC#USING >##### ############################ ##

NULLS LAST#######NULL######NULL########### NULLS FIRST#######NULL#### ###NULL########### ################################ASC####NULLS LAST#DESC# #######NULLS FIRST### ##############NULL##NULL#################### USING#############NULL################################

######################### #####ORDER BY x, y DESC#ORDER BY x DESC, y DESC# ############

################################## ########*expression*##COLLATE########### #### ###ORDER BY mycolumn COLLATE "en_US"### #########4.2.10###23.2#########

LIMIT#

LIMIT##2################

```
LIMIT { count | ALL }
OFFSET start
```

#####*count*###############*start*#################### ###########*start*###### ########*count*#######

count##NULL########LIMIT ALL################### *start*#NULL####### ##OFFSET 0#########

SQL:2008######################### PostgreSQL########### ########

```
OFFSET start { ROW | ROWS }
FETCH { FIRST | NEXT } [ count ] { ROW | ROWS } { ONLY | WITH TIES }
```

#########*start*###*count*#####SQL############################ PostgreSQL#######
count#FETCH#################### WITH
TIES############ORDER BY################################ORDER BY###
##SKIP LOCKED######## ROW###ROWS####FIRST###NEXT#########################
SQL####OFFSET###FETCH############################ ###PostgreSQL#########
###########

LIMIT###################ORDER BY########### ##############################
10#20#######################10#20######### ORDER BY#####################

#######################LIMIT#######LIMIT#OFFSET##########################
################## ######LIMIT/OFFSET#######################ORDER BY#######
###################### ############# #SQL##ORDER BY#########################
####################

################ORDER BY#######LIMIT#######################################
##########################

######

FOR UPDATE#FOR NO KEY UPDATE#FOR SHARE###FOR KEY SHARE######### ######
################SELECT#################

#####################

```
FOR lock_strength [ OF from_reference [, ...] ] [ NOWAIT | SKIP LOCKED ]
```

###*lock_strength*##################

```
UPDATE
NO KEY UPDATE
SHARE
KEY SHARE
```

#####*from_reference*##FROM##########*alias*#####*table_name*########## ####
##############13.3.2#########

################################NOWAIT####SKIP LOCKED############## NOWAIT#
#################################### SKIP LOCKED#######################
##
####################### NOWAIT###SKIP LOCKED########################### ####
###ROW SHARE################## #13############ ##############################
########LOCK#NOWAIT##############

SELECT####################
####################################
######################## ####################WITH############## WITH########
###########WITH######################

############################
################################### #######################NOWAIT##########
#####NOWAIT######### ##########################SKIP LOCKED###############
#SKIP LOCKED#########

################

#######SELECT### ##################
################# ##
######################## LIMIT################################## #####OFFSET##
########################### ##
###########

########SELECT## ######################
####

```
SELECT * FROM (SELECT * FROM mytable FOR UPDATE) ss WHERE col1 = 5;
```

##############################col1 = 5##############

#################################### ###########

```
BEGIN;
SELECT * FROM mytable WHERE key = 1 FOR UPDATE;
SAVEPOINT s;
UPDATE mytable SET ... WHERE key = 1;
ROLLBACK TO s;
```

ROLLBACK TO##FOR UPDATE############ #######9.3########

> ##
>
> ORDER BY#############READ COMMITTED#################SELECT########
> ################### ORDER BY############# #####################1#######
> ######################## ##SELECT#######################################
> ### ## ######################
> #FOR UPDATE/SHARE####################
>
> ```
> SELECT * FROM (SELECT * FROM mytable FOR UPDATE) ss ORDER BY
> column1;
> ```
>
> #########FOR UPDATE##############################mytable############
> ########## ######ORDER BY#LIMIT#####################################
> ## ##
>
> REPEATABLE READ###SERIALIZABLE################'40001'###SQLSTATE#
> ##################### ####################################

TABLE####

####

TABLE *name*

#########

SELECT * FROM *name*

###
WITH#UNION#INTERSECT#EXCEPT#ORDER BY#LIMIT#OFFSET#FETCH#FOR######
#TABLE############# WHERE########################

# #

films#####distributors##########

```
SELECT f.title, f.did, d.name, f.date_prod, f.kind
    FROM distributors d JOIN films f USING (did);

       title       | did |     name     | date_prod  |   kind
-------------------+-----+--------------+------------+----------
 The Third Man     | 101 | British Lion | 1949-12-23 | Drama
 The African Queen | 101 | British Lion | 1951-08-11 | Romantic
 ...
```

######len#####kind################

```
SELECT kind, sum(len) AS total FROM films GROUP BY kind;

   kind   | total
----------+-------
 Action   | 07:34
 Comedy   | 02:58
 Drama    | 14:28
 Musical  | 06:42
 Romantic | 04:38
```

######len#####kind#################5####################

```
SELECT kind, sum(len) AS total
    FROM films
    GROUP BY kind
    HAVING sum(len) < interval '5 hours';

   kind   | total
----------+-------
 Comedy   | 02:58
 Romantic | 04:38
```

######2#####name################2#######

```
SELECT * FROM distributors ORDER BY name;
SELECT * FROM distributors ORDER BY 2;

 did |       name
-----+------------------
 109 | 20th Century Fox
 110 | Bavaria Atelier
 101 | British Lion
 107 | Columbia
 102 | Jean Luc Godard
 113 | Luso films
 104 | Mosfilm
 103 | Paramount
 106 | Toho
 105 | United Artists
 111 | Walt Disney
 112 | Warner Bros.
 108 | Westward
```

#####distributors#####actors#####################################W#########
########### ###############ALL##############

```
distributors:                actors:
 did |     name              id |     name
-----+--------------        ----+----------------
 108 | Westward               1 | Woody Allen
 111 | Walt Disney            2 | Warren Beatty
 112 | Warner Bros.           3 | Walter Matthau
 ...                         ...

SELECT distributors.name
    FROM distributors
    WHERE distributors.name LIKE 'W%'
UNION
SELECT actors.name
    FROM actors
    WHERE actors.name LIKE 'W%';

      name
----------------
 Walt Disney
 Walter Matthau
 Warner Bros.
 Warren Beatty
 Westward
 Woody Allen

###FROM##########################################


CREATE FUNCTION distributors(int) RETURNS SETOF distributors AS $$
    SELECT * FROM distributors WHERE did = $1;
$$ LANGUAGE SQL;

SELECT * FROM distributors(111);
 did |    name
-----+-------------
 111 | Walt Disney

CREATE FUNCTION distributors_2(int) RETURNS SETOF record AS $$
    SELECT * FROM distributors WHERE did = $1;
$$ LANGUAGE SQL;

SELECT * FROM distributors_2(111) AS (f1 int, f2 text);
 f1  |     f2
-----+-------------
 111 | Walt Disney

##################


SELECT * FROM unnest(ARRAY['a','b','c','d','e','f']) WITH ORDINALITY;
 unnest | ordinality
--------+----------
 a      |        1
 b      |        2
 c      |        3
 d      |        4
 e      |        5
 f      |        6
(6 rows)

#########WITH############


WITH t AS (
```

```
    SELECT random() as x FROM generate_series(1, 3)
  )
SELECT * FROM t
UNION ALL
SELECT * FROM t;
         x
--------------------
  0.534150459803641
  0.520092216785997
 0.0735620250925422
  0.534150459803641
  0.520092216785997
 0.0735620250925422
```

WITH########################## ####3############2###########

######WITH RECURSIVE###########################Mary#########################
####

```
WITH RECURSIVE employee_recursive(distance, employee_name, manager_name)
 AS (
    SELECT 1, employee_name, manager_name
    FROM employee
    WHERE manager_name = 'Mary'
  UNION ALL
    SELECT er.distance + 1, e.employee_name, e.manager_name
    FROM employee_recursive er, employee e
    WHERE er.employee_name = e.manager_name
  )
SELECT distance, employee_name FROM employee_recursive;
```

########UNION## ########################
########### ################### ###########7.8###########

#######manufacturers###############get_product_names()########
#LATERAL#######

```
SELECT m.name AS mname, pname
FROM manufacturers m, LATERAL get_product_names(m.id) pname;
```

##############################

```
SELECT m.name AS mname, pname
FROM manufacturers m LEFT JOIN LATERAL get_product_names(m.id) pname ON
 true;
```

###

######SELECT####SQL########## ############################

FROM####

PostgreSQL###FROM############## ##############################

```
SELECT 2+2;

 ?column?
----------
        4
```

##SQL##############SELECT###########1#######################

##SELECT###

SELECT## #####SQL#############
##PostgreSQL################################### ####DISTINCT################
#####

AS########

##SQL########AS###
PostgreSQL############### ################################AS######
############################AS#########################

FROM##########PostgreSQL###################AS############ ##################
#############

FROM###SELECT######

##SQL###FROM######SELECT######### PostgreSQL##############

ONLY#####

##SQL###SELECT * FROM ONLY (tab1), ONLY (tab2) WHERE ...#####ONLY####
########################## PostgreSQL######################

PostgreSQL#####*####### ###############ONLY################## ##############

#######ONLY###############SQL###############

TABLESAMPLE####

#######TABLESAMPLE################################ SQL#####FROM############
##############

FROM########

PostgreSQL###FROM############################## ##SQL#############SELECT####
####### ###FROM *func*(...) *alias*#####FROM LATERAL (SELECT *func*(...))
alias###### ####LATERAL################### ####FROM##UNNEST()####LATERAL#
########### PostgreSQL##UNNEST()######################

GROUP BY#ORDER BY#############

##SQL-92###ORDER BY######################GROUP BY#######################
PostgreSQL########################## ########################### ####PostgreSQL
#################### ##

SQL:1999#####SQL-92############################## ############PostgreSQL#
SQL:1999######ORDER BY#GROUP BY######

#####

#########GROUP BY#############PostgreSQL##GROUP BY####################### ##
SQL###########################

LIMIT###OFFSET

LIMIT###OFFSET##PostgreSQL########MySQL########## LIMIT############
SQL:2008#######OFFSET ... FETCH {FIRST|NEXT} ...######## #####IBM DB2##

######## #Oracle##rownum###############
########PostgreSQL##########

FOR NO KEY UPDATE#FOR UPDATE#FOR SHARE#FOR KEY SHARE

FOR UPDATE###SQL############DECLARE CURSOR################### PostgreSQL#
###SELECT#####SELECT####### ######## FOR NO KEY UPDATE#FOR SHARE#FOR KEY
SHARE#######NOWAIT#SKIP LOCKED###############

WITH########

#####[81.666667]#PostgreSQL##WITH########INSERT#UPDATE###DELETE############ #
####SQL######## ######PostgreSQL###INSERT#UPDATE#DELETE#MERGE#WITH########
########## #####SQL########

####

DISTINCT ON (...)###SQL######

ROWS FROM(...)###SQL######

WITH#MATERIALIZED#NOT MATERIALIZED######SQL########

SELECT INTO

SELECT INTO — ######################

##

```
[ WITH [ RECURSIVE ] with_query [, ...] ]
SELECT [ ALL | DISTINCT [ ON ( expression [, ...] ) ] ]
    [ { * | expression [ [ AS ] output_name ] } [, ...] ]
    INTO [ TEMPORARY | TEMP | UNLOGGED ] [ TABLE ] new_table
    [ FROM from_item [, ...] ]
    [ WHERE condition ]
    [ GROUP BY expression [, ...] ]
    [ HAVING condition ]
    [ WINDOW window_name AS ( window_definition ) [, ...] ]
    [ { UNION | INTERSECT | EXCEPT } [ ALL | DISTINCT ] select ]
    [ ORDER BY expression [ ASC | DESC | USING operator ] [ NULLS { FIRST
 | LAST } ] [, ...] ]
    [ LIMIT { count | ALL } ]
    [ OFFSET start [ ROW | ROWS ] ]
    [ FETCH { FIRST | NEXT } [ count ] { ROW | ROWS } ONLY ]
    [ FOR { UPDATE | SHARE } [ OF table_name [, ...] ] [ NOWAIT ] [...] ]
```

##

SELECT INTO## #########SELECT############ ####### ##########SELECT#####################

#####

TEMPORARY###TEMP

  ################################### ###CREATE TABLE##########

UNLOGGED

  ################################# ###CREATE TABLE##########

new_table

  ########################

###############SELECT#############

##

CREATE TABLE AS######SELECT INTO###### ECPG#PL/pgSQL##INTO######## ##SELECT INTO############# #####CREATE TABLE AS################ ####CREATE TABLE AS##SELECT INTO#####################

CREATE TABLE AS######SELECT INTO##USING *method*##############TABLESPACE *tablespace_name*########################## ####CREATE TABLE AS######## ### ################################## ########default_table_access_method##########

#

####films#####################films_recent#######

```
SELECT * INTO films_recent FROM films WHERE date_prod >= '2002-01-01';
```

###

##SQL###SELECT INTO## ### ###ECPG##34######PL/pgSQL##41################ PostgreSQL#############SELECT INTO############ ##SQL####SELECT INTO#################(#######SQL######### #CREATE TABLE AS##########)# ##################################CREATE TABLE AS##########

####

CREATE TABLE AS

SET

SET — ############

##

```
SET [ SESSION | LOCAL ] configuration_parameter { TO | = } { value |
 'value' | DEFAULT }
SET [ SESSION | LOCAL ] TIME ZONE { value | 'value' | LOCAL | DEFAULT }
```

##

SET#################### #19#########################SET########### ######### #################SET######################### ############################## ####### SET################################

SET(###SET SESSION###)##SET####### ####### ####################SET#####################################

SET LOCAL## 1###########SET### #SET LOCAL################ ####################SET LOCAL################## ############SET#########

SET####SET LOCAL###

SET LOCAL########SET#####CREATE FUNCTION#################SET LOCAL### ################ ############################## #####SET LOCAL############## #####SET## #######SET########### #SET####################################

> ##
>
> PostgreSQL#####8.0##8.2#####SET LOCAL############################PL/pgSQL######################### #########################

#####

SESSION

  ############################ #SESSION#LOCAL#######################

LOCAL

  ################################## COMMIT###ROLLBACK################### ##### ##

configuration_parameter

  ################ ############19#########

value

  ############# ## ##### DEFAULT##################################SET######################## ########

#19#################SET##

SCHEMA

SET SCHEMA '*value*'#SET search_path TO *value*###### ###########1#####
###############

NAMES

SET NAMES *value*##SET client_encoding TO *value*######

SEED

#########random################ #-1#1#####-1##1####################

####setseed#################

SELECT setseed(*value*);

TIME ZONE

SET TIME ZONE '*value*'#SET timezone TO '*value*'###### SET TIME ZONE#
####################### ##############

'PST8PDT'

####################

'Europe/Rome'

###########

-7

UTC####7############PDT####### ####UTC########

INTERVAL '-08:00' HOUR TO MINUTE

UTC####8############PST#######

LOCAL
DEFAULT

##########################timezone########

#######################POSIX############### ####SET TIME ZONE -7###
##SHOW TIME ZONE###########<-07>+07######

########SET#######################8.5.3##########

##

set_config############## 9.28.1########## ###pg_settings###############SET##
###############

# #

###############

SET search_path TO my_schema, public;

#############POSTGRES#############day before month(#####)#######

```
SET datestyle TO postgres, dmy;
```

```
#######################
```

```
SET TIME ZONE 'PST8PDT';
```

```
###############
```

```
SET TIME ZONE 'Europe/Rome';
```

###

SET TIME ZONE###SQL################# ##############################PostgreSQL ######################### SET#############PostgreSQL######

####

RESET, SHOW

SET CONSTRAINTS

SET CONSTRAINTS — ##########################

##

```
SET CONSTRAINTS { ALL | name [, ...] } { DEFERRED | IMMEDIATE }
```

##

SET CONSTRAINTS############################## IMMEDIATE####1############## ##### DEFERRED############################ #######IMMEDIATE#DEFERRED###### ########

###########DEFERRABLE INITIALLY DEFERRED#DEFERRABLE INITIALLY IMMEDIATE#NOT DEFERRABLE#3################### 3###NOT DEFERRABLE#### ##IMMEDIATE######SET CONSTRAINTS############# DEFERRABLE INITIALLY DEFERRED###DEFERRABLE INITIALLY IMMEDIATE###2########################## #########SET CONSTRAINTS#######################

SET CONSTRAINTS## ## ################## ## SET CONSTRAINTS ALL####################

SET CONSTRAINTS#######DEFERRED##IMMEDIATE############################ ####################################SET CONSTRAINTS############### ### ##################SET CONSTRAINTS###### ################## ######SET CONSTRAINTS##

##UNIQUE#PRIMARY KEY#REFERENCES######EXCLUDE################ NOT NULL# ##CHECK## DEFERRABLE############## ###############

#################################### ##############################

##

PostgreSQL## ###### ####SET CONSTRAINTS################ ##########################1###### ################################

#################################### ## ###########

###

##########SQL################ ####PostgreSQL##NOT NULL###CHECK############## ##### ##PostgreSQL####################################

SET ROLE

SET ROLE — ##########################

##

```
SET [ SESSION | LOCAL ] ROLE role_name
SET [ SESSION | LOCAL ] ROLE NONE
RESET ROLE
```

##

#####[55.079365]##########SQL##################*role_name*###### ############################ SET ROLE###SQL################################### ####### ################SQL###############*role_name*###### ################################ SET ROLE###SQL## ######## SET ROLE#SET SESSION AUTHORIZATION###### ######################################

###################SET################################*role_name*####SET###### ######### ##

SESSION###LOCAL######SET#############

SET ROLE NONE##########session_user########################## RESET ROLE##################################ALTER ROLE#ALTER DATABASE########## ###### ########RESET ROLE############################## ##################

##

################################## #############WITH INHERIT TRUE############# ########################## #####SET ROLE#################################### ########### ###############WITH INHERIT FALSE############################## ############# #######WITH SET TRUE##################SET ROLE############ ## ####WITH INHERIT FALSE, SET FALSE##########SET ROLE######################

SET ROLE####SET SESSION AUTHORIZATION######################## ###SET SESSION AUTHORIZATION##########SET ROLE#######################SET ROLE# ########################SET ROLE######################

SET ROLE#####ALTER ROLE###################### ################

SET ROLE#SECURITY DEFINER################

# #

```
SELECT SESSION_USER, CURRENT_USER;

 session_user | current_user
--------------+--------------
 peter        | peter

SET ROLE 'paul';
```

```
SELECT SESSION_USER, CURRENT_USER;

 session_user | current_user
--------------+--------------
 peter        | paul
```

###

PostgreSQL########"*rolename*"######## ######SQL############################ SQL################################## PostgreSQL############################ #### `SESSION#LOCAL#######RESET`###PostgreSQL######

####

SET SESSION AUTHORIZATION

SET SESSION AUTHORIZATION

SET SESSION AUTHORIZATION — ##################################

##

```
SET [ SESSION | LOCAL ] SESSION AUTHORIZATION user_name
SET [ SESSION | LOCAL ] SESSION AUTHORIZATION DEFAULT
RESET SESSION AUTHORIZATION
```

##

##########################SQL##################user_name####### ###########
######################## ##

############################
##SECURITY DEFINER############################ SET ROLE############### ##
#####################

#####[94.092827]###
##

SESSION######LOCAL########SET##################

DEFAULT#####RESET################################ ####################

##

SET SESSION AUTHORIZATION#SECURITY DEFINER################

# #

```
SELECT SESSION_USER, CURRENT_USER;

 session_user | current_user
--------------+--------------
 peter        | peter

SET SESSION AUTHORIZATION 'paul';

SELECT SESSION_USER, CURRENT_USER;

 session_user | current_user
--------------+--------------
 paul         | paul
```

###

##SQL###user_name######################## ###################### PostgreSQL
#######"username"#########SQL######### SQL############################
###PostgreSQL################################ SESSION######LOCAL####RESET##
##PostgreSQL######

##SQL#####################################

####

SET ROLE

SET TRANSACTION

SET TRANSACTION — ##################

##

```
SET TRANSACTION transaction_mode [, ...]
SET TRANSACTION SNAPSHOT snapshot_id
SET SESSION CHARACTERISTICS AS TRANSACTION transaction_mode [, ...]
```

```
###transaction_mode###########

    ISOLATION LEVEL { SERIALIZABLE | REPEATABLE READ | READ COMMITTED |
 READ UNCOMMITTED }
    READ WRITE | READ ONLY
    [ NOT ] DEFERRABLE
```

##

```
SET    TRANSACTION#####################  ###########################  SET
SESSION CHARACTERISTICS############################################### ####
##########SET TRANSACTION######################

######################################################################## ##
#######################################################

#######################################################################

READ COMMITTED

   1########################################### ###########

REPEATABLE READ

   ##########################################################################
   ###

SERIALIZABLE

   ##########################################################################
   ### ##############################################################(#####)#####
   ##########################serialization_failure#################
```

```
##SQL###READ          UNCOMMITTED#####1##############     PostgreSQL##READ
UNCOMMITTED#READ COMMITTED########

##########################################
#SELECT#INSERT#DELETE#UPDATE#MERGE#FETCH#COPY################# #########
###################13###########

################################################ ############### ##############
#######SQL###############                                  ####################
##INSERT#UPDATE#DELETE#MERGE#COPY          FROM###SQL############    ###
#CREATE#ALTER#DROP##SQL#####COMMENT#GRANT#REVOKE#TRUNCATE######### ####
############EXPLAIN ANALYZE#EXECUTE############ ##########################
##################
```

DEFERRABLE####################SERIALIZABLE##READ ONLY############## # ## ############## #SERIALIZABLE## ################### ################################

SET TRANSACTION SNAPSHOT## ######### ###########pg_export_snapshot##(9.28.5##)######################### ##### ##################### ##############################SET TRANSACTION SNAPSHOT############ #################'00000003-0000001B-1'############## ############ SET TRANSACTION SNAPSHOT################################ ##########(SELECT#INSERT#DELETE#UPDATE#MERGE#FETCH#COPY)########### #### ##############SERIALIZABLE###REPEATABLE READ#################### (#### ##READ COMMITTED###) ####### #####SERIALIZABLE## # ##

##

SET TRANSACTION#####START TRANSACTION#BEGIN############################ #############

BEGIN####START TRANSACTION####*transaction_modes*######SET TRANSACTION# ################ ###SET TRANSACTION SNAPSHOT################

################################ #default_transaction_isolation#default_transaction_read_only#default_transaction_deferrable##### ################ ####SET SESSION CHARACTERISTICS#######SET################ ######## ##############################ALTER DATABASE########## ####19######## ###

######################### #transaction_isolation#transaction_read_only#transaction_deferrable#################### ### #####1#########################SET TRANSACTION################ ############# ######SQL############################

#

##########################

```
BEGIN TRANSACTION ISOLATION LEVEL REPEATABLE READ;
SELECT pg_export_snapshot();
 pg_export_snapshot
---------------------
 00000003-0000001B-1
(1 row)
```

######################SET TRANSACTION SNAPSHOT###################

```
BEGIN TRANSACTION ISOLATION LEVEL REPEATABLE READ;
SET TRANSACTION SNAPSHOT '00000003-0000001B-1';
```

###

#########SQL########## DEFERRABLE###########SET TRANSACTION SNAPSHOT#### #####PostgreSQL######

##SQL#####################SERIALIZABLE### PostgreSQL######READ COMMITTED## #######################

##SQL#####1## #########SQL#######
##PostgreSQL###############

##SQL#######*transaction_modes*############ ########PostgreSQL################
##

SHOW

SHOW — ###############

##

```
SHOW name
SHOW ALL
```

##

SHOW####################### ########SET##postgresql.conf############ #PGOPTIONS#libpq########libpq#######################postgres################ ############ ####19###########

#####

name

############# ############19##SET################## ######################### ##########

SERVER_VERSION

################

SERVER_ENCODING

###################### ## ########

LC_COLLATE

#################################### ################################### #####################

LC_CTYPE

############################ ## ###############

IS_SUPERUSER

############################

ALL

########################

##

##current_setting############ 9.28.1########## ###pg_settings############### ####

# #

#####DateStyle############

```
SHOW DateStyle;
 DateStyle
-----------
 ISO, MDY
(1 row)
```

#####geqo#############

```
SHOW geqo;
 geqo
------
 on
(1 row)
```

##########

```
SHOW ALL;
            name         | setting |                 description
-------------------------+---------
+-------------------------------------------------
 allow_system_table_mods | off     | Allows modifications of the structure
 of ...
    .
    .
    .
 xmloption               | content | Sets whether XML data in implicit
 parsing ...
 zero_damaged_pages      | off     | Continues processing past damaged
 page headers.
(196 rows)
```

###

SHOW#####PostgreSQL######

####

SET, RESET

START TRANSACTION

START TRANSACTION — ################

##

```
START TRANSACTION [ transaction_mode [, ...] ]
```

transaction_mode###############

```
    ISOLATION LEVEL { SERIALIZABLE | REPEATABLE READ | READ COMMITTED |
 READ UNCOMMITTED }
    READ WRITE | READ ONLY
    [ NOT ] DEFERRABLE
```

##

############################ ##########/####################SET TRANSACTION## ################################ ###########BEGIN##########

#####

################SET TRANSACTION##########

###

##SQL##################START TRANSACTION################# ###SQL########## ############ PostgreSQL###START TRANSACTION#####BEGIN#################### ###########COMMIT################ ################# ######################### ############################

DEFERRABLE *transaction_mode*#PostgreSQL#######

##SQL######*transaction_modes*############ ####PostgreSQL################### #######

SET TRANSACTION###############

####

BEGIN, COMMIT, ROLLBACK, SAVEPOINT, SET TRANSACTION

TRUNCATE

TRUNCATE — 1################

##

```
TRUNCATE [ TABLE ] [ ONLY ] name [ * ] [, ... ]
    [ RESTART IDENTITY | CONTINUE IDENTITY ] [ CASCADE | RESTRICT ]
```

##

TRUNCATE###################### ################DELETE#######################
###################### ########VACUUM################################ ########
######################

#####

name

######################### ########ONLY######################### ONLY#######
#################################### ###############*######################
################

RESTART IDENTITY

####################################

CONTINUE IDENTITY

#########################

CASCADE

##############CASCADE##

RESTRICT

###########

##

####################TRUNCATE###############

TRUNCATE###############ACCESS EXCLUSIVE########## ##########################
######## RESTART IDENTITY#################################### ###########
############ DELETE#################

###TRUNCATE###
########### ##
CASCADE###################################
###################### ##
##################################

TRUNCATE#######ON DELETE################## ####ON TRUNCATE########## ######
####ON TRUNCATE##############################BEFORE TRUNCATE########### ###

##########################AFTER TRUNCATE########## ######################### ##################################

TRUNCATE#MVCC########### ## #################### #####13.6##########

################TRUNCATE############### ##################################### ##

RESTART IDENTITY############ALTER SEQUENCE RESTART################ ## ##################################### ##################################### ##################### ##############################currval()############## ## #############currval()## ################ ##################currval()###########

TRUNCATE###postgres_fdw##########

# #

bigtable#######fattable##########

```
TRUNCATE bigtable, fattable;
```

##

```
TRUNCATE bigtable, fattable RESTART IDENTITY;
```

othertable##################othertable###################

```
TRUNCATE othertable CASCADE;
```

###

##SQL:2008###TRUNCATE TABLE *tablename*######TRUNCATE########## CONTINUE IDENTITY/RESTART IDENTITY############################ ################### ##################### ####################################

####

DELETE

UNLISTEN

UNLISTEN — #########

##

```
UNLISTEN { channel | * }
```

##

UNLISTEN#######NOTIFY################### UNLISTEN#####PostgreSQL#######
##*name*######################### #######*#################################

NOTIFY###LISTEN#NOTIFY#################

#####

channel

###################

*

##########################

##

################################ ###############

#############UNLISTEN *#######

UNLISTEN##################################

# #

########

```
LISTEN virtual;
NOTIFY virtual;
Asynchronous notification "virtual" received from server process with PID
 8448.
```

UNLISTEN###########NOTIFY############

```
UNLISTEN virtual;
NOTIFY virtual;

-- NOTIFY#############
```

###

##SQL##UNLISTEN##########

####

LISTEN, NOTIFY

UPDATE

UPDATE — ##########

##

```
[ WITH [ RECURSIVE ] with_query [, ...] ]
UPDATE [ ONLY ] table_name [ * ] [ [ AS ] alias ]
    SET { column_name = { expression | DEFAULT } |
          ( column_name [, ...] ) = [ ROW ] ( { expression | DEFAULT }
 [, ...] ) |
          ( column_name [, ...] ) = ( sub-SELECT )
        } [, ...]
    [ FROM from_item [, ...] ]
    [ WHERE condition | WHERE CURRENT OF cursor_name ]
    [ RETURNING { * | output_expression [ [ AS ] output_name ] } [, ...] ]
```

##

UPDATE########################## SET######################## SET############## ############

################################### 2########## 1###SELECT##########1##FROM##### ############## ####################

RETURNING########UPDATE################################ ##########FROM### ############################## ########################## RETURNING###### #SELECT###########

################################UPDATE############### # #expressions#condition###########SELECT##########

#####

with_query

WITH####UPDATE#################################### 7.8#SELECT##########

table_name

########################## #######ONLY################################ ONLY## ###############*#### ############################

alias

############ ############################## #####UPDATE foo AS f# ##UPDATE#########foo####f######################

column_name

table_name################ ################################### ########## ############## #####UPDATE table_name SET table_name.col = 1######

expression

########## ##

DEFAULT

##################################NULL###### IDENTITY################### ########### ##

sub-SELECT

##################################SELECT######## ##############1########## #### 1########################### 1##################NULL######## ###### ##############################

from_item

WHERE## ###SELECT##FROM########### ################### ############*from_item*################################ ##*from_item*##############################

condition

boolean######### ####true#############

cursor_name

WHERE CURRENT OF################# ############################### #### #UPDATE################################## WHERE CURRENT OF############# ######### WHERE CURRENT OF###################DECLARE#########

output_expression

#############UPDATE########### ######*table_name*###FROM################ ############### ###########*#########

output_name

###############

##

###########UPDATE######################

UPDATE *count*

count############################ #####BEFORE UPDATE############## ##*condition*############################## *count*#0########################### ### ################

UPDATE#####RETURNING############RETURNING################################ ######SELECT################

##

FROM####################*from_item*### #### FROM#################1############################## ################### ###################### ##############################1######################## ##########

##########################SELECT##

#############DELETE#INSERT#####

################UPDATE#DELETE####################### #####1############
#UPDATE##############################2##################2########UPDATE##
#DELETE########### #########2#UPDATE###DELETE##################(##
SQLSTATE##'40001'###)####### ###################################### #######
##################################2############################
#UPDATE/DELETE######

(############################)##
############################

################################UPDATE#####################################
####################

#

films#####kind####Drama######Dramatic######

```
UPDATE films SET kind = 'Dramatic' WHERE kind = 'Drama';
```

weather##

```
UPDATE weather SET temp_lo = temp_lo+1, temp_hi = temp_lo+15, prcp =
 DEFAULT
  WHERE city = 'San Francisco' AND date = '2003-07-03';
```

####################

```
UPDATE weather SET temp_lo = temp_lo+1, temp_hi = temp_lo+15, prcp =
 DEFAULT
  WHERE city = 'San Francisco' AND date = '2003-07-03'
  RETURNING temp_lo, temp_hi, prcp;
```

##############################

```
UPDATE weather SET (temp_lo, temp_hi, prcp) = (temp_lo+1, temp_lo+15,
 DEFAULT)
  WHERE city = 'San Francisco' AND date = '2003-07-03';
```

FROM##########Acme Corporation#######################1#######

```
UPDATE employees SET sales_count = sales_count + 1 FROM accounts
  WHERE accounts.name = 'Acme Corporation'
  AND employees.id = accounts.sales_person;
```

WHERE###SELECT###############

```
UPDATE employees SET sales_count = sales_count + 1 WHERE id =
  (SELECT sales_person FROM accounts WHERE name = 'Acme Corporation');
```

accounts##

```
UPDATE accounts SET (contact_first_name, contact_last_name) =
    (SELECT first_name, last_name FROM employees
     WHERE employees.id = accounts.sales_person);
```

#########################

```
UPDATE accounts SET contact_first_name = first_name,
```

```
                    contact_last_name = last_name
  FROM employees WHERE employees.id = accounts.sales_person;
```

####employees.id##########2########################## ################id# ######################## #####accounts.sales_person#################### ################NULL#######2####################

summary##########################

```
UPDATE summary s SET (sum_x, sum_y, avg_x, avg_y) =
    (SELECT sum(x), sum(y), avg(x), avg(y) FROM data d
     WHERE d.group_id = s.group_id);
```

################# ################################## ######################## ########################

```
BEGIN;

-- ###############
SAVEPOINT sp1;
INSERT INTO wines VALUES('Chateau Lafite 2003', '24');

-- ##########################
-- ##################
ROLLBACK TO sp1;
UPDATE wines SET stock = stock + 24 WHERE winename = 'Chateau Lafite
 2003';

-- ####################
COMMIT;
```

films########c_films##############kind#######

```
UPDATE films SET kind = 'Dramatic' WHERE CURRENT OF c_films;
```

####################VACUUM#################################### UPDATE# #LIMIT### ###PostgreSQL############## #####ctid#################

```
WITH exceeded_max_retries AS (
  SELECT w.ctid FROM work_item AS w
    WHERE w.status = 'active' AND w.num_retries > 10
    ORDER BY w.retry_timestamp
    FOR UPDATE
    LIMIT 5000
)
UPDATE work_item SET status = 'failed'
  FROM exceeded_max_retries AS emr
  WHERE work_item.ctid = emr.ctid;
```

#################################### ORDER BY############################## # ## ##############CTE#SKIP LOCKED#### ############################## ##########################SKIP LOCKED###LIMIT# #######UPDATE######

###

#########SQL######## ###FROM####RETURNING##PostgreSQL###### UPDATE#WITH#### ##############

##############FROM##############FROM############################ ### PostgreSQL####FROM############ ################################

PostgreSQL############## ######sub-SELECT######### ######################DEFAULT###############sub-SELECT##########

VACUUM

VACUUM — ################################

##

```
VACUUM [ ( option [, ...] ) ] [ table_and_columns [, ...] ]
```

###option##########

```
    FULL [ boolean ]
    FREEZE [ boolean ]
    VERBOSE [ boolean ]
    ANALYZE [ boolean ]
    DISABLE_PAGE_SKIPPING [ boolean ]
    SKIP_LOCKED [ boolean ]
    INDEX_CLEANUP { AUTO | ON | OFF }
    PROCESS_MAIN [ boolean ]
    PROCESS_TOAST [ boolean ]
    TRUNCATE [ boolean ]
    PARALLEL integer
    SKIP_DATABASE_STATS [ boolean ]
    ONLY_DATABASE_STATS [ boolean ]
    BUFFER_USAGE_LIMIT size
```

table_and_columns#########

```
    table_name [ ( column_name [, ...] ) ]
```

##

VACUUM #################### PostgreSQL## ########## ########VACUUM############## ####################VACUUM######### #######

table_and_columns###########VACUUM## ################## ##########VACUUM##################

VACUUM ANALYZE##########1########VACUUM######ANALYZE###### ################ ################ ##########ANALYZE#########

#FULL############VACUUM############################ ######################## ################################## ######################(#######)####### ####### ################### ###################CPU############# ##################### ## ############PARALLEL######################## VACUUM FULL################ ### ######################## #############ACCESS EXCLUSIVE###########

#####

FULL

> ################################## ##################### ################### ## ## ######### ##################################

FREEZE

################## FREEZE####vacuum_freeze_min_age###vacuum_freeze_table_age#
##########VACUUM############# ##############################FULL##########
###############

VERBOSE

############################

ANALYZE

##

DISABLE_PAGE_SKIPPING

###VACUUM####################### ### ###
##
######
######################### ##
######################################

SKIP_LOCKED

VACUUM###
############# ##############################VACUUM######################
#######VACUUM ANALYZE###
####### ###VACUUM###
################VACUUM####################

INDEX_CLEANUP

###VACUUM## ###################
###############1#######
#VACUUM####################### ######AUTO#######VACUUM###################
INDEX_CLEANUP#ON#########VACUUM###########################
###PostgreSQL################################

INDEX_CLEANUP#OFF######VACUUM##
#################### ############ID########VACUUM#######################
##########(24.1.5#########)# ############vacuum_failsafe_age##################
###ID################################### ################################
###
###############

##################################FULL###################### ########ID##
############### ########INDEX_CLEANUP#ON##############################
##

PROCESS_MAIN

VACUUM####################### ######################### ###################
########TOAST############################

PROCESS_TOAST

############TOAST############VACUUM################ ####################
FULL############
############

TRUNCATE

VACUUM## #####
##########################vacuum_truncate######################### ####
##############################ACCESS EXCLUSIVE################### #######
#FULL###################

PARALLEL

integer###################VACUUM##
###################### 27.46########## ##################################
#####################PARALLEL######################
##max_parallel_maintenance_workers################## ################
#min_parallel_index_scan_size############################## *integer*#########
################################## ######################################
#################### 1##################1############################2####
################## ## ##############
#####################FULL####################

SKIP_DATABASE_STATS

VACUUM##############XID################################## ###VACUUM#####
##################### ##
##XID#################################### #######VACUUM#################
#########################1###### #################VACUUM################
###VACUUM
(ONLY_DATABASE_STATS)############

ONLY_DATABASE_STATS

VACUUM##############XID## ##########
####*table_and_columns*#############VERBOSE##################

BUFFER_USAGE_LIMIT

VACUUM############################## ####################################
########## 0######################## ANALYZE########
##BUFFER_USAGE_LIMIT######################## ANALYZE#################
####FULL################ ##################VACUUM#vacuum_buffer_usage_limit#
######## ########VACUUM##
############# #### 128 kB##### 16 GB###

boolean

############################## ############TRUE#ON###1##############
#FALSE#OFF###0########## *boolean*################TRUE#######

integer

###########################

size

################### ################B(###)#kB(#####)#MB(#####)#GB(#####)##
#TB(#####)#####1##########################

table_name

################################
#################

column_name

############ ########## ############ANALYZE#############

##

VERBOSE########VACUUM############################## ######################
####

##

################################MAINTAIN####### ############################
######################## VACUUM#####################################

######VACUUM#####search_path#####pg_catalog, pg_temp#######

#############VACUUM#############

GIN###############VACUUM####################GIN###########################
###################### 64.4.4.1#########

PostgreSQL################
##autovacuum######## ####################################24.1#########

FULL## ############################
VACUUM FULL###########VACUUM#
############

PARALLEL####################### ########ANALYZE############ANALYZE######
##

VACUUM###I/O## ####################
################ ## ###19.4.4#########

VACUUM#FULL##########################pg_stat_progress_vacuum#########
VACUUM FULL#######################pg_stat_progress_cluster######### ##
#27.4.5#27.4.2#########

# #

onek #######1##

```
VACUUM (VERBOSE, ANALYZE) onek;
```

###

##SQL##VACUUM#######

######PostgreSQL#####9.0################################

```
VACUUM [ FULL ] [ FREEZE ] [ VERBOSE ] [ ANALYZE ] [ table_and_columns
 [, ...] ]
```

##

####

vacuumdb, 19.4.4, 24.1.6, 27.4.5, 27.4.2

VALUES

VALUES — ########

##

```
VALUES ( expression [, ...] ) [, ...]
    [ ORDER BY sort_expression [ ASC | DESC | USING operator ] [, ...] ]
    [ LIMIT { count | ALL } ]
    [ OFFSET start [ ROW | ROWS ] ]
    [ FETCH { FIRST | NEXT } [ count ] { ROW | ROWS } ONLY ]
```

##

VALUES ########################## ###################################### ###### ###########

################################## ###################### ## ### UNION ####### (10.5 #######)#

############# SELECT ##############VALUES############ ####SELECT######### ##ORDER BY#LIMIT#######FETCH FIRST####OFFSET##VALUES################

#####

expression

#################### # (####) ################ VALUES #### INSERT ######### ###### *expression* # DEFAULT ###################################### ##### VALUES ######### DEFAULT ########

sort_expression

############################## #####VALUES ###### column1#column2######### ######## ###SELECT###ORDER BY########

operator

########### ###SELECT###ORDER BY########

count

########## ###SELECT###LIMIT########

start

################ ###SELECT###LIMIT########

##

VALUES ################## ######################### VALUES # INSERT ######### ##### (#### INSERT ############# VALUES ######################) ################ ################

#

VALUES

```
VALUES (1, 'one'), (2, 'two'), (3, 'three');
```

####################################

```
SELECT 1 AS column1, 'one' AS column2
UNION ALL
SELECT 2, 'two'
UNION ALL
SELECT 3, 'three';
```

####VALUES #### SQL ############# #########INSERT #######

```
INSERT INTO films (code, title, did, date_prod, kind)
    VALUES ('T_601', 'Yojimbo', 106, '1961-06-16', 'Drama');
```

INSERT ###########VALUES ##### DEFAULT ############# ########################
##############

```
INSERT INTO films VALUES
    ('UA502', 'Bananas', 105, DEFAULT, 'Comedy', '82 minutes'),
    ('T_601', 'Yojimbo', 106, DEFAULT, 'Drama', DEFAULT);
```

VALUES ###SELECT################## ### FROM ###########

```
SELECT f.*
  FROM films f, (VALUES('MGM', 'Horror'), ('UA', 'Sci-Fi')) AS t (studio,
 kind)
  WHERE f.studio = t.studio AND f.kind = t.kind;

UPDATE employees SET salary = salary * v.increase
  FROM (VALUES(1, 200000, 1.2), (2, 400000, 1.4)) AS v (depno, target,
 increase)
  WHERE employees.depno = v.depno AND employees.sales >= v.target;
```

VALUES#FROM#############AS################# ### SELECT ########
AS## #VALUES###########PostgreSQL####
#column1#column2######## ##############################

VALUES # INSERT ########## ###################### ###########################
############ ##

```
SELECT * FROM machines
WHERE ip_address IN (VALUES('192.168.0.1'::inet), ('192.168.0.10'),
 ('192.168.1.43'));
```

> ###
>
> ## IN ############# VALUES ########## IN ####################### ########
> ##############################

###

VALUES#SQL######## LIMIT###OFFSET#PostgreSQL###### SELECT##########

####

INSERT, SELECT

PostgreSQL###############

#####PostgreSQL## ######################### ############## ############### ## ##############

##

##

clusterdb 1809
createdb 1812
createuser 1815
dropdb 1819
dropuser 1822
ecpg 1824
pg_amcheck 1826
pg_basebackup 1831
pgbench 1839
pg_combinebackup 1861
pg_config 1864
pg_dump 1867
pg_dumpall 1880
pg_isready 1886
pg_receivewal 1888
pg_recvlogical 1892
pg_restore 1896
pg_verifybackup 1905
psql 1908
reindexdb 1945
vacuumdb 1948

clusterdb

clusterdb — PostgreSQL##############

##

`clusterdb` [*connection-option*...] [*option*...] [`--table` | `-t` *table*] ... [*dbname* | `-a` | `--all`]

##

clusterdb##PostgreSQL############################## ############################## ################# #########################

clusterdb##SQL####CLUSTER######## ## ###############

#####

clusterdb######################

`-a`
`--all`

   ##################

`[-d]` *dbname*
`[--dbname=]`*dbname*

   `-a`/`--all`################################## ######################### `#PGDATABASE`######### ##################################### *dbname*##### ######## ######################################

`-e`
`--echo`

   clusterdb########################

`-q`
`--quiet`

   ##############

`-t` *table*
`--table=`*table*

   table########### ###`-t`################################

`-v`
`--verbose`

   ################

`-V`
`--version`

   clusterdb################

-?
--help

clusterdb##########################

clusterdb##################################

-h *host*
--host=*host*

######################## ################Unix#########################

-p *port*
--port=*port*

###########TCP#######Unix######################

-U *username*
--username=*username*

#############

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

#############clusterdb################

#################clusterdb################################## ####clusterdb##
################################## ################-W################

--maintenance-db=*dbname*

-a/--all### ###
#####postgres###################template1####### ############### #####
################################## ######################################
#########

##

PGDATABASE
PGHOST
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL##########libpq##################32.15##########

##

##############################CLUSTER#psql######### ########################
###libpq#####################################

# #

```
######test#########

$ clusterdb test

xyzzy###############1#foo#########

$ clusterdb --table=foo xyzzy
```

####

CLUSTER

createdb

createdb — ###PostgreSQL##########

##

`createdb` [*`connection-option`*...] [*`option`*...] [*`dbname`* [*`description`*]]

##

createdb#####PostgreSQL############

`##########################################  ##############################-O##########################`

createdb#`CREATE DATABASE`###SQL########### ##

#####

createdb######################

`dbname`

################### ##############PostgreSQL##################### ##

`description`

##############################

`-D` *`tablespace`*
`--tablespace=`*`tablespace`*

######################### ##############################

`-e`
`--echo`

createdb##########################

`-E` *`encoding`*
`--encoding=`*`encoding`*

########################### PostgreSQL####################23.3.1######

`-l` *`locale`*
`--locale=`*`locale`*

####################### ####`--lc-collate`#`--lc-ctype`####`--icu-locale`#
############### ########ICU###########`--icu-locale`############

`--lc-collate=`*`locale`*

##############LC_COLLATE########

`--lc-ctype=`*`locale`*

##############LC_CTYPE########

--builtin-locale=*locale*

##################################### ##############23.1##########

--icu-locale=*locale*

ICU#################################ICU####ID#######

--icu-rules=*rules*

###ICU##############

--locale-provider={builtin|libc|icu}

#################################

-O *owner*
--owner=*owner*

################################ ############################

-S *strategy*
--strategy=*strategy*

################ ###CREATE DATABASE STRATEGY##########

-T *template*
--template=*template*

################################### ############################

-V
--version

createdb################

-?
--help

createdb#########################

#####-D#-l#-E#-O####-T#######CREATE DATABASE###SQL###################### # ##############

##createdb##############################

-h *host*
--host=*host*

######################## ################Unix#########################

-p *port*
--port=*port*

###########TCP#######Unix######################

-U *username*
--username=*username*

#################

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

#############createdb################

#################createdb################################### ####createdb####
################################## ################-W################

--maintenance-db=*dbname*

#################################### #######postgres############# ######
################################template1####### ############### #######
################################

##

PGDATABASE

###

PGHOST
PGPORT
PGUSER

############### #########PGDATABASE################PGUSER##############
######

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL##########libpq##################32.15##########

##

################################CREATE DATABASE#psql######### #############
################# ###libpq###
##

# #

###################demo############

$ **createdb demo**

eden#########5000########template0################demo##################
##################SQL#########

$ **createdb -p 5000 -h eden -T template0 -e demo**
CREATE DATABASE demo TEMPLATE template0;

####

dropdb, CREATE DATABASE

createuser

createuser — ###PostgreSQL##############

##

`createuser` [*connection-option*...] [*option*...] [*username*]

##

createuser####PostgreSQL####################### ######################
#CREATEROLE############ ######createuser############CREATEROLE###############
######################

SUPERUSER#REPLICATION####BYPASSRLS#####################################
###CREATEROLE########### ##
################################### CREATEROLE###############

createuser#SQL####CREATE ROLE######## ##
###################

#####

createuser######################

username

####PostgreSQL############ ########PostgreSQL################################
######

-a *role*
--with-admin=*role*

############################
######## -a##############################

-c *number*
--connection-limit=*number*

################## ############

-d
--createdb

#########################

-D
--no-createdb

######################### ##########

-e
--echo

createuser######################

-E
--encrypted

##################################

-g *role*
--member-of=*role*
--role=*role* (deprecated)

-g#############################
##

-i
--inherit

################################ ##########

-I
--no-inherit

################################

--interactive

################################## -d/-D#-r/-R#-s/-S#####################
############### ####PostgreSQL 9.1###############

-l
--login

################### ### ##########

-L
--no-login

################### ##################################

-m *role*
--with-member=*role*

################################## -m##############################

-P
--pwprompt

###############createuser######################## ##########################
####

-r
--createrole

##CREATEROLE#
########### ######################################

-R
--no-createrole

######################### ##########

-s
--superuser

###################

`-S`
`--no-superuser`

##################### ##########

`-v` *`timestamp`*
`--valid-until=`*`timestamp`*

######################### #######################

`-V`
`--version`

createuser################

`--bypassrls`

####################(RLS)############

`--no-bypassrls`

#################(RLS)############# ##########

`--replication`

#######REPLICATION######## #########CREATE ROLE#############

`--no-replication`

#######REPLICATION######### #########CREATE ROLE############# ##########

`-?`
`--help`

createuser#########################

createuser##############################

`-h` *`host`*
`--host=`*`host`*

######################## ################Unix#########################

`-p` *`port`*
`--port=`*`port`*

##########TCP#######Unix######################

`-U` *`username`*
`--username=`*`username`*

###############################

`-w`
`--no-password`

############### ###################.pgpass##############################
##

`-W`
`--password`

createuser################## ####################################

##################createuser#################################### ####createuser
#################################### #################-W#################

##

PGHOST
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL##########libpq##################32.15##########

##

################################CREATE ROLE#psql######### ################
############# ###libpq###

#

###############joe############

```
$ createuser joe
```

###############joe##########################

```
$ createuser --interactive joe
Shall the new role be a superuser? (y/n) n
Shall the new role be allowed to create databases? (y/n) n
Shall the new role be allowed to create more new roles? (y/n) n
```

###eden######5000#############joe#######################################

```
$ createuser -h eden -p 5000 -S -D -R -e joe
CREATE ROLE joe NOSUPERUSER NOCREATEDB NOCREATEROLE INHERIT LOGIN;
```

joe#####################################

```
$ createuser -P -s -e joe
Enter password for new role: xyzzy
Enter it again: xyzzy
CREATE ROLE joe PASSWORD 'md5b5f5ba1a423792b526f799ae4eb3d59e' SUPERUSER
 CREATEDB CREATEROLE INHERIT LOGIN;
```

##########################
#########

####

dropuser, CREATE ROLE, createrole_self_grant

dropdb

dropdb — PostgreSQL##########

##

`dropdb` [*`connection-option`*...] [*`option`*...] *`dbname`*

##

dropdb####PostgreSQL############ ##

dropdb##SQL####`DROP DATABASE`####### ######################################
#########

#####

dropdb####################

`dbname`

#################

`-e`
`--echo`

dropdb#########################

`-f`
`--force`

################################## ################DROP DATABASE#######
###

`-i`
`--interactive`

#########################

`-V`
`--version`

dropdb################

`--if-exists`

############################ ##############

`-?`
`--help`

dropdb#########################

##dropdb##############################

`-h` *`host`*
`--host=`*`host`*

######################## ################Unix#########################

-p *port*
--port=*port*

##########TCP#######Unix#####################

-U *username*
--username=*username*

##############

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

##############dropdb#################

#################dropdb################################### ####dropdb######
################################ ################-W################

--maintenance-db=*dbname*

################################## #########postgres############# #####
################################template1####### ############### ########
################################

##

PGHOST
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL##########libpq##################32.15##########

##

##############################DROP DATABASE#psql######### #################
############ ###libpq#######################################

#

#########################demo############

$ **dropdb demo**

##########eden######5000#################demo###### ########################
#########

```
$ dropdb -p 5000 -h eden -i -e demo
Database "demo" will be permanently deleted.
Are you sure? (y/n) y
DROP DATABASE demo;
```

####

createdb, DROP DATABASE

dropuser

dropuser — PostgreSQL##############

##

`dropuser` [*connection-option*...] [*option*...] [*username*]

##

dropuser#####PostgreSQL########### ##
############CREATEROLE###############ADMIN OPTION######################

dropuser#SQL####DROP ROLE######## ##
#################

#####

dropuser######################

username

####PostgreSQL########### ###############-i/--interactive##############
###################

-e
--echo

dropuser#######################

-i
--interactive

########################### ####################################

-V
--version

dropuser################

--if-exists

###################### #############

-?
--help

dropuser#########################

dropuser##############################

-h *host*
--host=*host*

######################## ################Unix#########################

-p *port*
--port=*port*

###########TCP#######Unix######################

-U *username*
--username=*username*

############ #################

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

##############dropuser#################

##################dropuser#################################### ####dropuser##
################################### ################-W################

##

PGHOST
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL###########libpq##################32.15###########

##

##############################DROP ROLE#psql######### ######################
####### ###libpq##

# #

####################joe######

$ **dropuser joe**

###eden####5000###############joe###### ################################

$ **dropuser -p 5000 -h eden -i -e joe**
Role "joe" will be permanently removed.
Are you sure? (y/n) **y**
DROP ROLE joe;

####

createuser, DROP ROLE

ecpg

ecpg — ####SQL#C######

##

ecpg [*option*...] *file*...

##

ecpg##C##########SQL######### SQL##############################SQL####C#######C######### ################C#######################

ecpg##########################C############# ################.pgc####### ################.c######### ###########-o#######################

#######-#######ecpg################(-o#####################)###

################SQL############# #34##########

#####

ecpg#####################

-c

SQL#######C############# ##########EXEC SQL TYPE##########

-C *mode*

########### *mode*#INFORMIX#INFORMIX_SE#ORACLE##############

-D *symbol*[=*value*]

####################EXEC SQL DEFINE######## *value*###############1#######

-h

############# ########################.c####.h####################.pgc#######.pgh###### ###-c#################

-i

#######################

-I *directory*

################## ####EXEC SQL INCLUDE############################## ######### ############/usr/local/include############PostgreSQL############## ##############/usr/local/pgsql/include##/usr/include###

-o *filename*

ecpg#######*filename*############# ##############-o -########

-r *option*

############ ########*option*############

no_indicator

#########NULL############## #########################

prepare

##############(####)#### libecpg################################### ####### #######libecpg##################

questionmarks

######################### ##############

-t

#################### ########SQL## ### ###########EXEC SQL COMMIT#####################

-v

############################

--version

ecpg################

-?
--help

ecpg#########################

##

######C#####################PostgreSQL################ECPG################# ######### ##################-I##########################-I/usr/local/pgsql/include##

SQL######C###############-L/usr/local/pgsql/lib -lecpg####### ##libecpg##################

##############2##################pg_config#######

# #

####SQL#####prog1.pgc###C##

```
ecpg prog1.pgc
cc -I/usr/local/pgsql/include -c prog1.c
cc -o prog1 prog1.o -L/usr/local/pgsql/lib -lecpg
```

pg_amcheck

pg_amcheck — #####PostgreSQL####################

##

pg_amcheck [*option*...] [*dbname*]

##

pg_amcheck##
##########################amcheck#############

#############TOAST###########################B########################### #
#########################

dbname###
###
#################################
#######################
--all#--database#--exclude-database#
#############
--relation#--exclude-relation#--table#--exclude-table#--index#--
exclude-index# ##############3###############mydb*.myschema*.myrel*####
############# ####2###############mydb*.myschema*############--schema#--
exclude-schema######

dbname##############

#####

##############################

-a
--all

--exclude-database#########################

-d *pattern*
--database=*pattern*

--exclude-database#####################################
#######2#######
###

-D *pattern*
--exclude-database=*pattern*

##########################
#######2#########

-i *pattern*
--index=*pattern*

####################################
#######2#########

##--relation#############

-I *pattern*
--exclude-index=*pattern*

##########################
#######2#########

##--exclude-relation###########
##

-r *pattern*
--relation=*pattern*

################################## ########2#########

#########myrel*##############myschema*.myrel*##############
##mydb*.myschema*.myrel*############################## #################
############################

-R *pattern*
--exclude-relation=*pattern*

########################### ########2#########

--relation##

-s *pattern*
--schema=*pattern*

########2#########

################################--table=SCHEMAPAT.* --no-dependent-
indexes################ ################--index=SCHEMAPAT.*###########
#####

######################### #####mydb*#############myschema*##############
###--schema=mydb*.myschema*#############

-S *pattern*
--exclude-schema=*pattern*

#################################### ########2#########

--schema########################

-t *pattern*
--table=*pattern*

########2#########

###--relation###########
###

-T *pattern*
--exclude-table=*pattern*

######################### ########2#########

###--exclude-relation#
#############

--no-dependent-indexes

########--index#--relation#######################################B#####
############ ####################

--no-dependent-toast

########--table#--relation#######################################TOAST
############ ####################

--no-strict-names

#######--database#--table#--index#--relation#######################
############## ##########################

###########################

--exclude-toast-pointers

#############TOAST###########TOAST#################################
##

--on-error-stop

###
##

################################

--skip=*option*

all-frozen###

all-visible##

######################### none###

--startblock=*block*

#################### ## #######
###########--endblock#
########

--endblock=*block*

################### ## ########
#######TOAST###########
#######################--exclude-toast-pointers############TOAST###
#############TOAST##################

###############B##################

--checkunique

#####[54.123711]################amcheck#heapallindexed###################
################################## ##########################
##amcheck#checkunique################################1################

--heapallindexed

################amcheck#heapallindexed##################################
##################

--parent-check

######B##################################
#amcheck#bt_index_parent_check#######

#######amcheck#bt_index_check########--rootdescend###########
#bt_index_parent_check################

--rootdescend

###############amcheck#rootdescend#################################### ##############

###########--parent-check###############

############B############################ ################################ ############## ##

##

--parent-check#########--rootdescend############B################ ############################## ########INSERT#UPDATE#DELETE######### #################

##############################

-h *hostname*
--host=*hostname*

###################### ############Unix#######################

-p *port*
--port=*port*

TCP##########################Unix######################

-U
--username=*username*

###########

-w
--no-password

#################### #####################.pgpass######################### ######## ##

-W
--password

pg_amcheck#####################################

####################pg_amcheck### # ###pg_amcheck## ################### ###-W##############

--maintenance-db=*dbname*

--all#################### ## ################################### ################### ###################postgres############template1###

###############

-e
--echo

############SQL###############

`-j` *num*
`--jobs=`*num*

#####*num*#################1####1##################

#################

`-P`
`--progress`

############# ## #########
##

`-v`
`--verbose`

################ ##

`-V`
`--version`

pg_amcheck################

`--install-missing`
`--install-missing=`*schema*

#################################### ################################
#*schema*##############pg_catalog###############

###########amcheck###

`-?`
`--help`

pg_amcheck########################

##

#############PostgreSQL###########pg_amcheck#libpq####################32.15## ###

##########PG_COLOR######################## #####always#auto#never###

##

pg_amcheck #PostgreSQL 14.0##################

####

amcheck

pg_basebackup

pg_basebackup — PostgreSQL##################

##

`pg_basebackup` [*`option`*...]

##

pg_basebackup######PostgreSQL################################ ################## ################################### ##########################25.3############################## ###########26.2##############

######pg_basebackup## ################## ################################### ######## ################ ################## ################## ################################### ###### ################## #######pg_combinebackup################################### # ####################25.3.3###################25.3.5#########

#####[75.073314]#pg_basebackup## ############# ################################## ############################## ############# ######################pg_dump#################### ############ ##pg_basebackup## ######################## ####### ###################################### ###############pg_dump######### ############

##########################PostgreSQL############# #####`REPLICATION`###21.2### ####################ID############# ###`pg_hba.conf`########################### ### ######max_wal_senders##############1##walsender#WAL##########1############ ##################

###`pg_basebackup`######### ###########1################################

pg_basebackup### ######################### ######################################`max_wal_senders`#hot_standby### ##`pg_hba.conf`############# ########full_page_writes##############

###

- ##
- pg_basebackup########### ########WAL######################## `-X none`####### ##################### pg_basebackup#############WAL##### ################### ############# ##########WAL######################## `pg_switch_wal`######## ########
- #####################################
- ##############WAL################################## ############ #`full_page_writes`###############

pg_basebackup#######################`pg_stat_progress_basebackup`########### ######## ###27.4.6#########

#####

##############################

-D *directory*
--pgdata=*directory*

###################### pg_basebackup##################(#################### ####)####### ################################

#######tar##################-#########tar#####stdout#######

############

-F *format*
--format=*format*

########## *format*##################

p
plain

(########--tablespace-mapping#########)

##############

t
tar

############tar################ ##############base.tar################## ##################OID########tar############

##########-##################tar################ ###(###)gzip############ ### ####################WAL###########################

-i *old_manifest_file*
--incremental=*old_manifest_file*

########################## ################################ ############### ##

-R
--write-recovery-conf

standby.signal ###############tar#######################
#postgresql.auto.conf########### ####################################
##

######postgresql.auto.conf#####################pg_basebackup############
############## ##
dbname##dbname################################

-t *target*
--target=*target*

########################## ##########client######pg_basebackup############
#################### ######server:/some/path############################
####/some/path############## ############################
#pg_write_server_files########### ######blackhole####################
########## #####################################

WAL##############pg_basebackup#####################-Xstream############
###########################-Xfetch#-Xnone############

-T *olddir=newdir*
--tablespace-mapping=*olddir=newdir*

######*olddir*##################*newdir*######## ############*olddir*######### ############################## (###########*olddir*######################) ##*newdir*########################## ##############*newdir*################ ######################## *olddir#newdir*################### ######(=)######## ######################### ####################################

##

###########plain#############tar####################

--waldir=*waldir*

WAL(########)##################### ########WAL########pg_wal############ ############################ *waldir*############### ##############*waldir*## ##################################### ################plain###############

-X *method*
--wal-method=*method*

###WAL########################## ################################ ##none# ########WAL###########################postmaster############# ############## ################

###################*method*##########

n
none

#####################

f
fetch

############################## #############wal_keep_size################## ########################### ## #####

tar######################base.tar###########

s
stream

################################ ######2################################## ################ ##################1#####2####### ###################### ##

tar######################pg_wal.tar#######################10########### ##pg_xlog.tar##############

############

-z
--gzip

tar####################gzip######### tar##############################tar##### ######.gz###########

-Z *level*
-Z [{client|server}-]*method*[:*detail*]
--compress=*level*
--compress=[{client|server}-]*method*[:*detail*]

############### client###server########################### ############
#############CPU######## ######client####--target############## #######
################################ ######-Xstream##################WAL###
WAL#######################-Xfetch######

#####[81.668947]######gzip#lz4#zstd#########none##################0####
####0######gzip## ################### ######################## ####
#####keyword###keyword=value########################## ###############
##level#long###workers### ################################## #########
##gzip#lz4#zstd#none############none#######gzip###################0#
#######0######## ################### ######################## ###
#######*keyword*###*keyword=value*#################### ###############
#level#long#workers### ##################################

################################## ##############################0######
#gzip###########0##############

tar###gzip#lz4#zstd############.gz#.lz4#.zst########tar###############
plain## ###############
#########################

########-Xstream####################gzip############pg_wal.tar#gzip####
###

##

-c {fast|spread}
--checkpoint={fast|spread}

############fast(#####)###spread(#####)######(25.3.4#########)#

-C
--create-slot

##########--slot################################### ####################
##

-l *label*
--label=*label*

################ ###########pg_basebackup base backup################

-n
--no-clean

########pg_basebackup###
################################ ######################################

#####################################

-N
--no-sync

#######pg_basebackup############################ #######
#pg_basebackup###
############# ##

-P
--progress

############# ################################## ##########################
##################100%############## ###WAL############################
######### ################WAL#################

-r *rate*
--max-rate=*rate*

######################### ########pg_basebackup############## ############
##M###################### ##k#################### ########32###########
1024#########

################################## WAL#################fetch###########
##

-S *slotname*
--slot=*slotname*

########-X stream############ ###WAL#######################
####### ##
#primary_slot_name####################### ################################
##########################WAL###################

#######################-C##########################

###10######WAL###############
###################

--sync-method=*method*

###########fsync######pg_basebackup##################################
########################WAL##################################

######Linux#######syncfs###
############ #################pg_basebackup#WAL#######################
####### recovery_init_sync_method#########

##############--no-sync##################

-v
--verbose

############ ########################### ################################
##

--manifest-checksums=*algorithm*

#############
#NONE#CRC32C#SHA224#SHA256#SHA384#SHA512### ######CRC32C###

NONE################################ ####################################
########### ################SHA256############## SHA#######CRC32C######
CPU############1##############################

SHA###
##CRC32C###
####### ##
####################################

pg_verifybackup#####################################

--manifest-force-encode

##########################16########### ################UTF8###########16
############ ##

--no-estimate-size

#######################################
##pg_stat_progress_basebackup####backup_total####NULL#####

##############
#######################
##########

--progress######################

--no-manifest

######################## ####################pg_verifybackup################
############# #################WAL############################## #######
###########################

--no-slot

##################################

#########################-S######################(################)#######
###############

############
############WAL###

--no-verify-checksums

###

##############--no-
clean################################## ###########pg_stat_database###
########

##############################

-d *connstr*
--dbname=*connstr*

################################### ########################

##--dbname########pg_basebackup#####
##################################PostgreSQL########### ####PostgreSQL#######
################################## ##
###

-h *host*
--host=*host*

######################### ################Unix######################### ####
##############PGHOST######## ###########Unix#################

-p *port*
--port=*port*

###########TCP###########Unix###################### ###################
#PGPORT####################################

-s *interval*
--status-interval=*interval*

############################## ##
####################### ## #
######10####

-U *username*
--username=*username*

############

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

##############pg_basebackup#################

##################pg_basebackup#################################### ##
##pg_basebackup#################################### ################-W###
##############

#######################

-V
--version

pg_basebackup###############

-?
--help

pg_basebackup#######################

##

#######PostgreSQL###################libpq############32.15#########

####PG_COLOR####################### #####always#auto#never###

##

(####### --checkpoint=fast
#)#################### pg_basebackup #################

###
####PostgreSQL## ####################
PostgreSQL############################
################# ################################## (##########53.4##########)

plain#########--tablespace-mapping#################################### #
#######################plain##
#################

tar###############PostgreSQL##########tar#################### ###############
#######tar####################### #######################base.tar########
#tablespace_map#####################

pg_basebackup######9.1########################### ###WAL###########-X stream## #####9.3#################### ############tar###--format=tar#######9.5########### #########

pg_basebackup###

# #

mydbserver#################################/usr/local/pgsql/data#######

```
$ pg_basebackup -h mydbserver -D /usr/local/pgsql/data
```

##############tar#####1#########################backup############# ############## ####

```
$ pg_basebackup -D backup -Ft -z -P
```

#####################################bzip2#######

```
$ pg_basebackup -D - -Ft -X fetch | bzip2 > backup.tar.bz2
```

#######################################

/opt/ts#########./backup/ts#############################

```
$ pg_basebackup -D backup/data -T /opt/ts=$(pwd)/backup/ts
```

#####[93.650794]####9#gzip################1##tar############################## #backup#######

```
$ pg_basebackup -D backup -Ft --compress=gzip:9
```

####

pg_dump, 27.4.6

pgbench

pgbench — PostgreSQL##############

##

pgbench -i [*option*...] [*dbname*]

pgbench [*option*...] [*dbname*]

##

pgbench#PostgreSQL####################### ######SQL##########################
#################### ###############1######################## ######pgbench##1#
#########5##SELECT#UPDATE#INSERT##########TPC-B############## ############
##

pgbench###############

```
transaction type: <builtin: TPC-B (sort of)>
scaling factor: 10
query mode: simple
number of clients: 10
number of threads: 1
maximum number of tries: 1
number of transactions per client: 1000
number of transactions actually processed: 10000/10000
number of failed transactions: 0 (0.000%)
latency average = 11.013 ms
latency stddev = 7.351 ms
initial connection time = 45.758 ms
tps = 896.967014 (without initial connection time)
```

###7########################## 6##
################################## 8##
################################## ######################SQL###################
###-T############################ ##
#####(########################)# #####1###################

######TPC-B#################################### ##########################-i##
############pgbench############## ########################### ################
################## ##############

pgbench -i [*other-options*] *dbname*

###*dbname*######################## ##########################-h#-p#-U#######
#######

##
pgbench -i#4##### #pgbench_accounts#pgbench_branches#pgbench_history#pgbench_tellers# ###### ######################## ##############################

###########1######################

```
table                   # of rows
---------------------------------
pgbench_branches        1
pgbench_tellers         10
pgbench_accounts        100000
pgbench_history         0
```

-s### ###-F (fillfactor)###### #############

####################-i#########################

pgbench [*options*] *dbname*

########-c##########-t############-T (####)#-f##################

#####

###

###########

pgbench#########################

[-d] *dbname*
[--dbname=]*dbname*

: ################### #################PGDATABASE######## ################# ####################

-i
--initialize

: ##################

-I *init_steps*
--init-steps=*init_steps*

: ########################### *init_steps*#########1######################## ################## ######dtgvp### ###############

d (Drop)

: ###pgbench##############

t (create Tables)

: ###pgbench################
#pgbench_accounts#pgbench_branches#pgbench_history##
#pgbench_tellers#######

g###G (Generate data, ##############)

: ##################################

: #####[61.374795]#g(###########)######pgbench######################## ## #COPY#######/############## pgbench#######14########PostgreSQL######## ################################VACUUM######## g### ##pgbench_accounts#################100,000########1############## ## ####g#########################pgbench###################### ###COPY### #######/################ pgbench######################

##pgbench_accounts##############14###PostgreSQL#FREEZE############ #VACUUM######## g##########################100,000####1############### ######

G(#########)#############pgbench############################## ######## ##################################### G################################ ###

#########################(g###)######

v (Vacuum)

###########VACUUM#######

p (create Primary keys)

############################

f (create Foreign keys)

##################### ################################

-F *fillfactor*
--fillfactor=*fillfactor*

####fillfactor#pgbench_accounts#pgbench_tellers#pgbench_branches###### ##### ######100###

-n
--no-vacuum

################ #########-I##################v########

-q
--quiet

######5############################## #############100,000################## ############1###################

-I###G####################

-s *scale_factor*
--scale=*scale_factor*

################### ####-s 100# pgbench_accounts#####10,000,000########### ### ######1### #####20000## ######aid##########bigint################

--foreign-keys

#################### ############################f###########

--index-tablespace=*index_tablespace*

##

--partition-method=*NAME*

NAME###############pgbench_accounts########## ######range###hash### # #######--partitions#0################### ##############range###

--partitions=*NUM*

####################*NUM*####################pgbench_accounts########## ######0#####################

`--tablespace=tablespace`

####################################

`--unlogged-tables`

##################################

#############

pgbench##########################

`-b scriptname[@weight]`
`--builtin=scriptname[@weight]`

################################ ################tpcb-like#simple-update#select-only### ###################### #####list################## ##############

#######@######weight## ########1### ##############

`-c clients`
`--client=clients`

#################################### ######1###

`-C`
`--connect`

######################

`-D varname=value`
`--define=varname=value`

######################### ###-D#################

`-f filename[@weight]`
`--file=filename[@weight]`

filename##

#######@######weight## ######1### #@###################### filen@me@1 ################### ######

`-j threads`
`--jobs=threads`

pgbench############# #################CPU########### ################### ############ ######1###

`-l`
`--log`

############################## ##############

`-L limit`
`--latency-limit=limit`

limit###

#########(`--rate=...`)#*limit*## ################ ##################################

--max-tries##
#*limit*################ #####################--max-tries=0####### ######
##--max-tries######1## ########
####################################

-M *querymode*
--protocol=*querymode*

############################

- simple: ##################
- extended: ##################
- prepared: ################################
prepared######pgbench#######2#########################pgbench#############
####

#################### ########53###########

-n
--no-vacuum

####################
pgbench_accounts#pgbench_branches#pgbench_history#
pgbench_tellers#####################################

-N
--skip-some-updates

#####simple-update################### -b simple-update#######

-P *sec*
--progress=*sec*

sec################ #########################TPS##########################
#################################### (-R)##################################
#################################### --max-tries#########################
###

-r
--report-per-command

##############################(##############)############################
###################### ##################--max-tries######1##########

-R *rate*
--rate=*rate*

####1###################
##

##############################
##################### ##
##

####################
#################### ## ###############
###
###########

--latency-limit#--rate##
###################### ##

###
############
################# ################################

-s *scale_factor*
--scale=*scale_factor*

pgbench################## ################### ######pgbench_branches#####
############### ############-f##################################1#######
###

-S
--select-only

#####select-only#SELECT############### -b select-only#######

-t *transactions*
--transactions=*transactions*

####################### ######10###

-T *seconds*
--time=*seconds*

-t#-T######
####

-v
--vacuum-all

####4#################### -n#-v#####pgbench
#pgbench_tellers#pgbench_branches###########pgbench_history######
#######

--aggregate-interval=*seconds*

############## ###-l############ ######################################
##

--exit-on-abort

################################-
t###-T####################pgbench##################

Note that serialization failures or deadlock failures do not abort the client, so they are not affected by this option. ################################

--failures-detailed

###

- #####
- ########
  ################################

--log-prefix=*prefix*

--log################################### ######pgbench_log###

--max-tries=*number_of_tries*

####################
###########--latency-limit################## ###--latency-limit###--

time##########################(--max-tries=0)# #######1################## ##################### ##

--progress-timestamp

######-P#####################################Unix############## ############ ########### ##############################

--random-seed=*seed*

##################### ## *seed*# ########## time###################rand#################################### ###### ###########pgbench######random...##########################--rate## ############################ ################################ *seed*######## #############PGBENCH_RANDOM_SEED############## ########################## ##################################

##########################pgbench################ ######################## ##pgbench############ ###### ## ########### ### ##########

--sampling-rate=*rate*

1.0###############0.05##########5%################

####TPS########################## ###############0.01######TPS#1/100#########

--show-script=*scriptname*

#########*scriptname*##########################

--verbose-errors

#################################### ## ## ############################

#######

#####[76.388889]#pgbench################################ ######pgbench########### ####################################

--debug

##############

-h *hostname*
--host=*hostname*

##############

-p *port*
--port=*port*

###############

-U *login*
--username=*login*

######

```
-V
--version
```

pgbench################

```
-?
--help
```

pgbench#######################

#######

#####[84.220532]##############0####### #######1################################ ########### ##1###### ################ ####################2###### ######pgbench############## ################### 0 ### #### ####### 1 ## ################### ###############1####### ##################################2####### ######pgbench #--exit-on-abort############################

##

```
PGDATABASE
PGHOST
PGPORT
PGUSER
```

################

##################PostgreSQL###########libpq###################32.15###########

```
####PG_COLOR#######################
```

#####always#auto#never###

##

pgbench#######################?

pgbench################################ ####-b###########-f################## ######### ########@## #########1### ## #0##############

```
#######################-b tpcb-like################aid#tid#bid#delta#######
```

################7############# #######TPC-B#######################TPC-B######## ###########

1. BEGIN;
2. UPDATE pgbench_accounts SET abalance = abalance + :delta WHERE aid = :aid;
3. SELECT abalance FROM pgbench_accounts WHERE aid = :aid;
4. UPDATE pgbench_tellers SET tbalance = tbalance + :delta WHERE tid = :tid;
5. UPDATE pgbench_branches SET bbalance = bbalance + :delta WHERE bid = :bid;
6. INSERT INTO pgbench_history (tid, bid, aid, delta, mtime) VALUES (:tid, :bid, :aid, :delta, CURRENT_TIMESTAMP);
7. END;

`simple-update`##############`-N`#########4######5#################### ##TPC-B#########

`select-only`##############`-S`########`SELECT`#########

#######

pgbench##########################`-f`###

####################SQL#####1######## #####`--`############## ############pgbench#################################

> ##
>
> PostgreSQL#9.6################SQL###SQL#########################SQL###pgbench###SQL#####1###
>
> #####[83.544304]#pgbench#######SQL###pgbench#######SQL###

###`-D`###`-D`###298###`-D`###:*variablename*#####SQL############ 1###pgbench#1#####255#################

#298 pgbench Automatic Variables

##	##
client_id	##############################
default_seed	##############################
random_seed	##############`-D`############
scale	#####

#########################\###

\gset [*prefix*] \aset [*prefix*]

> ####################(`;`)#######SQL#################
>
> \gset################SQL######1###*prefix*####################
>
> \aset###########(\`;`######)########SQL#############################*prefix*###1####################
>
> \gset#\aset###
>
> ##############################*abalance*######*p_two*#*p_three*#3##2##################################2#################*four*#*five*########

```
UPDATE pgbench_accounts
  SET abalance = abalance + :delta
  WHERE aid = :aid
  RETURNING abalance \gset

-- 2#############
SELECT 1 \;
SELECT 2 AS two, 3 AS three \gset p_
SELECT 4 AS four \; SELECT 5 AS five \aset
```

\if *expression*
\elif *expression*
\else
\endif

#######psql#\if *expression*#######################################\set#############
#####

\set *varname expression*

varname###*expression*############## ##expression####NULL#######
#TRUE#FALSE#5432#########3.14159#################### :*variablename*####
SQL#####################SQL#CASE####################

###########NULL###NULL######

##############TRUE######NULL#FALSE###

#################################(+#-#*#/)####################

CASE####ELSE#################NULL###

#

```
\set ntellers 10 * :scale
\set aid (1021 * random(1, 100000 * :scale)) % \
           (100000 * :scale) + 1
\set divx CASE WHEN :x <> 0 THEN :y/:x ELSE NULL END
```

\sleep *number* [us | ms | s]

###############us######ms####s###*number*##
#############:*variablename*#########

#

```
\sleep 10 ms
```

\setshell *varname command* [*argument* ...]

command###########*argument*#######*varname*######## ######################
########

command####*argument*#################:*variablename*########### #######
#*argument*########*argument*######################

#:

```
\setshell variable_to_be_assigned command
 literal_argument :variable ::literal_starting_with_colon
```

\shell *command* [*argument* ...]

\setshell####################

#:

```
\shell command literal_argument :variable ::literal_starting_with_colon
```

```
\startpipeline
\syncpipeline
\endpipeline
```

##############SQL############### ######\startpipeline### ##\endpipeline############ ####################sync################## ####\syncpipeline################# ################################ # ##32.5######### ################################

#######

299#######################pgbench#########\set######### ################## #############2###

#299 pgbench####

#
boolean OR *boolean* → *boolean*   ##OR   5 or 0 → TRUE
boolean AND *boolean* → *boolean*   ##AND   3 and 0 → FALSE
NOT *boolean* → *boolean*   ##NOT   not false → TRUE
boolean IS [NOT] (NULL|TRUE|FALSE) → *boolean*   #######   1 is null → FALSE
value ISNULL|NOTNULL → *boolean*   NULL#######   1 notnull → TRUE
number = *number* → *boolean*   ##   5 = 4 → FALSE
number <> *number* → *boolean*   ##   5 <> 4 → TRUE
number != *number* → *boolean*   ##   5 != 5 → FALSE
number < *number* → *boolean*   #####

5 < 4 → FALSE
number <= *number* → *boolean* ## 5 <= 4 → FALSE
number > *number* → *boolean* ##### 5 > 4 → TRUE
number >= *number* → *boolean* ## 5 >= 4 → TRUE
integer | *integer* → *integer* #####OR 1 | 2 → 3
integer # *integer* → *integer* #####XOR 1 # 3 → 2
integer & *integer* → *integer* #####AND 1 & 3 → 1
~ *integer* → *integer* #####NOT ~ 1 → -2
integer << *integer* → *integer* ######## 1 << 2 → 4
integer >> *integer* → *integer* ######## 8 >> 2 → 2
number + *number* → *number* ## 5 + 4 → 9
number - *number* → *number* ## 3 - 2.0 → 1.0
number * *number* → *number* ## 5 * 4 → 20
number / *number* → *number* ##(################0########) 5 / 3 → 1

integer % *integer* → *integer* ##(##) 3 % 2 → 1
- *number* → *number* #### - 2.0 → -2.0

######

300######pgbench#########\set################

#300 pgbench###

abs (*number*) → ###### ### abs(-17) → 17
debug (*number*) → ###### ###stderr########## debug(5432.1) → 5432.1
double (*number*) → double ############ double(5432) → 5432.0
exp (*number*) → double ##(e######) exp(1.0) → 2.718281828459045
greatest (*number* [, ...]) → double if any argument is double, else integer ############## greatest(5, 4, 3, 2) → 5
hash (*value* [, *seed*]) → integer ###hash_murmur2###### hash(10, 5432) → -5817877081768721676
hash_fnv1a (*value* [, *seed*]) → integer FNV-1a####[1]###### hash_fnv1a(10, 5432) → -7793829335365542153
hash_murmur2 (*value* [, *seed*]) → integer MurmurHash2####[2]###### hash_murmur2(10, 5432) → -5817877081768721676
int (*number*) → integer

[1] https://en.wikipedia.org/wiki/Fowler%E2%80%93Noll%E2%80%93Vo_hash_function
[2] https://en.wikipedia.org/wiki/MurmurHash

int(5.4 + 3.8) → 9
least (*number* [, ...]) → double if any argument is double, else integer ############## least(5, 4, 3, 2.1) → 2.1
ln (*number*) → double #### ln(2.718281828459045) → 1.0
mod (*integer*, *integer*) → integer ##(##) mod(54, 32) → 22
permute (*i*, *size* [, *seed*]) → integer [0, size)#### *i* ##### #### *seed* ##############0...size-1########### #*i*#modulo *size*######### ########### permute(0, 4) → 0#3#####
pi () → double π#### pi() → 3.14159265358979323846
pow (*x*, *y*) → double power (*x*, *y*) → double *x*#*y*# pow(2.0, 10) → 1024.0
random (*lb*, *ub*) → integer [lb, ub]################ random(1, 10) → 1#10#####
random_exponential (*lb*, *ub*, *parameter*) → integer [lb, ub]################### random_exponential(1, 10, 3.0) → 1#10#####
random_gaussian (*lb*, *ub*, *parameter*) → integer [lb, ub]#################### random_gaussian(1, 10, 2.5) → 1#10#####
random_zipfian (*lb*, *ub*, *parameter*) → integer [lb, ub]#################### random_zipfian(1, 10, 1.5) → 1#10#####
sqrt (*number*) → double ### sqrt(2.0) → 1.414213562

random################## ############################
random_exponential#random_gaussian#####random_zipfian################
#######################

- #######parameter######### ###########parameter################### ###### #########

  f(x) = exp(-parameter * (x - min) / (max - min + 1)) / (1 - exp(-parameter))

  ######min#max############i#f(i) - f(i + 1)##########

  ######parameter#######min################max################## parameter#0########################### ############min######1%##### ##parameter%######### parameter################

- ####################################-parameter###+parameter############ ########### ######################## #######PHI(x)#################mu#(max + min) / 2.0#######

  f(x) = PHI(2.0 * parameter * (x - mu) / (max - min + 1)) /
  (2.0 * PHI(parameter) - 1)

  #####min#max##########i#######f(i + 0.5) - f(i - 0.5)###### ### ###parameter#############################min#max################### #67%## #####1.0 / parameter###########0.5 / parameter#######95%####2.0 / parameter###########1.0 / parameter######### ###parameter#4.0###67%### ######4##1(1.0/4.0)######3.0 / 8.0##5.0 / 8.0####95%#########2.0 / 4.0### #2###3############## #######parameter##2.0###

- random_zipfian################# parameter################# #### #parameter######################### ###1########k######k+1#######((k+1)/k)**parameter########## ####random_zipfian(1, ..., 2.5)###1#2# #(2/1)**2.5 = 5.66##########2#3##(3/2)**2.5 = 2.76###################

  pgbench#####Non-Uniform Random Variate Generation#Luc Devroye(Springer 1986, p. 550-551) #######parameter##[1.001, 1000]############

  ##

  ######################################ID################################ ########################

  #########permute###

######hash#hash_murmur2###hash_fnv1a######################## ############ ##:default_seed###################-D###########################

permute############################## [0, size)########################### ############### ##################################:default_seed########### ###########permute#################### ################################### ########## permute#random_zipfian#random_exponential################### ################################### ########pgbench########################### ##

```
\set size 1000000
\set r random_zipfian(1, :size, 1.07)
\set k 1 + permute(:r, :size)
```

###

```
\set k1 1 + permute(:r, :size, :default_seed + 123)
\set k2 1 + permute(:r, :size, :default_seed + 321)
```

#######hash###############

```
\set size 1000000
\set r random_zipfian(1, 100 * :size, 1.07)
\set k 1 + abs(hash(:r)) % :size
```

####hash##

#########TPC-B#######################

```
\set aid random(1, 100000 * :scale)
\set bid random(1, 1 * :scale)
\set tid random(1, 10 * :scale)
\set delta random(-5000, 5000)
BEGIN;
UPDATE pgbench_accounts SET abalance = abalance + :delta WHERE aid = :aid;
SELECT abalance FROM pgbench_accounts WHERE aid = :aid;
UPDATE pgbench_tellers SET tbalance = tbalance + :delta WHERE tid = :tid;
UPDATE pgbench_branches SET bbalance = bbalance + :delta WHERE bid = :bid;
INSERT INTO pgbench_history (tid, bid, aid, delta, mtime) VALUES
 (:tid, :bid, :aid, :delta, CURRENT_TIMESTAMP);
END;
```

###########################
################### ##############################

###############

#--aggregate-interval########-l###########pgbench#######################
#########*prefix.nnn*##*prefix*######pgbench_log#*nnn*#pgbench#####PID#
############--log-prefix################### -j######2###################
######################## ## ############
######*prefix.nnn.mmm*########*mmm*#1###############

##########1################### ##################################:

client_id

###########################

transaction_no

##############################

time

###################

script_no

#####################################-f###-b###############

time_epoch

##############Unix###########

time_us

#########################

schedule_lag

    ##--rate#
    #################

retries

    ## #--max-tries#1############

--rate#--latency-limit#########################*time*#skipped######### ###
###############*time*#failed######### --failures-detailed################
#######*time*##

################################

0 199 2241 0 1175850568 995598
0 200 2465 0 1175850568 998079
0 201 2513 0 1175850569 608
0 202 2038 0 1175850569 2663

--rate=100#--latency-limit=5###########(*schedule_lag*##############)

0 81 4621 0 1412881037 912698 3005
0 82 6173 0 1412881037 914578 4304
0 83 skipped 0 1412881037 914578 5217
0 83 skipped 0 1412881037 914578 5099
0 83 4722 0 1412881037 916203 3108
0 84 4142 0 1412881037 918023 2333
0 85 2465 0 1412881037 919759 740

##############82###(6.173###)#5################ ##2#############################
##############

############10#####################################

3 0 47423 0 1499414498 34501 3
3 1 8333 0 1499414498 42848 0
3 2 8358 0 1499414498 51219 0
4 0 72345 0 1499414498 59433 6
1 3 41718 0 1499414498 67879 4
1 4 8416 0 1499414498 76311 0
3 3 33235 0 1499414498 84469 3
0 0 failed 0 1499414498 84905 9
2 0 failed 0 1499414498 86248 9
3 4 8307 0 1499414498 92788 0

--failures-detailed################################

3 0 47423 0 1499414498 34501 3
3 1 8333 0 1499414498 42848 0
3 2 8358 0 1499414498 51219 0
4 0 72345 0 1499414498 59433 6
1 3 41718 0 1499414498 67879 4
1 4 8416 0 1499414498 76311 0
3 3 33235 0 1499414498 84469 3
0 0 serialization 0 1499414498 84905 9
2 0 serialization 0 1499414498 86248 9
3 4 8307 0 1499414498 92788 0

--sampling-
rate##

#######

--aggregate-interval################################## #####1############ ##############################:

interval_start

########Unix###########

num_transactions

############

sum_latency

##############

sum_latency_2

###############

min_latency

#############

max_latency

#############

sum_lag

###############--rate###########0#

sum_lag_2

################--rate###########0#

min_lag

##############--rate###########0#

max_lag

##############--rate###########0#

skipped

###########################--rate#--latency-limit###########0#

retried

################--max-tries#1######0#

retries

######################## #--max-tries#1######0#

serialization_failures

################################--failures-detailed###########0#

deadlock_failures

####################################--failures-detailed###########0#

#####[89.090909]############################ ############################

pgbench --aggregate-interval=10 --time=20 --client=10 --log --rate=1000 --latency-limit=10 --failures-detailed --max-tries=10 test

```
1650260552 5178 26171317 177284491527 1136 44462 2647617 7321113867 0 9866
 64 7564 28340 4148 0
1650260562 4808 25573984 220121792172 1171 62083 3037380 9666800914 0 9998
 598 7392 26621 4527 0
```

##
####### ##

######

-r###########pgbench#################

- latency—############### pgbench#######################
- ######## ############################
- ############################## ############################

####################--max-tries#####1############

##

###########################

```
starting vacuum...end.
transaction type: <builtin: TPC-B (sort of)>
scaling factor: 1
query mode: simple
number of clients: 10
number of threads: 1
maximum number of tries: 1
number of transactions per client: 1000
number of transactions actually processed: 10000/10000
number of failed transactions: 0 (0.000%)
number of transactions above the 50.0 ms latency limit: 1311/10000 (13.110
 %)
latency average = 28.488 ms
latency stddev = 21.009 ms
initial connection time = 69.068 ms
tps = 346.224794 (without initial connection time)
statement latencies in milliseconds and failures:
   0.012  0  \set aid random(1, 100000 * :scale)
   0.002  0  \set bid random(1, 1 * :scale)
   0.002  0  \set tid random(1, 10 * :scale)
   0.002  0  \set delta random(-5000, 5000)
   0.319  0  BEGIN;
   0.834  0  UPDATE pgbench_accounts SET abalance = abalance + :delta
 WHERE aid = :aid;
   0.641  0  SELECT abalance FROM pgbench_accounts WHERE aid = :aid;
  11.126  0  UPDATE pgbench_tellers SET tbalance = tbalance + :delta WHERE
 tid = :tid;
  12.961  0  UPDATE pgbench_branches SET bbalance = bbalance + :delta
 WHERE bid = :bid;
   0.634  0  INSERT INTO pgbench_history (tid, bid, aid, delta, mtime)
 VALUES (:tid, :bid, :aid, :delta, CURRENT_TIMESTAMP);
   1.957  0  END;
```

##(PGOPTIONS='-c default_transaction_isolation=serializable'pgbench.):

```
starting vacuum...end.
transaction type: <builtin: TPC-B (sort of)>
scaling factor: 1
query mode: simple
number of clients: 10
number of threads: 1
maximum number of tries: 10
number of transactions per client: 1000
number of transactions actually processed: 6317/10000
number of failed transactions: 3683 (36.830%)
number of transactions retried: 7667 (76.670%)
total number of retries: 45339
number of transactions above the 50.0 ms latency limit: 106/6317 (1.678 %)
latency average = 17.016 ms
latency stddev = 13.283 ms
initial connection time = 45.017 ms
tps = 186.792667 (without initial connection time)
statement latencies in milliseconds, failures and retries:
  0.006     0      0  \set aid random(1, 100000 * :scale)
  0.001     0      0  \set bid random(1, 1 * :scale)
  0.001     0      0  \set tid random(1, 10 * :scale)
  0.001     0      0  \set delta random(-5000, 5000)
  0.385     0      0  BEGIN;
  0.773     0      1  UPDATE pgbench_accounts SET abalance = abalance
 + :delta WHERE aid = :aid;
  0.624     0      0  SELECT abalance FROM pgbench_accounts WHERE aid
 = :aid;
  1.098   320   3762  UPDATE pgbench_tellers SET tbalance = tbalance
 + :delta WHERE tid = :tid;
  0.582  3363  41576  UPDATE pgbench_branches SET bbalance = bbalance
 + :delta WHERE bid = :bid;
  0.465     0      0  INSERT INTO pgbench_history (tid, bid, aid, delta,
 mtime) VALUES (:tid, :bid, :aid, :delta, CURRENT_TIMESTAMP);
  1.933     0      0  END;
```

###

################
#TPS########### ################################ #######################TPS####
##

#################

pgbench##################3########

- ############## ############pgbench########################### ################ ######:
  - pgbench######################
  - ##
  - ##
  - ##pgbench##################…##
- #####[63.020833]###/########### ############################ ## ## ############################ ##

################# ################################--exit-on-abort########
####################

- #####[78.055965]################ ##pgbench########pgbench##################
#######################...## ##
##
#######################################pgbench############## ###############
########pgbench##################...#####--exit-on-abort################
#########pgbench########### ##
#######################
################pgbench################

############################## ##
############## ###SQL###
########SQL## ####################
######## ######################################1#################1#############
#################pgbench########"########"###?########### ###################
#######################--max-tries################--latency-limit######
############--time############################## ###########################
##############################

> ##
>
> --max-tries###################1#####################################
> ################## #########(--max-tries=0)#--latency-limit########
> ################## ###--time##
>
> ################################## ######################################
> ###########
>
> ################################ SQL############\setshell############
> #######################

######################
##

############################## --max-tries######1###########################
############################## ### #######
####--max-tries######1####################

###--failures-detailed########### ##############(###########)#############--verbose-errors########### ##

#############

pgbench########################## ####PGOPTIONS###########PostgreSQL#########
##############(19.1.4#########)# ####pgbench###########wuzza####################
################

```
PGOPTIONS='-c default_table_access_method=wuzza'
```

#####

##################pgbench############### ################################

############################ -t###-T#######################################
################################

######TPC-B##################-s###############-c####################
pgbench_branches######-s######## ################1########### #####-c##-s#
###

########
################################ ##

pgbench##############################pgbench######################## #############
########pgbench#################### #################### ######################
########pgbench##########################

######

###pgbench###########
pgbench################################

pg_combinebackup

pg_combinebackup — reconstruct a full backup from an incremental backup and dependent backups

##

pg_combinebackup [*option*...] [*backup_directory*...]

##

######pg_combinebackup###
###

##############-o#
##########################

######pg_combinebackup###
#######################1###
############################## ####pg_combinebackup############################
################################## ######pg_verifybackup##########

######pg_combinebackup######################pg_combinebackup################# #
##

#####

-d
--debug

######stderr################

-n
--dry-run

######-n/--dry-run##################################pg_combinebackup###
########## ###--debug##################

-N
--no-sync

##############pg_combinebackup############################## ###########
###pg_combinebackup########## ##
############# ##

-o *outputdir*
--output=*outputdir*

#################################### #############

-T *olddir=newdir*
--tablespace-mapping=*olddir=newdir*

#################### *olddir* ######## *newdir* ###### *olddir*##############
##########################*newdir*############################## ##########
(=)######################### ############################

`--clone`

##reflinks###########
##########################

###
###########################

#############
#######pg_combinebackup############   ###Linux#Btrfs###XFS################
macOS#APFS###############Btrfs###XFS###############

`--copy`

`--copy`

#################### ##########`--copy-file-range`###`--clone`##########

`--copy-file-range`

#########################`copy_file_range`######   ####################
######`--clone`###   #
##Linux#FreeBSD###########

###`copy_file_range`########
##

`--manifest-checksums=`*`algorithm`*

######pg_basebackup#####pg_combinebackup############################ #######
#############
##NONE#CRC32C#SHA224#SHA256#SHA384#SHA512### ######CRC32C###

`--no-manifest`

############################# ##
##############

`--sync-method=`*`method`*

############fsync#######pg_combinebackup##############################
#########################WAL##################################

######Linux#####################################syncfs#####syncfs#######
#################pg_combinebackup#WAL##########################
recovery_init_sync_method#########

##############`--no-sync`##################

`-V`
`--version`

######pg_combinebackup################

`-?`
`--help`

######pg_combinebackup###########################

####

######pg_combinebackup################################ ####################
##
###

##################pg_checksums## #
##################pg_combinebackup###
#########

##

#######################PostgreSQL###########libpq################## #32.15######
#####

##########PG_COLOR######################## #####always#auto#never###

####

pg_basebackup

pg_config

pg_config — ##############PostgreSQL###########

##

pg_config [*option*...]

##

pg_config##########################PostgreSQL############## #######PostgreSQL### ##

#####

pg_config##################1########

`--bindir`

################ ####psql################## ######pg_config############## ########

`--docdir`

##############

`--htmldir`

HTML###############

`--includedir`

##############C################

`--pkgincludedir`

####C################

`--includedir-server`

############C################

`--libdir`

###########################

`--pkglibdir`

(################################ #############)

`--localedir`

#################### #PostgreSQL############################

`--mandir`

##################

`--sharedir`

##########################

`--sysconfdir`

#####################

`--pgxs`

###Makefile#########

`--configure`

PostgreSQL######configure################## ##########PostgreSQL#####################################
########## #############

`--cc`

PostgreSQL#########CC########## ####C############

`--cppflags`

PostgreSQL#########CPPFLAGS########## ##########C################# ####-I#######

`--cflags`

PostgreSQL#########CFLAGS########## C################

`--cflags_sl`

PostgreSQL#########CFLAGS_SL########## ##################C#############
###

`--ldflags`

PostgreSQL#########LDFLAGS########## ##############

`--ldflags_ex`

PostgreSQL#########LDFLAGS_EX########## ##############################

`--ldflags_sl`

PostgreSQL#########LDFLAGS_SL########## ##############################

`--libs`

PostgreSQL#########LIBS########## ######PostgreSQL##############-l#####
#####

`--version`

PostgreSQL############

`-?`
`--help`

pg_config#########################

1################################1##1############ ############################
######

##

#####--docdir#--pkgincludedir#--localedir#--mandir#--sharedir#--sysconfdir#--cc#--cppflags#--cflags#--cflags_sl#--ldflags#--ldflags_sl#--libs #PostgreSQL 8.1######### #####--htmldir#PostgreSQL 8.4###### ### #####--ldflags_ex#PostgreSQL 9.0#########

# #

####PostgreSQL#####################################

```
eval ./configure `pg_config --configure`
```

pg_config --configure### ########### #######eval######

pg_dump

pg_dump — PostgreSQL##################################

##

pg_dump [*connection-option*...] [*option*...] [*dbname*]

##

pg_dump#PostgreSQL######################## ##################################### #### pg_dump####################################

pg_dump################### ## ##########pg_dumpall##########

SQL################### ########################psql######### ################## ## ############SQL###################

##1#####################################pg_restore############## ########### ##pg_restore################################### ################################ #######

################pg_restore###############pg_dump########/############ #### ##pg_dump######################pg_restore### ############ ######################-Fc############-Fd#### ################### ##################################### ################################

pg_dump###

#####

################################

dbname

#################### ###############PGDATABASE####### #################### ##################

-a
--data-only

############################## ################################

########--section=data##############################

-b
--large-objects
--blobs (deprecated)

################## --schema#--table#--schema-only#################### #### ######-b### ################ ######--data-only##############--schema-only#########################

-B
--no-large-objects
--no-blobs (deprecated)

###################

-b#-B################################### -b############

-c
--clean

##DROP############ ################
################### ############################--if-exists############
######################

##########pg_restore##########
############

-C
--create

####################
--clean#######
#####################################

--create###
#ALTER DATABASE ... SET ...#ALTER ROLE ... IN DATABASE ... SET ...#
########## --no-acl####################################

##########pg_restore##########
############

-e *pattern*
--extension=*pattern*

pattern################# ##
###-e########################## *pattern*######psql#\d####################
################
##

pg_extension_config_dump######################--extension############
#######

##
-e########pg_dump## #### ##

-E *encoding*
--encoding=*encoding*

######################### ############################## ####
#PGCLIENTENCODING##################################### ###########
#23.3.1##########

-f *file*
--file=*file*

################ ################################ ################ ##########
############pg_dump##############
#########

-F *format*
--format=*format*

########## *format*##################

p
plain

###SQL#####################

c
custom

pg_restore######################### ##################################
################################ ###################

d
directory

pg_restore############################ ############################1####
#######pg_restore###
############ ###############Unix############### ####################
gzip#lz4#zstd################ ###########gzip########## ###############

t
tar

pg_restore########tar############## ##tar#################### tar#########
########################### ###tar############## ##tar###################
#######################

-j *njobs*
--jobs=*njobs*

njobs################################## ################################
################### ############################### ####################
#####################

pg_dump#*njobs*+1############### ###################max_connections#########
#####

################
###pg_dump###########
###(ACCESS SHARE)############# ############################## #########
################################
############ ################################## ###########################
###########pg_dump########NOWAIT#######################
pg_dump##############
#######

###############PostgreSQL
9.2################10########## ##
################### pg_dump -j################## ######################
################## ##
###################

-n *pattern*
--schema=*pattern*

pattern################### ################################ #############
##################################### ###-n##############################

pattern######psql#\d################################
############################## ##
############### #############

##

-n########pg_dump##
##

##

-n#####################################--large-objects########
########################

-N *pattern*
--exclude-schema=*pattern*

pattern################### #######-n################ -N###################
######################

-n#-N################1##-n#####-N########################## -n###-N######
###-N########################

-O
--no-owner

#################################### ########pg_dump##ALTER OWNER###
#SET SESSION AUTHORIZATION################################### ###########
###############
############-O########## ##

##########pg_restore#########
############

-R
--no-reconnect

####################################

-s
--schema-only

##############################

########--data-only##### ###--section=pre-data --section=post-data##
##############################

####--schema##################schema#####################

################################--exclude-table-data##########

-S *username*
--superuser=*username*

################################## ###--disable-triggers###############
##

-t *pattern*
--table=*pattern*

pattern##################### ###-t##############################
pattern######psql# \d###########(######)#################### ###########
################################## ####################################
################## #############

###########
#########################--include-foreign-data#######################
#######

-t########-n###-N############### -t#####################################
##################

##

-t########pg_dump##
##

-T *pattern*
--exclude-table=*pattern*

pattern################## #######-t############### -T####################
############################

-t#-T################1##-t##########-T######################### -t###-
T################-T#################

-v
--verbose

########### #########pg_dump##
######### ####################################

-V
--version

pg_dump################

-x
--no-privileges
--no-acl

#######grant/revoke###############

-Z *level*
-Z *method*[:*detail*]
--compress=*level*
--compress=*method*[:*detail*]

############################## ######gzip#lz4#zstd##############none#
########keyword##
#keyword=value#################### #################level#long###

################################## ############################0#######
#gzip###########0##############

##gzip###########
#########################gzip#lz4#zstd#########################
######## zstd#####long#######################################

tar##########################

--binary-upgrade

####
################################

--column-inserts
--attribute-inserts

##########INSERT#####INSERT INTO *table* (*column*, ...)VALUES...######
######## ################### ##PostgreSQL############################## #####
################################INSERT##############

--disable-dollar-quoting

##################################SQL####################

--disable-triggers

################################### ######################################
###pg_dump####### ##
##

#######--disable-triggers################################## #####-S#####
###

##########pg_restore#########
############

--enable-row-security

########pg_dump#row_security#off
############################## ##
########pg_dump#row_security#on##
##########

######COPY FROM##INSERT############
########

--exclude-extension=*pattern*

#####[75.085324]#*pattern*################## #######-n################ -N###
###################################### ######*pattern*################## ###
####-e############### --exclude-extension##############################
####

#####[68.571429]#-n#-N################1##-n#####-N#######################
-n###-N########-N#########################   ######-e#--exclude-
extension###################1##-e####################--exclude-
extension#####################   --exclude-extension#--exclude-
extension########--exclude-extension#####################

--exclude-table-and-children=*pattern*

###-T/--exclude-table###########*pattern*##############################
######

`--exclude-table-data=`*`pattern`*

pattern########################## #####-t#################### ########## ########################--exclude-table-data############## ########## ##

################################--schema-only#########

`--exclude-table-data-and-children=`*`pattern`*

###--exclude-table-data###########*pattern*########################### #########

`--extra-float-digits=`*`ndigits`*

################################extra_float_digits############ ######## ################################

`--filter=`*`filename`*

######include####################################### ######################### ########### ####### -t/--table#--table-and-children#-T/--exclude-table#--exclude-table-and-children# #######-n/--schema#-N/--exclude-schema# ############--include-foreign-data###########--exclude-table-data#--exclude-table-data-and-children######-e/--extension#--exclude-extension### STDIN################-###### --### ## ######################### ##########

################## ##### 1 ## 1 ##################

```
{ include | exclude } { extension | foreign_data | table |
 table_and_children | table_data | table_data_and_children |
 schema } PATTERN
```

2 ###########################

- ######extension:extensions# ###-e/--extension###--exclude-extension## ############
- ######foreign_###:########## ###--include-foreign-################ #######include################
- ##########:##### ###-t/--#######-T/--exclude-##################
- ##########_and_children:########################### ###--####-and-children###--exclude-####-and-children##############
- ##########_###:######################### ###--exclude-####-########## ###### #######exclude################
- ##########_###_and_children:## ######## ###--exclude-####-###-and-children############## ###### #exclude################
- ##########:schemas# ###-n/--#######-N/--exclude-##################

############################## ############################## ########## ##### ############################

#######################

`--if-exists`

DROP ... IF EXISTS#########--clean################ ################does not exist########## ########--clean#################

`--include-foreign-data=`*`foreignserver`*

foreignserver################################ ###--include-foreign-data######################### ###*foreignserver*######psql#\d############ ###################(############)## ## ## # ######################

> ##
>
> --include-foreign-data## ## ##

> ##
>
> --include-foreign-data########pg_dump######################## ## ####################################

`--inserts`

#####COPY#####INSERT############## ###################### ##PostgreSQL### ########################## ######################################INSERT## ############ ##################################### --column-inserts##### ######################

`--load-via-partition-root`

#########################COPY####INSERT################################## ######## ### ######################## ################################## #############text######################### ##############################

`--lock-wait-timeout=`*`timeout`*

########################## #######*timeout*###################### ###### #SET statement_timeout################### ############################ ##

`--no-comments`

############

`--no-publications`

################

`--no-security-labels`

#################

`--no-subscriptions`

#################

--no-sync

########pg_dump################################ ###########pg_dump########
#########
##

--no-table-access-method

############################ ##
################

#########pg_restore#########
###########

--no-tablespaces

###################### ##

#########pg_restore#########
###########

--no-toast-compression

TOAST################### ##################################

--no-unlogged-table-data

############################ ## #
############################ ##
####

--on-conflict-do-nothing

INSERT#####ON CONFLICT DO NOTHING###### ########--inserts#--column-inserts###--rows-per-insert#######################

--quote-all-identifiers

##################### #########pg_dump#######################PostgreSQL
######
##pg_dump## ####################
##################### ################################ --quote-all-identifiers##

--rows-per-insert=*nrows*

(COPY####)INSERT############### INSERT####1############### ######0#######
####### ######################################INSERT##############

--section=*sectionname*

############### ####pre-data#data#post-data############### ###########
######################### ####################

data## post-data######################
######################### pre-data#######################

--serializable-deferrable

##serializable############
#########serialization_failure#####################################

###################################### ##########################13########
###

###################################### ##################################
##################
################################### ################### ##################
################################

pg_dump### #############
###################################### ##################################
##

--snapshot=*snapshotname*

#################################### ###### 9.98#########

############################47##################################

###

--strict-names

#####[77.539062]####(-e/--extension)######(-n/--schema)########(-t/--table)
#########################1############################## ###############
############################--strict-names#####pg_dump################
########(-e/--extension)#####(-n/--schema)#####(-t/--table)#########
##################1####/####/############### ###--filter################
###/####/###################pg_dump#--strict-names################

#####[75.767918]########-N/--exclude-schema#-T/--exclude-table#--
exclude-table-data##########   ####################################
#############--exclude-extension#-N/--exclude-schema#-T/--
exclude-table#-T/--exclude-table-data########## ####################
##########

--sync-method=*method*

###########fsync######pg_dump --format=directory###################
#############

######Linux#######syncfs##
########## recovery_init_sync_method#########

###############--no-sync###########--format#directory###############
####

--table-and-children=*pattern*

###-t/--table###########*pattern*##
##

--use-set-session-authorization

##################ALTER   OWNER##########SQL#SET   SESSION
AUTHORIZATION########## ##
############# ###SET SESSION AUTHORIZATION############################
############# ALTER OWNER##################

-?
--help

pg_dump#########################

####################################

-d *dbname*
--dbname=*dbname*

#################### ########################*dbname*############# *dbname*####
########## ######################################

-h *host*
--host=*host*

######################### ################Unix########################### ####
##############PGHOST######## ###########Unix#################

-p *port*
--port=*port*

###########TCP###########Unix######################### ####################
#PGPORT####################################

-U *username*
--username=*username*

#########

-w
--no-password

############### ####################.pgpass##############################
##

-W
--password

##############pg_dump#################

##################pg_dump#################################### ####pg_dump##
#################################### ################-W################

--role=*rolename*

######################### ##########pg_dump##############SET ROLE *rolename*#
################ ###########-U###########pg_dump###########################
################### ##
##################################

##

PGDATABASE
PGHOST
PGOPTIONS
PGPORT
PGUSER

################

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL###########libpq##################32.15###########

##

pg_dump####SELECT######## pg_dump################psql#####################select# ############## ###libpq###

##pg_dump##################################### ##########PGOPTIONS###ALTER USER#########track_counts################

##

##############template1######################pg_dump######################### ########## ## ############################ ##template1####template0########## ##########

```
CREATE DATABASE foo WITH TEMPLATE template0;
```

##############--disable-triggers############pg_dump######################## ## ############################## #################

pg_dump### ################ ####################ANALYZE############## ####24.1.3###24.1.6##########

pg_dump##########PostgreSQL#################pg_dump####pg_dump################# PostgreSQL########################### ###pg_dump############PostgreSQL######### ########## #########9.2################ ####pg_dump####################PostgreSQL ################## ################################ ###pg_dump############## ## ######################## ## ############--quote-all-identifiers############### ######PostgreSQL################################## #############

#####[94.639556]############################pg_dump#connect = false###### ####CREATE SUBSCRIPTION## ############### ## ################### ############### ## ################### ######################### ############################two_phase = false# #################### ################two_phase = true################### ##two_phase############################

# #

mydb############SQL################

```
$ pg_dump mydb > db.sql
```

newdb##################################

```
$ psql -d newdb -f db.sql
```

##############################

```
$ pg_dump -Fc mydb > db.dump
```

###########################

```
$ pg_dump -Fd mydb -f dumpdir
```

5####################directory################

```
$ pg_dump -Fd mydb -j 5 -f dumpdir
```

newdb#######################################

```
$ pg_restore -d newdb db.dump
```

##

```
$ pg_restore -d postgres --clean --create db.dump
```

mytab####################

```
$ pg_dump -t mytab mydb > db.sql
```

detroit#########emp#################### ####employee_log################

```
$ pg_dump -t 'detroit.emp*' -T detroit.employee_log mydb > db.sql
```

east###west####gsm################### ####test################

```
$ pg_dump -n 'east*gsm' -n 'west*gsm' -N '*test*' mydb > db.sql
```

#################################

```
$ pg_dump -n '(east|west)*gsm' -N '*test*' mydb > db.sql
```

ts_#######################################

```
$ pg_dump -T 'ts_*' mydb > db.sql
```

#################-t################################## ######################
######## ## ########################1
#########################

```
$ pg_dump -t "\"MixedCaseName\"" mydb > mytab.sql
```

#####[53.142857]#detroit#########emp#################### ####employee_log##
############## ######mytable2#######mytable################################
#filter.txt#######

```
include table mytable*
exclude table mytable2
```

```
$ pg_dump --filter=filter.txt mydb > db.sql
```

####

pg_dumpall, pg_restore, psql

pg_dumpall

pg_dumpall — PostgreSQL##########################

##

pg_dumpall [*connection-option*...] [*option*...]

##

pg_dumpall#########PostgreSQL########1#################################### ### ##########################psql###############SQL############# ################### #pg_dump############## ####pg_dumpall## ############################## #pg_dump####################

pg_dumpall###
####### ##

SQL################## ####################-f/--file#######################

pg_dumpall##PostgreSQL## ############### ################### ########~/.pgpass################ ###32.16##########

#####

##############################

-a
--data-only

##############################

-c
--clean

#######################################DROP##SQL########## ################
################### ########################--if-exists###################
###################

-E *encoding*
--encoding=*encoding*

######################### ################################ ############
#PGCLIENTENCODING############################

-f *filename*
--file=*filename*

################# ####################

--filter=*filename*

#####--exclude-database#####
########## STDIN################-####### --filter####################--
exclude-database#####################################

#############1 ## 1 ######## ####################

exclude database *PATTERN*

############################## ############################## ########## #####
############################

-g
--globals-only

##

-O
--no-owner

##################################### ########pg_dumpall#ALTER OWNER###
#SET SESSION AUTHORIZATION######################### ################
#######################
###-O##

-r
--roles-only

##################################

-s
--schema-only

################################

-S *username*
--superuser=*username*

################################ ###--disable-triggers###############
##

-t
--tablespaces-only

##################################

-v
--verbose

########### #########pg_dumpall##
##################################### ########pg_dump########

-V
--version

pg_dumpall################

-x
--no-privileges
--no-acl

###########grant/revoke############

--binary-upgrade

####
##############################

--column-inserts
--attribute-inserts

##########INSERT#####INSERT INTO *table* (*column*, ...) VALUES...###### ##### ################### ###PostgreSQL##############################

--disable-dollar-quoting

##################################SQL####################

--disable-triggers

################################## ### #########pg_dumpall####### ### ##########

#######--disable-triggers################################## #####-S##### ##

--exclude-database=*pattern*

###*pattern*#################### ###--exclude-database################## ######### *pattern*######psql#\d###########(######)########################## ## ############################ ##########################

--extra-float-digits=*ndigits*

##################################extra_float_digits############# ################ ########################

--if-exists

DROP ... IF EXISTS#########--clean################ #####does not exist#### ####### ########--clean#################

--inserts

#####COPY#####INSERT############# #################### ##PostgreSQL###### ######################## ##################################### ########## ##--column-inserts############

--load-via-partition-root

##########################COPY#INSERT#################################### ##### ## ################### ##################################### ############text######################## ##

--lock-wait-timeout=*timeout*

################################## ####*timeout*########################### # ######SET statement_timeout####################

--no-comments

############

--no-publications

################

`--no-role-passwords`

################# ###############NULL################################
##################################pg_authid############pg_roles###
####### ####pg_authid##

`--no-security-labels`

#################

`--no-subscriptions`

#################

`--no-sync`

########pg_dumpall############################## ###########pg_dumpall#
##
###

`--no-table-access-method`

############################ ##
################

`--no-tablespaces`

################################## ######################################
###########

`--no-toast-compression`

TOAST#################### ##################################

`--no-unlogged-table-data`

######################## ##################################### ##############
#########

`--on-conflict-do-nothing`

INSERT#####ON CONFLICT DO NOTHING###### ########--inserts###--
column-inserts#######################

`--quote-all-identifiers`

#################### #########pg_dumpall#########################PostgreSQL
######
##pg_dumpall## ##################
####################### ################################## --quote-all-
identifiers##

`--rows-per-insert=`*`nrows`*

(COPY####)INSERT############### INSERT####1############### ######0#######
######## ######################################INSERT##############

`--use-set-session-authorization`

ALTER OWNER###########SQL#SET SESSION AUTHORIZATION########### ######
###

-?
--help

pg_dumpall#######################

##################################

-d *connstr*
--dbname=*connstr*

##

################################--dbname###### ###pg_dumpall###########
################################## ####################################
#####################################-l#########

-h *host*
--host=*host*

############################## ################Unix#######################
##################PGHOST###################Unix#################

-l *dbname*
--database=*dbname*

########
##postgres####### ############template1#######

-p *port*
--port=*port*

###########TCP###########Unix######################   ###################
#PGPORT####################################

-U *username*
--username=*username*

########

-w
--no-password

############### ###################.pgpass##############################
###

-W
--password

##############pg_dumpall#################

#################pg_dumpall#################################   ##
##pg_dumpall## ################-W######
###########

############################## ######################~/.pgpass#########
####

--role=*rolename*

########################   #########pg_dumpall#############SET   ROLE
rolename################ ###########-U###########pg_dumpall################

```
    ############################## ##########################################
    ##############################################
```

##

```
PGHOST
PGOPTIONS
PGPORT
PGUSER

    ###############

PG_COLOR

    ###################### #####always#auto#never###
```

####################PostgreSQL###########libpq##################32.15###########

##

pg_dumpall#####pg_dump###################pg_dump#########

```
--clean################################################## --clean###########
#####postgres###template1##################################################
############################## ##########################################

#################################################ANALYZE############        #
##vacuumdb -a -z###########################

################################## ####################CREATE ROLE#########
#############################################################role already exists#######
########## ################### --clean#######################################
#########--if-exists###################
```

pg_dumpall## ###########################
###################

# #

```
##########################

$ pg_dumpall > db.out

##################################

$ psql -f db.out postgres

#####################################pg_dumpall################################
########################  ###--clean###########postgres##################
#####################################################
```

####

```
################pg_dump##########
```

pg_isready

pg_isready — PostgreSQL#############

##

`pg_isready` [*connection-option*...] [*option*...]

##

pg_isready#PostgreSQL############################## ####################

#####

`-d` *dbname*
`--dbname=`*dbname*

   ################### *dbname*############# ################################## ######

`-h` *hostname*
`--host=`*hostname*

   ######################### ###############Unix#########################

`-p` *port*
`--port=`*port*

   #############TCP######Unix#################### ##########PGPORT######### ########################5432###

`-q`
`--quiet`

   ############## #################

`-t` *seconds*
`--timeout=`*seconds*

   ################################## ################ ######3####

`-U` *username*
`--username=`*username*

   #########*username*##################

`-V`
`--version`

   pg_isready###############

`-?`
`--help`

   pg_isready#########################

#######

pg_isready#####################0#######################1################
#2##############################3#########

##

#######PostgreSQL##########pg_isready#libpq###############32.15#########

####PG_COLOR#################### #####always#auto#never###

##

##############################
############

# #

##############

```
$ pg_isready
/tmp:5432 - accepting connections
$ echo $?
0
```

####PostgreSQL#######################

```
$ pg_isready -h localhost -p 5433
localhost:5433 - rejecting connections
$ echo $?
1
```

#####PostgreSQL#######################

```
$ pg_isready -h someremotehost
someremotehost:5432 - no response
$ echo $?
2
```

pg_receivewal

pg_receivewal — PostgreSQL####################

##

`pg_receivewal` [*option*...]

##

pg_receivewal#####PostgreSQL################################ ##25.3#######################################

pg_receivewal#####################################archive_command#archive_library#################################### #####pg_receivewal#########archive_timeout##############

PostgreSQL##########WAL#########pg_receivewal#########WAL################WAL############ WAL###################`--synchronous`################## pg_receivewal#WAL########synchronous_commit#`remote_apply`#################### ##### ## ######## ##synchronous_standby_names############pg_receivewal#########`application_name`######## ########`synchronous_commit`###`remote_apply`##############

############PostgreSQL## ##### #`REPLICATION`###21.2#################################### ###`pg_hba.conf`###max_wal_senders#################

####################pg_receivewal#################

1. ###WAL###WAL############### ##########

2. ##`READ_REPLICATION_SLOT`########## ####`restart_lsn`################ #########PostgreSQL 15#######################################

3. ######################`IDENTIFY_SYSTEM`############WAL###############

#######################################pg_receivewal## ############`-n`###############

############pg_receivewal#SIGINT#**Control**+**C**####SIGTERM####################

#####

`-D` *`directory`*
`--directory=`*`directory`*

###############

#############

`-E` *`lsn`*
`--endpos=`*`lsn`*

#######LSN##################################0#######

lsn#######LSN#######################

--if-not-exists

--create-slot##

-n
--no-loop

############## ################

--no-sync

#######pg_receivewal#WAL########################## ################
###########WAL################### ##########################WAL######
###################

#######--synchronous#######

-s *interval*
--status-interval=*interval*

########################## ############################## ###############
########## ### ############

-S *slotname*
--slot=*slotname*

pg_receivewal#######################(26.2.6########)# ###########
##pg_receivewal################## ##
###################

pg_receivewal##
################WAL################ ##################################
###################### ################--synchronous####################
####

--synchronous

WAL######################## ############--status-interval###########
################

pg_receivewal###
########################

-v
--verbose

############

-Z *level*
-Z *method*[:*detail*]
--compress=*level*
--compress=*method*[:*detail*]

##################

#####[81.754386]######gzip#(PostgreSQL#--with-lz4##########)lz4#########
#####none### ##################### ######################## #######
##keyword###keyword=value##################### ################level#
###########gzip#lz4#PostgreSQL#--with-lz4####################
#none####### ##################### ######################## ##########

#*keyword*###*keyword=value*######################## ################level##
###

################################### ##############################0#######
#gzip###########0##############

gzip##########################.gz#######lz4#############.lz4#######

##################################

-d *connstr*
--dbname=*connstr*

################################### ##########################

#####[54.460094]################################--dbname###### ##
##pg_receivewal## ###########
##############################--dbname#######pg_receivewal################
######################PostgreSQL########## ####PostgreSQL##################
######################

-h *host*
--host=*host*

######################## ###############Unix######################## #####
#####PGHOST###########################Unix################

-p *port*
--port=*port*

###########TCP######Unix###################### ##########PGPORT##########
#####################################

-U *username*
--username=*username*

##########

-w
--no-password

############# ####################.pgpass################################
##

-W
--password

pg_receivewal##############################

################# pg_receivewal##################################### ###
pg_receivewal#################################### -W######################
##########

pg_receivewal###########################2#######1########

--create-slot

--slot####################################

--drop-slot

--slot##################################

#######################

-V
--version

pg_receivewal################

-?
--help

pg_receivewal###########################

#######

pg_receivewal#SIGINT###SIGTERM#################0####### ################# #### ########### ##################################

##

#######PostgreSQL####################libpq############32.15#########

####PG_COLOR####################### #####always#auto#never###

##

WAL#################archive_command#archive_library#####pg_receivewal############ ###################### ###########archive_command#archive_library################ ###WAL## ####### ####WAL## ##############

pg_receivewal##################################WAL######################

# #

#########mydbserver##########################/usr/local/pgsql/archive## #####

```
$ pg_receivewal -h mydbserver -D /usr/local/pgsql/archive
```

####

pg_basebackup

pg_recvlogical

pg_recvlogical — PostgreSQL##################

##

pg_recvlogical[*option*...]

##

pg_recvlogical##

######################pg_receivewal####################(#47####)###########

pg_recvlogical#########SQL#######peek#get################# ################ ################## ##########################pg_logical_slot_peek_changes# ########

############pg_recvlogical#SIGINT#**Control**+**C**####SIGTERM###################

#####

##########################1##############

--create-slot

  ##################--slot########--plugin###########--dbname########## #########

  --two-phase#--create-slot#######################################

--drop-slot

  --slot################################

--start

  --slot### ################ ##########################--no-loop#######################

  ##

  ##

--create-slot#--start############### --drop-slot####################

###

-E *lsn*
--endpos=*lsn*

  --start############################LSN################0####### --start### ########################

  LSN#*lsn*################################

  --endpos### ############ ## ########################

-f *filename*
--file=*filename*

##################################### stdout#######-######

-F *interval_seconds*
--fsync-interval=*interval_seconds*

##################################pg_recvlogical#fsync()##################

##################
##############

0###########fsync()#################################### ################
############

-I *lsn*
--startpos=*lsn*

--start#################LSN####### ############47####53.4########## ####
############

--if-not-exists

--create-slot##

-n
--no-loop

##################################

-o *name*[=*value*]
--option=*name*[=*value*]

#####*name*#(########)######*value*############## ##############################
####

-P *plugin*
--plugin=*plugin*

#47###########
#########

-s *interval_seconds*
--status-interval=*interval_seconds*

########pg_receivewal#################### ################

-S *slot_name*
--slot=*slot_name*

--start######*slot_name*############################ --create-slot######
############ --drop-slot####################

-t
--two-phase

#####[84.955752]######################### #######--create-slot########
############################### ########--create-slot##############

-v
--verbose

#############

################################

-d *dbname*
--dbname=*dbname*

############ ###################### *dbname*############# ################## ###################### ############

-h *hostname-or-ip*
--host=*hostname-or-ip*

######################## ##############Unix###################### ######### ###PGHOST#########################Unix###############

-p *port*
--port=*port*

############TCP############Unix###################### ##########PGPORT## ######################################

-U *user*
--username=*user*

############# ########OS#######

-w
--no-password

############# ################.pgpass################################### ### ##

-W
--password

pg_recvlogical##############################

##################pg_recvlogical#################################### ###### #######################pg_recvlogical################## #############-W###### ###############

####################

-V
--version

pg_recvlogical#################

-?
--help

pg_recvlogical##########################

#######

pg_recvlogical#SIGINT###SIGTERM#################0####### ################## #### ########### ##################################

##

##################PostgreSQL##########libpq################## (32.15#########)#

```
####PG_COLOR#######################
```
```
#####always#auto#never###
```

##

pg_recvlogical##################################WAL#######################

#

######47.1##########

####

pg_receivewal

pg_restore

pg_restore — pg_dump####################PostgreSQL#############

##

pg_restore [*connection-option*...] [*option*...] [*filename*]

##

pg_restore##pg_dump#############################PostgreSQL#################### ##### ## ###pg_restore############ ### ################################## ######

pg_restore#####2########### ################pg_restore############################## ######### ######################################SQL########################## ################ ###########pg_dump############## ####################pg_dump# #################

######pg_restore#################################### ###########INSERT######## ##################################pg_restore#COPY######################

#####

pg_restore####################

filename

####################

-a
--data-only

##

#######--section=data################################

-c
--clean

######################################DROP############ ################# ############# ############################--if-exists####################

-C
--create

--clean#####################################

--create###pg_restore#### ################ALTER DATABASE ... SET ...#ALTER ROLE ... IN DATABASE ... SET ...############ --no-acl########################### ###########

############-d##############DROP DATABASE#CREATE DATABASE########### ###### ##

-d *dbname*
--dbname=*dbname*

dbname############################ *dbname*############# ################ ######################

-e
--exit-on-error

#######SQL########################### ################################### #####

-f *filename*
--file=*filename*

##########-l############################## stdout(####)#######-#########

--filter=*filename*

#########################-n/--schema####################-N/--schema###############-P/--function##################-I/--index################-T/--trigger#### ########### STDIN################-####### --filter##

############1 ## 1 ######## ####################

{ include | exclude } { function | index | schema | table | trigger } *PATTERN*

2

- ######function: -P/--function############## #######include########### ####
- ######index: ####### -I/--indexes############## #######include######### ######
- ######schema: ##### -n/--schema#-N/--exclude-schema##############
- ######table: -t/--table################### #######include############## #######
- ######trigger: ###### -T/--trigger############## #######include####### ########

############################## ############################## ########## ##### ############################

-F *format*
--format=*format*

############### pg_restore############################### ###################

c
custom

######pg_dump#################

d
directory

##########################

t
tar

######tar###############

-I *index*
--index=*index*

######################### ###-I##########################

-j *number-of-jobs*
--jobs=*number-of-jobs*

pg_restore##*number-of-jobs*#####
############## ##
##

#####1#######1####(###############)### #####################

####CPU#################
#########CPU######## ##################################### ###############
############################

############################
###################### #########--single-transaction###################
####

-l
--list

#################### ###############-L######################## -n#-t######
#######-l##########################

-L *list-file*
--use-list=*list-file*

list-file###################### ######################### -n#-t#######
#####-L############################

list-file#########-l################ ######################(;)##########
######### #############

-n *schema*
--schema=*schema*

########################### ###-n######################### ###############
######-t###################

-N *schema*
--exclude-schema=*schema*

########################### -N####################################

########-n#-N############-N######################

-O
--no-owner

##################################### #######pg_restore##ALTER OWNER## #SET SESSION AUTHORIZATION######################### ################# ### -O################# ##

-P *function-name(argtype [, ...])*
--function=*function-name(argtype [, ...])*

################ ## ###-P# #####################

-R
--no-reconnect

#######################################

-s
--schema-only

##

#######--data-only##### #######--section=pre-data --section=post-data################################

####--schema#################schema####################

-S *username*
--superuser=*username*

################################### ###--disable-triggers##############

-t *table*
--table=*table*

#################################### ################################## ############## ###-t############################## -n################### ##########

##

-t########pg_restore## # ##

##

######pg_dump#-t################### #######pg_restore################# ########-t################## ####pg_dump#-t######################## ################pg_restore#-t######################

##

PostgreSQL#9.6###

-T *trigger*
--trigger=*trigger*

##################### ###-T#########################

-v
--verbose

############ #########pg_restore##
############ #####################################

-V
--version

pg_restore#################

-x
--no-privileges
--no-acl

#######grant/revoke#################

-1
--single-transaction

####################################BEGIN/COMMIT####### ####################
################################ ########--exit-on-error#######

--disable-triggers

##################################### ##########pg_restore####################
###################################### ##
##

#######--disable-triggers################################## #####-S#####
#########################PostgreSQL###########pg_restore############

--enable-row-security

#######pg_restore#row_security#off
##################################   ######################################
########pg_restore#row_security#on##
##

COPY FROM#####################################INSERT#####################

--if-exists

DROP ... IF EXISTS##########--clean################ #################does
not exist########### #########--clean#################

--no-comments

##

--no-data-for-failed-tables

####
################################## ################################### ####
PostGIS###PostgreSQL################################### ####################
####################

##############################SQL##################

`--no-publications`

##

`--no-security-labels`

##

`--no-subscriptions`

##

`--no-table-access-method`

#####[86.144578]############################ ##############################
############################ ################################## ###########
##

`--no-tablespaces`

###################### ##
###

`--section=`*`sectionname`*

################# ####`pre-data`#`data`#`post-data`############### ###########
######################### ####################

data################################ post-data##############################
################## pre-data#########################

`--strict-names`

######(`-n`/`--schema`)########(`-t`/`--table`)#################1##############
##############

`--transaction-size=`*`N`*

############################## ###########*N*################### ########--
exit-on-error######

######`--transaction-size`##########SQL######1#########-1/`--single-
transaction##################1##################### `--single-
transaction##
##############--
transaction-size##

`--use-set-session-authorization`

`ALTER OWNER`###########SQL#`SET SESSION AUTHORIZATION`################
######### ##

`-?`
`--help`

pg_restore#######################

pg_restore################################

`-h` *`host`*
`--host=`*`host`*

######################## ################Unix######################### ####
##############`PGHOST`######## ##########Unix################

-p *port*
--port=*port*

##########TCP##########Unix##################### ##################
#PGPORT##################################

-U *username*
--username=*username*

########

-w
--no-password

############## ##################.pgpass##############################
##

-W
--password

#############pg_restore################

#################pg_restore################################## ####pg_restore
##################################### ################-W################

--role=*rolename*

######################### #########pg_restore#############SET ROLE
rolename################ ##########-U##########pg_restore################
############################## ##
###

##

PGHOST
PGOPTIONS
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL##########libpq##################32.15##########
####################PGDATABASE#########

##

-d################################pg_restore####SQL######## pg_restore##########
#####psql#################################### ###libpq#########################
################

##

template1######################pg_restore################################ #####
################################## ############################template1###
#template0########## #########

```
CREATE DATABASE foo WITH TEMPLATE template0;
```

pg_restore###########

- ###################--disable-triggers###########pg_restore################ ### ################ ##############################
- pg_restore## ############ ############################## ##-L#-t################################

pg_dump###########pg_dump############

##ANALYZE############### ### #24.1.3###24.1.6##########

mydb##

```
$ pg_dump -Fc mydb > db.dump
```

##########################

```
$ dropdb mydb
$ pg_restore -C -d postgres db.dump
```

-d#################################### pg_restore##mydb####CREATE DATABASE# ############################## -C##

newdb############################

```
$ createdb -T template0 newdb
$ pg_restore -d newdb db.dump
```

-C################### ########################## ############template1### #template0######################## ################

###

```
$ pg_restore -l db.dump > db.list
```

##################1################

```
;
; Archive created at Mon Sep 14 13:55:39 2009
;     dbname: DBDEMOS
;     TOC Entries: 81
;     Compression: 9
;     Dump Version: 1.10-0
;     Format: CUSTOM
;     Integer: 4 bytes
;     Offset: 8 bytes
;     Dumped from database version: 8.3.5
;     Dumped by pg_dump version: 8.3.8
;
;
; Selected TOC Entries:
```

```
;
3; 2615 2200 SCHEMA - public pasha
1861; 0 0 COMMENT - SCHEMA public pasha
1862; 0 0 ACL - public pasha
317; 1247 17715 TYPE public composite pasha
319; 1247 25899 DOMAIN public domain0 pasha
```

################## ##############################ID######

##########

```
10; 145433 TABLE map_resolutions postgres
;2; 145344 TABLE species postgres
;4; 145359 TABLE nt_header postgres
6; 145402 TABLE species_records postgres
;8; 145416 TABLE ss_old postgres
```

#######pg_restore################10#6#######################

```
$ pg_restore -L db.list db.dump
```

####

pg_dump, pg_dumpall, psql

pg_verifybackup

pg_verifybackup — PostgreSQL#######################

##

pg_verifybackup [*option*...]

##

pg_verifybackup##pg_basebackup##
#backup_manifest################ #######"plain"##################"tar"######
##############

pg_verifybackup###
#################### ##
###################### #######pg_verifybackup################################

#####[72.446556]##########4########### #1
##pg_verifybackup#backup_manifest######### ##############################
#########################pg_verifybackup############## ################4####
####### ###pg_verifybackup#backup_manifest########### ####################
#####################pg_control#####################################
##pg_verifybackup##############

#2##pg_verifybackup###
################ 2#3############################## #####
#postgresql.auto.conf#standby.signal#recovery.signal################
###################
############################backup_manifest#####pg_wal############### ####
###

###pg_verifybackup###
#################################### ####################################
############# ##########################

###pg_verifybackup##
####### backup_manifest##############################pg_verifybackup####
#########################pg_waldump####### pg_waldump####################
##--quiet######### ##
##################################### ####################################
############################

###################WAL##pg_waldump#
################ ###WAL###
#pg_verifybackup######pg_waldump########### ##################
##backup_manifest###########################

#####

pg_verifybackup####################

-e
--exit-on-error

  ###################### ##################pg_verifybackup###############
  ############################

-i *path*
--ignore=*path*

##########################backup_manifest#################################
#################### ## ######
###########
#######

-m *path*
--manifest-path=*path*

##

-n
--no-parse-wal

##################################

-P
--progress

################# #####################################

##############--quiet############

-q
--quiet

#########################

-s
--skip-checksums

##################### ########################## ##########################
######

-w *path*
--wal-directory=*path*

pg_wal################WAL############## ######WAL###################
############

##############

-V
--version

pg_verifybackup################

-?
--help

pg_verifybackup##########################

#

mydbserver################################

```
$ pg_basebackup -h mydbserver -D /usr/local/pgsql/data
$ pg_verifybackup /usr/local/pgsql/data
```

```
mydbserver##########################################################

$ pg_basebackup -h mydbserver -D /usr/local/pgsql/backup1234
$ mv /usr/local/pgsql/backup1234/backup_manifest /my/secure/location/
backup_manifest.1234
$ pg_verifybackup -m /my/secure/location/backup_manifest.1234 /usr/local/
pgsql/backup1234

#######################################################

$ pg_basebackup -h mydbserver -D /usr/local/pgsql/data
$ edit /usr/local/pgsql/data/note.to.self
$ pg_verifybackup --ignore=note.to.self --skip-checksums /usr/local/pgsql/
data
```

####

pg_basebackup

psql

psql — PostgreSQL########

##

psql [*option*...] [*dbname* [*username*]]

##

psql##PostgreSQL################ ################PostgreSQL################### ### ################################## ####psql################################ ################################

#####

-a
--echo-all

########################## (######################) ###ECHO###all######## #######

-A
--no-align

################### ############aligned(######)#### ###\pset format unaligned######

-b
--echo-errors

####SQL################## ###ECHO###errors###########

-c *command*
--command=*command*

psql##############*command*############ ###################-f############## ########## -c###-f#######psql################### ##########-c#######-f## #################

command##################psql##################################1######## ### #####-c#######SQL#psql################### ##############-c########### ##################psql###### ####

```
psql -c '\x' -c 'SELECT * FROM foo;'
```

####

```
echo '\x \\ SELECT * FROM foo;' | psql
```

#######\\################

-c######SQL########################### ###################BEGIN/COMMIT# ###########################SQL######################1################## ####################################53.2.2.1##########

1#######################################-c#####################echo####### ######################psql##################

```
psql <<EOF
\x
SELECT * FROM foo;
EOF
```

--csv

CSV(#######)############ ###\pset format csv#####

-d *dbname*
--dbname=*dbname*

#################### ######################*dbname*################
dbname############# ####################################

-e
--echo-queries

############SQL############### ECHO###queries################

-E
--echo-hidden

\d###################################### #######psql################ ####
#ECHO_HIDDEN#on################

-f *filename*
--file=*filename*

############*filename*############# ###################-c################
######## -c###-f#######psql################## ##########-c#######-f####
############### ###############\i#############

filename#-##################EOF########\q############### #############
###################### ########Readline#################-n##############

##############psql < *filename*################### ###################
###-f## ########################
################## ##
##############

-F *separator*
--field-separator=*separator*

separator################################ \pset fieldsep####\f######
####

-h *hostname*
--host=*hostname*

######################## ################Unix#########################

-H
--html

HTML############ ####\pset format html####\H#########

-l
--list

########################## ######################## \list################

##-d#########################
#############psql##postgres############

-L *filename*
--log-file=*filename*

#################################*filename*########

-n
--no-readline

###########Readline#######(################)#

-o *filename*
--output=*filename*

############*filename*############ ###\o##############

-p *port*
--port=*port*

##############TCP###########Unix######################## ####PGPORT#########
##########################5432##############

-P *assignment*
--pset=*assignment*

\pset################## ###################################### ##########
LaTeX######-P format=latex#######

-q
--quiet

psql######################### #############welcome#################### ####
############################ -c############### #####QUIET#on##############
###

-R *separator*
--record-separator=*separator*

separator################################### ###\pset recordsep######

-s
--single-step

################## ##
#######################

-S
--single-line

#####################################SQL################

##
###1## SQL##

-t
--tuples-only

###################### ####\t###\pset tuples_only######

-T *table_options*
--table-attr=*table_options*

HTML#table################## ###\pset tableattr#########

-U *username*
--username=*username*

############*username*################ ########################

-v *assignment*
--set=*assignment*
--variable=*assignment*

\set#################### ######################=#################### ## ######################### ################################ ############# ######### ##############################

-V
--version

psql################

-w
--no-password

############## ##################.pgpass################################ ####### ##

####\connect###################

-W
--password

################################psql################

###################.pgpass##############################psql############## ## ####psql##################################### ################-W######## ########

#############\connect###########

-x
--expanded

################## ###\x###\pset expanded######

-X
--no-psqlrc

#######psqlrc###########~/.psqlrc##################

-z
--field-separator-zero

################################### ###\pset fieldsep_zero######

-0
--record-separator-zero

################################## ######xargs -0########### ###\pset recordsep_zero######

-1
--single-transaction

########1####-c######-f######################### #####psql################ #BEGIN###################COMMIT################ ######################### ################ON_ERROR_STOP##########ROLLBACK########### ### ##

#######BEGIN#COMMIT#ROLLBACK################################## ######### ##

-?
--help[=*topic*]

psql################# ######*topic*###########options##psql############ ##### commands#psql#################options#psql#################### ##variables#psql###################

########

psql############0##psql#######################################1################ #################2##ON_ERROR_STOP#############################3##########

####

###########

psql#PostgreSQL#################### ### ################################## psql### #-d######-h##########-p#########-U############## ######################### ##################################### ### ## ###########psql#Unix#######################Windows##localhost#TCP/IP##### #### ######################### ## ################OS######## ################################### ############## ################################## ##################################

##################################PGDATABASE#PGHOST#PGPORT#PGUSER######## ######## ##############32.15########## ###~/.pgpass######################### ####### ###32.16#########

################*conninfo*######URI###### ###################### ############## ################### #########

```
$ psql "service=myservice sslmode=require"
$ psql postgresql://dbmaster:5433/mydb?sslmode=require
```

#################32.18#####LDAP############ ########################32.1.2#### ######

##psql##################

#######################psql##############auto####### #########Unix######## #LC_CTYPE####################### ##################PGCLIENTENCODING####### #######################

SQL#######

##########psql##psql##################=>################### ##########

```
$ psql testdb
psql (17beta3)
Type "help" for help.

testdb=>
```

###########SQL################ ### ## ################# ### ################ ###################

###search_path###### ################### options=-csearch_path=############SELECT pg_catalog.set_config('search_path', '', false)###SQL############## ## ###psql############# ###SQL#########################

###############psql#LISTEN#NOTIFY#######################

C#####################################SQL#######psql###########

######

psql##psql######psql################ #### ####################################psql########### ########################### #############

psql## ################################## #####

#################### ##2######### ## ###########C################## #########\n#####\t#####\b (##)#\r(##)#\f (# #)#*digits*#8##########\x*digits*#16############### ########################### ##

SQL##################################:#######psql######################## # ########:'*variable_name*'###:"*variable_name*"############## : {?*variable_name*}###################### ###TRUE#FALSE######### ######### ###################

#########`################################ ################################## ########### ###############:*variable_name*######*variable_name*#psql######## ## ##:'*variable_name*'################# ## ################################### ################## ## ##:'*variable_name*'##########################

#####################SQL############## ###########SQL########## ######### ######################"## ########## ######2#######1############## ####FOO"BAR"BAZ#fooBARbaz######"A weird"" name"#A weird" name######

#######################1#################### ############################### #### \\########2##########################SQL####################### #### ##SQL#####psql#####1#################### ################################### ##

########################## ########SQL#################################### ######### ##

##################

\a

################################ ######################## ################## ####### ##########\pset##########

\bind [*parameter*] …

###$1##########

#:

```
INSERT INTO tbl1 VALUES ($1, $2) \bind 'first value' 'second value' \g
```

####\gx#\gset#####\g####################

###########psql####################(53.1.2###)############################## ## ############psql############################## ################################ ############################## ## ####################

\c ### \connect [-reuse-previous=*on*|*off*] [*dbname* [*username*] [*host*] [*port*] | *conninfo*]

PostgreSQL################ ###############(1#######################)#### #*conninfo*##################32.1.1####### ################################### ###

dbname#username#host#port########-##########################

###*sslmode*############ ## ############################*conninfo*#################### #####-reuse-previous=on####-reuse-previous=off########################### ### ##*conninfo*################ ######################### ####*host*#################################### #*hostaddr*############ ## ## ################################libpq##############

######################### ##################################psql########### ################ ## ################## ##########################\connect## ## ### ####################\connect## ################################ #############\connect################## #\connect##################################

##

```
=> \c mydb myuser host.dom 6432
=> \c service=foo
=> \c "host=localhost port=5432 dbname=mydb connect_timeout=10
 sslmode=disable"

=> \c -reuse-previous=on sslmode=require    -- sslmode####
=> \c postgresql://tom@localhost/mydb?application_name=myapp
```

\C [*title*]

##\pset title *title*#
########## ############(caption)######## ###HTML##############################

\cd [*directory*]

############*directory*####### ##############################

###
##################\! pwd#########

\conninfo

#######################

\copy { *table* [(*column_list*)] } from { *'filename'* | program *'command'* | stdin | pstdin } [[with] (*option* [, ...])] [where *condition*]
\copy { *table* [(*column_list*)] | (*query*) } to { *'filename'* | program *'command'* | stdout | pstdout } [[with] (*option* [, ...])]

####################### ###COPY SQL#####################################
#########psql## #######################
###################SQL##################

program#########*command*#psql#######*command*######*command*##############
########## ##############################SQL##################

\copy ... from stdin#######################\.##################
EOF############ ########SQL################################ \copy ... to stdout#####psql####################COPY *count*################(#
#######################)# ########\o##########psql####################from pstdin####to pstdout########

##########SQL#COPY########## #######################COPY###### ####\copy#
################################## ##############################\copy#####
##############################

###
\copy ... to############SQL#COPY ... TO STDOUT########\g *filename*#\g |*program*########## \copy############################## ###################

###
##########/###SQL #COPY################## #######SQL############### ############## ##CSV####\copy ... from#######\.#######################

\copyright

PostgreSQL#################

\crosstabview [*colV* [*colH* [*colD* [*sortcolH*]]]]

##############\g#################### ##########3############# *colV*###
################*colH*#################### *colD*################## #####
#*sortcolH*####################

##############1################## ##########SQL####################### #
######*colV*##1#*colH*##2###### *colH*#*colV*################ *colD*#############
########3###########*colV*##*colH*######*colD*######

##################*colV*#####################################

#######1#######*colH*######################## ###########################
########*sortcolH*###########*colH*######*sortcolH*##################
######*sortcolH*##################

##########*colH*########x#*colV*########y########(x,y)################
#*colH*###x#*colV*###y#####colD######## #################### #############
#########

\d[S+] [*pattern*]

pattern###
###############################NOT NULL################### #############
#################### ######################## ############################
#######

##############\d################## ######################################
################

\d+############################ ############################OID#######
################################replica identity################################
#######

############################## #########################S############

##

\d#*pattern*#############\dtvmsE################################
########################## ##############

\da[S] [*pattern*]

##################################### *pattern*############################
########################S##########
##

\dA[+] [*pattern*]

################# *pattern*### ######
##+##

\dAc[+] [*access-method-pattern* [*input-type-pattern*]]

###############(36.16.1########)# *access-method-pattern*##################
################################### *input-type-pattern*##################
############################ ########+####################################
#####

\dAf[+] [*access-method-pattern* [*input-type-pattern*]]

############(36.16.5########)# *access-method-pattern*################### ############################ *input-type-pattern*#################### ##################### #######+################################

\dAo[+] [*access-method-pattern* [*operator-family-pattern*]]

####################(36.16.2########)# *access-method-pattern*############ ### *operator-family-pattern*###### ################################### #######+#########(########)######### #####

\dAp[+] [*access-method-pattern* [*operator-family-pattern*]]

#######################(36.16.3########)# *access-method-pattern*######### ## *operator-family-pattern*#### ##################################### #######+############################ #######

\db[+] [*pattern*]

############## *pattern*## ######+##### ##

\dc[S+] [*pattern*]

######################## *pattern*################################### ####### ######################## ########################S############ ######+#### ##########################

\dconfig[+] [*pattern*]

####################### *pattern*### ## *pattern*## (############## #\dconfig *#########) ######+## ##(#######################)##################

\dC[+] [*pattern*]

############## *pattern*## # #####+############################

\dd[S] [*pattern*]

constraint#operator class#operator family#rule#trigger############ ############ ##

\dd#*pattern*#################### ###################################### # ################################ ############################## ############ #############S############

##########COMMENT SQL####################

\dD[S+] [*pattern*]

############# *pattern*###################################### ############## ################ ##############################S########### ########+#### ########################

\ddp [*pattern*]

####################### ##1######## *pattern*##

ALTER DEFAULT PRIVILEGES############################ ########5.8#######

\dE[S+] [*pattern*]
\di[S+] [*pattern*]
\dm[S+] [*pattern*]
\ds[S+] [*pattern*]
\dt[S+] [*pattern*]
\dv[S+] [*pattern*]

############E#i#m#s#t#v###
##
##\dti################## +#################################
(permanent#temporary####unlogged)############################ *pattern*########
############################ ############################### ###############
##########S#############

\des[+] [*pattern*]

##(foreign)########external servers########### *pattern*##########################
########### \des+###

\det[+] [*pattern*]

##(foreign)#########external tables########### *pattern*##########################
################## \det+################################

\deu[+] [*pattern*]

###########external users########### *pattern*##################################
\deu+##############################

##

\deu+################################## ###########################

\dew[+] [*pattern*]

##############external wrappers########### *pattern*############################
############## \dew+##

\df[anptwS+] [*pattern* [*arg_pattern* ...]]

##########################agg# (##)##normal###procedure###trigger###window######
################ #######################a#n#p#t#w############# *pattern*#
##################################### ####################1##2#############
(################################ ########*arg_pattern*########-######
###) ############################## ##########################S###########
\df+###C############
################### #############\sf##############

\dF[+] [*pattern*]

############### *pattern*################################### \dF+############
##

\dFd[+] [*pattern*]

############### *pattern*################################### \dFd+#########
###

\dFp[+] [*pattern*]

############### *pattern*################################### \dFp+######## ##

\dFt[+] [*pattern*]

#################### *pattern*### \dFt +##

\dg[S+] [*pattern*]

################# (######################################\du###########) ################################ ############S############ *pattern*####### ################################ \dg+################################## ### ##################

\dl[+]

\lo_list####################### ######+################################## ############

\dL[S+] [*pattern*]

############# *pattern*############################## ################### ###### ########################S############ +########################### ##

\dn[S+] [*pattern*]

################### *pattern*################################## ############# ################# #######S############################## ########+######### ################################

\do[S+] [*pattern* [*arg_pattern* [*arg_pattern*]]]

######################## *pattern*################################ *arg_pattern*#1## *arg_pattern*#2##### ################################ (#####################-#########) ######### ##################### ########################S############ ######+##### ##

\dO[S+] [*pattern*]

############ *pattern*################################## ################### ########### ########################S############ ########+############### ######################## ### ####### ####################################

\dp[S] [*pattern*]

#################################### *pattern*############################## ###### ############################## ##########################S########### ##

###########GRANT#####REVOKE########### ############5.8#######

\dP[itn+] [*pattern*]

########################### *pattern*## # ##t(####)#i(######)############################## ########################## ###############

###n (#nested#)##
#######################

########+## n#+###########
####2########1##############################1############################
#############

\drds [*role-pattern* [*database-pattern*]]

##################### ############################## *role-pattern*##
#*database-pattern*############################## ###############*#######
##

#########################ALTER ROLE###ALTER DATABASE##########

\drg[S] [*pattern*]

############(ADMIN#INHERIT#SET)## ##
############GRANT#############

##S############ #############
################################

\dRp[+] [*pattern*]

######################### *pattern*##
########+##

\dRs[+] [*pattern*]

########################### *pattern*######################################
########+############################

\dT[S+] [*pattern*]

############ *pattern*############################## +####################
######enum###################### ############################## ###########
##############S############

\du[S+] [*pattern*]

################## (######################################\dg############)
pattern## ###########################
############S############ \du+##################################
##################

\dx[+] [*pattern*]

################### *pattern*############################## \dx+##########
#####################################

\dX [*pattern*]

############## *pattern*##

######################(####Ndistinct)################ defined############
##########NULL################ ANALYZE############################
##pg_stats_ext#####

\dy[+] [*pattern*]

############### *pattern*################################### +##############
######################

\e###\edit [*filename*] [*line_number*]

filename####################### ####################################
filename## ####################
##################################

#############
############psql#############1########### ##################### #############
#############
############# #############\g#########################\r#########

#######1########################## ######################################
#################### ################################## #########\i#######
###

###(line_number)#########psql################################## ##########1###
########psql##################################

###
###

\echo *text* [...]

############################## ############################### ###########

```
=> \echo `date`
Tue Oct 26 21:40:57 CEST 1999
```

################-n##################(##############)#

###
\o##################################\qecho################   \warn#############

\ef [*function_description* [*line_number*]]

#######################CREATE OR REPLACE FUNCTION#CREATE OR REPLACE PROCEDURE################ ###\edit########### ######################### ### ## ######################### ####\g##########\r#########

##################foo(integer, text)#################### #############
####################

###############CREATE FUNCTION##################

############psql######################## ##################################

###########################\ef#####################################

###
###

\encoding [*encoding*]

###

\errverbose

#########################VERBOSITY#verbose#####SHOW_CONTEXT#always#
####################

\ev [*view_name* [*line_number*]]

##################CREATE OR REPLACE VIEW################## ###\edit###
########### ########################## ##################################
########## ############################\g##########\r#########

################CREATE VIEW################

###########psql#######################

##########################\ev#####################################

\f [*string*]

#################################### #########|#### ###\pset fieldsep##
####

\g [(*option=value* [...])] [*filename*]
\g [(*option=value* [...])] [|*command*]

###########################

\g###############*option=value*################################\pset
option value################################ ###########=############
################ =*value*############*option*######*value*###\pset *option*#
###########

filename#|*command*###
########## ### ##################
#SQL##############

#######\g######################
\g###########\o###########################\pset####################
########

######|#######################*command*##################################
########################

\gdesc

################################ ######################################
###############

##

\getenv *psql_var env_var*

####*env_var*######psql##*psql_var*####### psql########*env_var*#########
##*psql_var*######## #:

=> **\getenv home HOME**

```
=> \echo :home
/home/postgres
```

\gexec

###SQL########## ####my_table########
#################


```
=> SELECT format('create index on my_table(%I)', attname)
-> FROM pg_attribute
-> WHERE attrelid = 'my_table'::regclass AND attnum > 0
-> ORDER BY attnum
-> \gexec
CREATE INDEX
CREATE INDEX
CREATE INDEX
CREATE INDEX
```

##########################2############################## NULL###########
################################psql##################psql###############
###########################ON_ERROR_STOP###################
##ECHO######## #\gexec#####ECHO#all####queries######################
##########################(timing)##

###

\gset [*prefix*]

##############################psql################## ########################
##############


```
=> SELECT 'hello' AS var1, 10 AS var2
-> \gset
=> \echo :var1 :var2
hello 10
```

prefix###


```
=> SELECT 'hello' AS var1, 10 AS var2
-> \gset result_
=> \echo :result_var1 :result_var2
hello 10
```

#####NULL############################

################################

###

\gx [(*option=value* [...])] [*filename*]
\gx [(*option=value* [...])] [|*command*]

\gx##\pset########expanded=on#####################################
#\g###### \x##########

\h###\help [*command*]

####SQL################### *command*############psql######################
command#######*#########SQL#################

##########################\help####################################

##

#######################################**\help alter table**#########
###

\H###\html

HTML################ HTML###
############### ##############\pset#########

\i###\include *filename*

filename#####################################

filename#-#########EOF########\q######################## ############
######################## Readline##
###

##

###################ECHO###all############

\if *expression*
\elif *expression*
\else
\endif

################################ #######\if#####\endif############# ####
#\elif####################\else############### ########################
##

\if######\elif################################ #####true################
##############\elif#\else###\endif################ ####\if###\elif####
######################\elif########################### \else##############
#####\if#\elif######################

\if######\elif#####*expression*###
#####on/off#################### ########true#false#1#0#on#off#yes#no##
################################ ####t#T#tR####true##########

################################

###
#\if#\elif#\else#\endif######################## ########################
##################################

#########\if###################
########\include#################psql###########

#######

-- ########2####################

```
-- #####2##psql########
SELECT
    EXISTS(SELECT 1 FROM customer WHERE customer_id = 123) as
 is_customer,
    EXISTS(SELECT 1 FROM employee WHERE employee_id = 456) as
 is_employee
\gset
\if :is_customer
    SELECT * FROM customer WHERE customer_id = 123;
\elif :is_employee
    \echo 'is not a customer but is an employee'
    SELECT * FROM employee WHERE employee_id = 456;
\else
    \if yes
        \echo 'not a customer or employee'
    \else
        \echo 'this will never print'
    \endif
\endif
```

\ir###\include_relative *filename*

\ir#####\i###\ir###

\l[+] ### \list[+] [*pattern*]

###*pattern*###+###

\lo_export *loid filename*

########OID#*loid*#################*filename*######## ###lo_export######## ####### lo_export###

###
##########OID########\lo_list#########

\lo_import *filename* [*comment*]

#####PostgreSQL##

```
foo=> \lo_import '/home/peter/pictures/photo.xcf' 'a picture of me'
lo_import 152801
```

##########################ID 152801################# ### ##\lo_list######OID###############

###lo_import##lo_import################

\lo_list[+]

####################PostgreSQL###+###

\lo_unlink *loid*

OID#*loid*##########################

###
##########OID########\lo_list#########

\o###\out [*filename*]
\o###\out [|*command*]

############*filename*##########################*command*############## ####
############################

###|#######################*command*################################ ####
####################

###\d###
##

###
######################\qecho#########

\p###\print

######################## ##

\password [*username*]

################################ ############################ALTER ROLE#
############### ##

\prompt [*text*] *name*

##*name*######################## #####*text*################## ##############
############################

#######\prompt############# ####-f####################\prompt#########
#######

\pset [*option* [*value*]]

#####################################*option*######################*value*#
#################### ########################*value*########################
##############*value*###########################

########\pset############################

####################

border

value############## ##
########### HTML##########border=...######### ##############0#####
###1########2#########3###########2#######border = 2###########
latex###latex-longtable######################3##############

columns

wrapped## ######## ###############COLUMNS###COLUMNS########################## ######### ###columns######wrapped######################### columns################################

csv_fieldsep

CSV######################## ########################CSV############ ############## ###########

expanded (###x)

value##########(expanded)#############on###off#####auto############# ## *value*################################### ########################## #######2########## ################(horizontal)######################### ##(auto)##############2## #### ######################## ################################

fieldsep

################################## ##################################### ###### ######################\pset fieldsep '\t'####### ############ ####'|'#######

fieldsep_zero

#######################################

footer

value######################(*(n* rows)#)######on########off############ ### *value*################################

format

#####aligned#asciidoc#csv#html#latex#latex-longtable#troff-ms#unaligned#wrapped########### ##################

aligned#################################### ###########

unaligned########1##1################################## ############### ### ############################## ##################CSV###############

csv### RFC 4180[1]################################ #########COPY#####CSV ########### ############tuples_only#on########### ################ # #########################Unix##################(\n)####Microsoft Windows# #########(\r\n)### ################\pset csv_fieldsep######

wrapped###aligned## #### #columns######################## psql########################## ### #wrapped##########################aligned############

asciidoc#html#latex#latex-longtable###troff-ms################## ######################## ################# HTML################LaTeX ######################### latex###LaTeX#tabular####### latex-longtable####LaTeX#longtable###booktabs###########

[1] https://datatracker.ietf.org/doc/html/rfc4180

linestyle

########ascii#old-ascii###unicode########### ################### ################ ########ascii### #######aligned###wrapped######## ###

ascii#####ASCII###### ############+######### wrapped############2# ###################(.)#######################

old-ascii#####ASCII#######PostgreSQL 8.4################## ########## #####:########### ##########################;###########

unicode###Unicode############ ########################### ############ ######################################

border############linestyle######################## ###ASCII####### ###########Unicode##########################

null

null################### ################ ######################## ###\pset null '(null)'########

numericlocale

value###########10################################on#######off###### ######### *value*#####################################

pager

#######psql############################## pager#####off############# ####### pager#####on## # #pager#####always############ ##################################### ## *value*#####\pset pager###############

####PSQL_PAGER###PAGER################################ ############ ###############more##########

\watch#########################Unix#########PSQL_WATCH_PAGER####### ##################### ##################################psql########## ######(pspg --stream##)###############

pager_min_lines

pager_min_lines### ########## ########0###

recordsep

################################### ############

recordsep_zero

#####################################

tableattr (### T)

HTML#########table################ ######### ##cellpadding#bgcolor############ border#####\pset border######## ###########border############## *value*########################

latex-longtable####### ########################## ################ #'0.2 0.2 0.6'######## ########################

title (### C)

############################## ######################### *value*######## ############

tuples_only (### t)

value#########################on#######off############### *value*#### ############################## ##################################### # ############################

unicode_border_linestyle

unicode###########single###double###########

unicode_column_linestyle

unicode##########single###double###########

unicode_header_linestyle

unicode############single###double###########

xheader_width

#############full########column#page##################

full: ################################

column: ###################

page: ################

###: #################

####################################

###
\pset#################### \a#\C#\f#\H#\t#\T#\x#########

\q###\quit

psql########### ############################

\qecho *text* [...]

########\echo###### #######\o################################

\r###\reset

######################

\s [*filename*]

psql###########*filename*####### *filename*#######################(########## #####)# ########psql#Readline#########################

\set [*name* [*value* [...]]]

psql###*name*#*value*#######value######################### ####################### ############ ############\unset#############

##########\set##########psql################

############################## ################## ################

psql## ####################
######

##
#######SQL#SET#############

\setenv *name* [*value*]

####*name*#*value*####### *value*########################## ##########

```
testdb=> \setenv PAGER less
testdb=> \setenv LESS -imx4F
```

\sf[+] *function_description*

########CREATE OR REPLACE FUNCTION#####CREATE OR REPLACE PROCEDURE#
################################## ######\o##########################

#################foo(integer, text)#################### ##############
####################

######+##################1################

##########################\sf####################################

\sv[+] *view_name*

##################CREATE OR REPLACE VIEW################# #############
############# ###\o#######

########+#############1###########

##########################\sv####################################

\t

######################## #######\pset tuples_only####################

\T *table_options*

HTML#######table################ #######\pset tableattr *table_options*##########

\timing [*on* | *off*]

############SQL##############on###off####### ##############on#off######### ##
############ 1######## #:# ##############################

\unset *name*

psql##*name*################

psql###########################\unset########################## ##########
#####

\w###\write *filename*
\w###\write |*command*

##############*filename*######################*command*######### ##########
##############################

###|#########################*command*#################################### ####
#####################

\warn *text* [...]

##################psql####################\echo######

\watch [i[nterval]=*seconds*] [c[ount]=*times*] [m[in_rows]=*rows*] [*seconds*]

#####[78.758170]###
#\g############ #########(######2)######### #########*seconds*###interval=#
#################   ###########\pset   title######################
#####################   ##
############################\g#####   #################2#######   #######
##*seconds*#interval=#################### #######\pset title###########
########################

###

\x [*on* | *off* | *auto*]

######################### ##########\pset expanded##########

\z[S] [*pattern*]

pattern#######################
################### ############################## #######################
#S############

###\dp#######display privileges########

\! [*command*]

##############################psql######## ################*command*#######

############################\!#################################### #######
##################

\? [*topic*]

###########   ######*topic*###########commands##psql#################
commands#psql#################options#psql####################
##variables#psql##################

\;

########-###
###

###psql#####################################SQL################ ########

```
select 1; select 2; select 3;
```

###3##SQL### #######\;############
##########

```
select 1\; select 2\; select 3;
```

##########################3##SQL######################### ##############
##############BEGIN/COMMIT##
#################################53.2.2.1##########

####

##\d######*pattern*################################ ######################
#########SQL################### ###\dt FOO#foo################ SQL#
#################################### ######################################
################## ###SQL################### ####\dt "FOO""BAR"#FOO"BAR#
################foo"bar######## SQL############################## ####\dt
FOO"FOO"BAR#fooFOObar################

pattern##################\d################################## ###*########
########## (##
########### ##) ######################
#########*.*##############

#########*##0####################?####1######### ######Unix################
####\dt int*##int################### ##############*#?##################
#######

###(.)## ####\dt foo*.*bar*######
##foo################bar################ ######################################
######## ## ###(.)#2#########
################################
##################################

###(.)## ####\dn mydb.*foo*######
#foo################ ##
###########

#########(#############[0-9])################### ################9.7.3########
#########.###########*######.*#####?#.#####$################ .#####?##*R*####
#(*R*+|)##*R*?#####(*R*|)############################## ######################
########$#################### ########$################## ####################*#
############### ## #######
####################\do###################

#####

##

psql#####Unix####################### ################ ####################### ##
###(#########)##########################

##########psql#\set##################

```
testdb=> \set foo bar
```

########foo#bar############ ################################

```
testdb=> \echo :foo
bar
```

######SQL###################### ###SQL##############

#######\set######################## ###########(#######)#####\unset######## ##### ######################\set##########

> ##
>
> \set###################### #####\set :foo 'something'###########Perl# ###########PHP###################### ################################ ######## ###\set bar :foo##########################

##########psql########### ### ##psql#### ################## ##################ASCII###################### ############ ##################################

psql### \unset######################### ############### #2#####\set#####on################on######################## ###### ### on#off##############true#false##########################

#################

AUTOCOMMIT

> #######on################SQL########################## ###########BEGIN# ###SQL#START TRANSACTION################ ##off##############COMMIT### #END#######SQL############## ### ##############BEGIN#######################BEGIN######################### ################################VACUUM#############

> ##
>
> ##############ABORT#ROLLBACK################################## ### ######################################

> ##
>
> PostgreSQL##SQL########### # ######################psqlrc#############~/.psqlrc############## ##

COMP_KEYWORD_CASE

> SQL################################## lower###upper##################### ############# preserve-lower###preserve-upper############### ######## ##

DBNAME

> ################## ##
> ##

ECHO

> all################################ (#######################) ############ ########-a############## queries########psql######################### ####

##########-e### errors################################## ###############-b### none################################

ECHO_HIDDEN

#####on### ####
##PostgreSQL## ################
####-E############## #####noexec#######################################
####### #######off###

ENCODING

######################## ####################################\encoding##
################################

ERROR

###SQL###########true######false# SQLSTATE#########

FETCH_COUNT

#####0##################SELECT################################## ########
###################### ################################## ############100##
1000############## ##

###
################################aligned########## FETCH_COUNT##### ## ###############

HIDE_TABLEAM

#####true################################## ####################

HIDE_TOAST_COMPRESSION

#####true########################### ####################

HISTCONTROL

#####ignorespace############################## ignoredups############## ############### ignoreboth###########2############### none############## ##############################

##
#####Bash###########

HISTFILE

######################### #################PSQL_HISTORY######## ########
##########~/.psql_history####Windows##%APPDATA%\postgresql
\psql_history### #######

```
\set HISTFILE ~/.psql_history-:DBNAME
```

~/.psqlrc#######psql#####################

```
##

#####Bash############
```

HISTSIZE

#########################500#### ###################

```
##

#####Bash############
```

HOST

################## ##
##

IGNOREEOF

#####1#################EOF#####**Control**+**D**########psql####### 1##############
###########################EOF############### #############10######## #####
#0###

```
##

#####Bash############
```

LASTOID

INSERT#\lo_import#####################OID##### ########SQL############
############### #####12##PostgreSQL####OID##############################
#INSERT####LASTOID###0###

LAST_ERROR_MESSAGE
LAST_ERROR_SQLSTATE

###psql##################################SQLSTATE#######################
#######00000#

ON_ERROR_ROLLBACK

on###
interactive###
off###
#####################################SAVEPOINT##########################
################

ON_ERROR_STOP

########################## #####on###################### #######psql########
######## ######psql###########1#########################3###### #############

```
    ################################################## #################SQL#
    ##########################

PORT

    ################### ##########################################################
    ####

PROMPT1
PROMPT2
PROMPT3

    ########psql#################### #################

QUIET

    #####on###################-q########## #################

ROW_COUNT

    ###SQL#########################################################0#

SERVER_VERSION_NAME
SERVER_VERSION_NUM

    #####################9.6.2#10.1#11beta1####################
    #90602#100001##### ######################################################
    ####

SHELL_ERROR

    #################true#######false# ####\!#\g#\o#\w####\copy###########
    #################(`)######### \o######\o##################################
    ######## SHELL_EXIT_CODE#########

SHELL_EXIT_CODE

    ##########################     0–127##############128–255##############-1####
    ################################    ####\!#\g#\o#\w####\copy#############
    ###############(`)#########    \o###########\o############################
    SHELL_ERROR#########

SHOW_ALL_RESULTS

    #####off#####################(\;)##############################   ######on#
    ## off###########psql##########

SHOW_CONTEXT

    ######never#errors#####always##########CONTEXT#######################
    ########  ######errors###CONTEXT#######################################
    #####VERBOSITY#terse###sqlstate################### #\errverbose#######
    ##########################################

SINGLELINE

    #####on###################-S##########

SINGLESTEP

    #####on###################-s##########
```

SQLSTATE

###SQL####################A##########SQL#########00000#

USER

################ ##

VERBOSITY

#####default#verbose#terse#sqlstate############################
#\errverbose##

VERSION
VERSION_NAME
VERSION_NUM

#################psql################################
###9.6.2#10.1#11beta1######90602#100001#### ########################

SQL####

psql#########SQL#######################interpolate################### ###psql## SQL# ## ################################ ####(:)######### ##########

```
testdb=> \set foo 'my_table'
testdb=> SELECT * FROM :foo;
```

############my_table############## ######################## ################## ##################################### ## ##

##SQL## SQL#################### ############################## SQL## ## ################################## ##########################

```
testdb=> \set foo 'my_table'
testdb=> SELECT * FROM :"foo";
```

################SQL################## #####':foo'############################ ########## (##)

##############

```
testdb=> \set content `cat my_file.txt`
testdb=> INSERT INTO my_table VALUES (:'content');
```

(my_file.txt#NUL#################################### psql############NUL##### ########)

####:##SQL################## ################################ (:name#:'name'#:"name")######## ##

:{?*name*}#########################TRUE#FALSE################################ ###############

###########ECPG####################SQL########### ########################### PostgreSQL########################### SQL#################################### psql######

#####

psql########################## PROMPT1#PROMPT2#PROMPT3###3############### ######################## #####1#psql############################### #####2### ## #####3#SQL #COPY FROM STDIN################################

############################## #########%############## ###################### ####### #####################

%M

#################################Unix###########[local]###### ####Unix### #################################[local:*/dir/name*]######

%m

#################################Unix###########[local]######

%>

#####################

%n

################### ###########SET SESSION AUTHORIZATION############### ####################

%/

##############

%~

###############~################%/######

%#

#####################################>##### ###########SET SESSION AUTHORIZATION##################################

%p

##################ID###

%R

#####1#######=###################@###########^#########################!### ###\connect############## #####2####%R####psql############################ ##### #######################-####/* ... */################*############## ## #####(###### #####3####%R##############

%x

############## ##*############### ###########!##############################?######

%l

##############1#######

`%digits`

#####8#################

`%:name:`

psql###*name*##### #################

`%`*`command`*`

###################*command*######

`%[ ... %]`

######################### ##
Readline##############################%[#%]###########################
############################## #########

```
testdb=> \set PROMPT1 '%[%033[1;33;40m%]%n@%/%R%[%033[0m%]%# '
```

######VT100##################1;########33;40##############

`%w`

PROMPT1############### ######################2#################
#PROMPT2##########

#####################%%######### ##################1#2#'%/%R%x%# '######3#'>>
'###

##
#####tcsh############

#########

psql#Readline###libedit############################ #######psql##############psql###
############ ################control-P#######

#############(#######)######################SQL################### #############
#ins#####TAB######insert into ####### ########################TAB#########
(###########################TAB##
##############)

SQL### ######################
######### #####BEGIN#########################SET TRANSACTION ISOLATION
LEVEL############# ##################################.inputrc##############
#############

```
$if psql
set disable-completion on
$endif
```

####psql########Readline###### #######Readline#################

-n(--no-readline)##############Readline#####psql######################### ######
TAB##################################

##

COLUMNS

\pset columns######wrapped### ### ##############################

PGDATABASE
PGHOST
PGPORT
PGUSER

################32.15#####

PG_COLOR

###################### #####always#auto#never###

PSQL_EDITOR
EDITOR
VISUAL

\e#####\ef#####\ev################ ############################ ######### ###########Unix######vi##Windows######notepad.exe#######

PSQL_EDITOR_LINENUMBER_ARG

\e#\ef###\ev## ## Emacs###vi################(+)##### ## ###### #########

PSQL_EDITOR_LINENUMBER_ARG='+'
PSQL_EDITOR_LINENUMBER_ARG='--line '

Unix###########+### ##########vi################################## ## Windows#################

PSQL_HISTORY

################### ####~#########

PSQL_PAGER
PAGER

###########more####less### ###### ########PSQL_PAGER#PAGER#########\pset###################### ######## ############################ ################################more####### ##Cygwin##less#######

PSQL_WATCH_PAGER

\watch## ######Unix##### #PSQL_WATCH_PAGER###################### pspg####(PostgreSQL############ ################################)#######--stream#######\watch##########

PSQLRC

####.psqlrc############# ####~#########

SHELL

\!###############

`TMPDIR`

`###################` `######/tmp###`

`#################`PostgreSQL`##########`libpq`##################`32.15`##########`

####

`psqlrc` and `~/.psqlrc`

`-X#############`psql`##############################################(psqlrc)#` `###################(~/.psqlrc)###################` `#########\set#SET####` `########################################`

`#############psqlrc#######` `##############################################` `########pg_config --sysconfdir#############` `####`PostgreSQL`############` `##########../etc/######` `##############PGSYSCONFDIR##############`

`##############.psqlrc################################` Windows`########` `###########%APPDATA%\postgresql\psqlrc.conf##########` `##############` `##########PSQLRC##############`

`##################################~/.psqlrc-17#~/.psqlrc-17beta3#####` `#######`PostgreSQL`##############################################`psql`########` `########` `##########################################` `######################` `##############`

`.psql_history`

`#############~/.psql_history#`Windows`####%APPDATA%\postgresql` `\psql_history#######`

`##########`psql`###HISTFILE#######PSQL_HISTORY##################`

##

- psql`####################################` `###################`psql`############` `###########` `\d###############`9.2`##########################`psql`############` `###############` SQL`##################################################################` `##########################`

  `##########################psql###########psql################` `##############` `####psql##########################################` `##########################` `#####`

- PostgreSQL#9.6`######-c######-X(--no-psqlrc)#####################`

- PostgreSQL#8.4`######`psql#1`##############################################` `####` `################`

Windows########

psql`############################` Windows`######################################` `##psql#8#######################` psql`########################################` `##` `######################2#######`

- **`cmd.exe /c chcp 1252`**`################` #1252`##############################` `########` Cygwin`##################/etc/profile#########`

- `##########Lucida Console#########` `########`ANSI`####################`

# #

```
#########################################################

testdb=> CREATE TABLE my_table (
testdb(>  first integer not null default 0,
testdb(>  second text)
testdb-> ;
CREATE TABLE

#######################

testdb=> \d my_table
              Table "public.my_table"
 Column |  Type   | Collation | Nullable | Default
--------+---------+-----------+----------+---------
 first  | integer |           | not null | 0
 second | text    |           |          |

##########################

testdb=> \set PROMPT1 '%n@%m %~%R%# '
peter@localhost testdb=>

#########################################

peter@localhost testdb=> SELECT * FROM my_table;
 first | second
-------+--------
     1 | one
     2 | two
     3 | three
     4 | four
(4 rows)

\pset#################################

peter@localhost testdb=> \pset border 2
Border style is 2.
peter@localhost testdb=> SELECT * FROM my_table;
+-------+--------+
| first | second |
+-------+--------+
|     1 | one    |
|     2 | two    |
|     3 | three  |
|     4 | four   |
+-------+--------+
(4 rows)

peter@localhost testdb=> \pset border 0
Border style is 0.
peter@localhost testdb=> SELECT * FROM my_table;
first second
----- ------
    1 one
    2 two
    3 three
    4 four
(4 rows)
```

```
peter@localhost testdb=> \pset border 1
Border style is 1.
peter@localhost testdb=> \pset format csv
Output format is csv.
peter@localhost testdb=> \pset tuples_only
Tuples only is on.
peter@localhost testdb=> SELECT second, first FROM my_table;
one,1
two,2
three,3
four,4
peter@localhost testdb=> \pset format unaligned
Output format is unaligned.
peter@localhost testdb=> \pset fieldsep '\t'
Field separator is "    ".
peter@localhost testdb=> SELECT second, first FROM my_table;
one     1
two     2
three   3
four    4

###########################


peter@localhost testdb=> \a \t \x
Output format is aligned.
Tuples only is off.
Expanded display is on.
peter@localhost testdb=> SELECT * FROM my_table;
-[ RECORD 1 ]-
first  | 1
second | one
-[ RECORD 2 ]-
first  | 2
second | two
-[ RECORD 3 ]-
first  | 3
second | three
-[ RECORD 4 ]-
first  | 4
second | four

###############\g####1################


peter@localhost testdb=> SELECT * FROM my_table
peter@localhost testdb-> \g (format=aligned tuples_only=off expanded=on)
-[ RECORD 1 ]-
first  | 1
second | one
-[ RECORD 2 ]-
first  | 2
second | two
-[ RECORD 3 ]-
first  | 3
second | three
-[ RECORD 4 ]-
first  | 4
second | four

####\df###########int*pl#####2########bigint###############


testdb=> \df int*pl * bigint
```

```
                          List of functions
   Schema   |  Name   | Result data type | Argument data types | Type
------------+---------+------------------+---------------------+------
 pg_catalog | int28pl | bigint           | smallint, bigint    | func
 pg_catalog | int48pl | bigint           | integer, bigint     | func
 pg_catalog | int8pl  | bigint           | bigint, bigint      | func
(3 rows)

#####################\crosstabview##########################


testdb=> SELECT first, second, first > 2 AS gt2 FROM my_table;
 first | second | gt2
-------+--------+-----
     1 | one    | f
     2 | two    | f
     3 | three  | t
     4 | four   | t
(4 rows)

testdb=> \crosstabview first second
 first | one | two | three | four
-------+-----+-----+-------+------
     1 | f   |     |       |
     2 |     | f   |       |
     3 |     |     | t     |
     4 |     |     |       | t
(4 rows)

##2#########################################


testdb=> SELECT t1.first as "A", t2.first+100 AS "B", t1.first*(t2.first
+100) as "AxB",
testdb-> row_number() over(order by t2.first) AS ord
testdb-> FROM my_table t1 CROSS JOIN my_table t2 ORDER BY 1 DESC
testdb-> \crosstabview "A" "B" "AxB" ord
 A | 101 | 102 | 103 | 104
---+-----+-----+-----+-----
 4 | 404 | 408 | 412 | 416
 3 | 303 | 306 | 309 | 312
 2 | 202 | 204 | 206 | 208
 1 | 101 | 102 | 103 | 104
(4 rows)
```

reindexdb

reindexdb — PostgreSQL##################

##

reindexdb [*connection-option*...] [*option*...] [-S | --schema *schema*] ... [-t | --table *table*] ... [-i | --index *index*] ... [-s | --system] [*dbname* | -a | --all]

##

reindexdb##PostgreSQL############################

reindexdb##SQL####REINDEX######## ## ################

#####

reindexdb####################

-a
--all

#########################

--concurrently

CONCURRENTLY########### REINDEX##################################

[-d] *dbname*
[--dbname=]*dbname*

-a/--all## ################
#PGDATABASE############### ################################ *dbname*#####
######### ####################################

-e
--echo

reindexdb######################

-i *index*
--index=*index*

index########## ###-i##################################

-j *njobs*
--jobs=*njobs*

njobs##################reindex############## ################################
#############

reindexdb###########*njobs*##########max_connections##########################
###

##############--system###############

-q
--quiet

##############

-s
--system

##############################

-S *schema*
--schema=*schema*

schema################ ###-S##################################

-t *table*
--table=*table*

table################ ###-t##################################

--tablespace=*tablespace*

######################### (###########################)

-v
--verbose

###############

-V
--version

reindexdb################

-?
--help

reindexdb#########################

###reindexdb##############################

-h *host*
--host=*host*

######################### #################Unix#########################

-p *port*
--port=*port*

##########TCP#######Unix######################

-U *username*
--username=*username*

##############

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

##############reindexdb#################

##################reindexdb#################################### ####reindexdb
#################################### ################-W################

--maintenance-db=*dbname*

-a/--all###
#######postgres###################template1####### ############### ##
################################### ####################################
#########

##

PGDATABASE
PGHOST
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL###########libpq##################32.15###########

##

################################REINDEX#psql######### ######################
######### ###libpq##

#

######test##############

$ **reindexdb test**

abcd##############foo#############bar#######

$ **reindexdb --table=foo --index=bar abcd**

####

REINDEX

vacuumdb

vacuumdb — PostgreSQL##################

##

vacuumdb [*connection-option*...] [*option*...] [-t | --table *table* [(*column* [,...])]] ... [*dbname* | -a | --all]

vacuumdb [*connection-option*...] [*option*...] [-n | --schema *schema*] ... [*dbname* | -a | --all]

vacuumdb [*connection-option*...] [*option*...] [-N | --exclude-schema *schema*] ... [*dbname* | -a | --all]

##

vacuumdb##PostgreSQL############################### ###vacuumdb#PostgreSQL###### ##########################

vacuumdb##SQL####VACUUM######## ## #########

#####

vacuumdb######################

-a
--all

###########################

--buffer-usage-limit *size*

vacuumdb## ########################## ########################## VACUUM##########

[-d] *dbname*
[--dbname=]*dbname*

-a####--all## ################ ################PGDATABASE########## ################################## *dbname*##

--disable-page-skipping

################################

-e
--echo

vacuumdb########################

-f
--full

#####full###############

`-F`
`--freeze`

###############

`--force-index-cleanup`

##############################

`-j` *`njobs`*
`--jobs=`*`njobs`*

`njobs`#################vacuum###analyze#################################### ####################

vacuumdb###########*`njobs`*#########max_connections########################### ####

######`-f`#`FULL`## ##

`--min-mxid-age` *`mxid_age`*

###########ID########*`mxid_age`*################################### ######### ########ID##############################(24.1.5.1#########)#

################################ID##############################TOAST##### ############ vacuumdb##########################TOAST#################### #########

`--min-xid-age` *`xid_age`*

########ID########*`xid_age`*################################### ############# #ID##############################(24.1.5#########)#

#############################ID##############################TOAST######## ########## vacuumdb##########################TOAST######################## ######

`-n` *`schema`*
`--schema=`*`schema`*

`schema`############################ `-n`##############################

`-N` *`schema`*
`--exclude-schema=`*`schema`*

`schema`######################## `-N`############################

`--no-index-cleanup`

############################

`--no-process-main`

###############

`--no-process-toast`

########################TOAST############

`--no-truncate`

########################

-P *parallel_workers*
--parallel=*parallel_workers*

###CPU################### VACUUM#######

-q
--quiet

##############

--skip-locked

##############################

-t *table* [(*column* [,...])]
--table=*table* [(*column* [,...])]

table#########/###### ###--analyze#--analyze-only################### ###-t##################################

###
################################## ################

-v
--verbose

###############

-V
--version

vacuumdb################

-z
--analyze

#############################

-Z
--analyze-only

##

--analyze-in-stages

--analyze-only## ###3######### ### ######################################(default_statistics_target###)########## ################

#pg_upgrade###################### #####################################

-?
--help

vacuumdb#########################

vacuumdb#####################################

-h *host*
--host=*host*

######################### #################Unix#########################

-p *port*
--port=*port*

###########TCP#######Unix######################

-U *username*
--username=*username*

##############

-w
--no-password

############## ###################.pgpass##############################
##

-W
--password

#############vacuumdb#################

#################vacuumdb#################################### ####vacuumdb
#################################### ################-W################

--maintenance-db=*dbname*

-a####--all###
#################postgres####################template1####### #########
##############################
###############

##

PGDATABASE
PGHOST
PGPORT
PGUSER

###############

PG_COLOR

###################### #####always#auto#never###

####################PostgreSQL###########libpq##################32.15###########

##

##################################VACUUM#psql########## ######################
########## ###libpq##

# #

test##################################

```
$ vacuumdb test

bigdb#####################################################

$ vacuumdb --analyze bigdb

xyzzy############foo###1#########################bar###1#####################
####################

$ vacuumdb --analyze --verbose --table='foo(bar)' xyzzy

xyzzy############foo#####bar########################################

$ vacuumdb --schema='foo' --schema='bar' xyzzy
```

####

VACUUM

PostgreSQL###########

#####PostgreSQL## ######################### ################ #################PostgreSQL###################

##

initdb 1954
pg_archivecleanup 1959
pg_checksums 1961
pg_controldata 1963
pg_createsubscriber 1964
pg_ctl 1968
pg_resetwal 1973
pg_rewind 1977
pg_test_fsync 1981
pg_test_timing 1982
pg_upgrade 1985
pg_waldump 1994
pg_walsummary 1997
postgres 1998

initdb

initdb — PostgreSQL##################

##

initdb [*option*...] [--pgdata | -D] *directory*

##

initdb#PostgreSQL####################

##(############################)#
######postgres######template1######template0#####################
postgres###
template1#template0####CREATE DATABASE########################### ########
########template1#################### template0############template1####
#########22.3##########

initdb###root################
######################root#####chown####################################
#su################initdb#######

initdb#################################### initdb###########################
############# ####root##################root#initdb########## ##################
#####

####################initdb################################### --allow-group-
access## ###########################
###

initdb### ##############################
######## # initdb###

########initdb#########libc######(23.1.4#########)# libc######################
#########################

###################--locale########### ##############################--lc-
*#--icu-locale#####(####)# ##
##

###--locale-provider=icu########initdb#ICU##############################
####ICU######################## #######ICU####ID########--icu-locale########
########################initdb#ICU####################libc################
###########

initdb######################## ##
##########

#############23.1#########

###################--encoding############ ###23.3########

#####

-A *authmethod*
--auth=*authmethod*

#########pg_hba.conf#########################host###local######## ##### #######20.1##########

initdb##pg_hba.conf##########

################################trust########### ##############trust##### #########

--auth-host=*authmethod*

########pg_hba.conf##TCP/IP#######################host########

--auth-local=*authmethod*

########pg_hba.conf##Unix################################local########

-D *directory*
--pgdata=*directory*

#################################### initdb####################PGDATA### ######################### ##########################postgres################ #########

-E *encoding*
--encoding=*encoding*

######################### ## ######## PostgreSQL######################23.3.1##########

#################################### --no-locale##########(############C# ##POSIX###)#######ICU########UTF8#libc########SQL_ASCII######

-g
--allow-group-access

####################initdb############################## Windows##POSIX### ####################################

--icu-locale=*locale*

ICU##################ICU########### #########23.1##########

--icu-rules=*rules*

###ICU##############

-k
--data-checksums

I/O## ############## ############## ########################## ## ############pg_stat_database########### #######28.2##########

--locale=*locale*

############################ ################initdb#################################### # ############23.1#######

######--locale-provider#builtin####--locale###--builtin-locale### ##C###C.UTF-8############

--lc-collate=*locale*
--lc-ctype=*locale*
--lc-messages=*locale*
--lc-monetary=*locale*
--lc-numeric=*locale*
--lc-time=*locale*

--locale############################

--no-locale

--locale=C######

--builtin-locale=*locale*

#####[73.504274]#ICU#################ICU########## #########23.1########## ## ################################ ##############23.1##########

--locale-provider={builtin|libc|icu}

####################CREATE DATABASE############ ######libc##(23.1.4#########)#

--pwfile=*filename*

initdb#################################### ##########################

-T *config*
--text-search-config=*config*

################## #######default_text_search_config##########

-U *username*
--username=*username*

######################### #######initdb#########################

-W
--pwprompt

initdb## ######################### #### ##

-X *directory*
--waldir=*directory*

##############################

--wal-segsize=*size*

WAL###################### ###WAL################ ##########16######## ##1# #1024###2############ #####################################

WAL## ######WAL############## #######WAL############################ WAL#############WAL############ ##

##################################

-c *name=value*
--set *name=value*

initdb###########*name*#####*value*################################
#postgresql.conf############ #######################2######### ########
##################################

-d
--debug

##############
#initdb########################### ###########################

--discard-caches

debug_discard_caches=1############################ ####################
#############

-L *directory*

##################initdb################## ############ #################
###############

-n
--no-clean

########initdb###
##

-N
--no-sync

#######initdb################################ ###########initdb###########
#######
##

--no-instructions

#######initdb######################### ########################## ####
#initdb###

-s
--show

########################## ###initdb######################

--sync-method=*method*

############fsync#######initdb################################ ########
WAL##################################

######Linux###syncfs################WAL##################################
######################## recovery_init_sync_method##########

##############--no-sync##################

-S
--sync-only

################################## ######initdb############## ##########
##fsync#off##on###########################

##################

-V
--version

initdb################

-?
--help

initdb##########################

##

PGDATA

########################### -D######################

PG_COLOR

###################### #####always#auto#never###

TZ

############################### ##########################(8.5.3##)#

##

initdb#pg_ctl initdb###############

####

pg_ctl, postgres, 20.1

pg_archivecleanup

pg_archivecleanup — PostgreSQL WAL##############

##

pg_archivecleanup [*option*...] *archivelocation oldestkeptwalfile*

##

pg_archivecleanup####################WAL################
#archive_cleanup_command####################26.2#### ####pg_archivecleanup#
WAL##

pg_archivecleanup#######################postgresql.conf############

```
archive_cleanup_command = 'pg_archivecleanup archivelocation %r'
```

archivelocation##WAL#####################

archive_cleanup_command############%r#######WAL#####*archivelocation*#######
############—##
##*archivelocation*####################################*archivelocation*#####
WAL###

#######################*oldestkeptwalfile*#######WAL#######
#*archivelocation*######## #######.partial###.backup#####################
####*oldestkeptwalfile*######### ##.backup################################
########WAL################## ######000000010000003700000010############
##########

```
pg_archivecleanup -d archive 000000010000003700000010.00000020.backup

pg_archivecleanup:  keep WAL file "archive/000000010000003700000010" and
 later
pg_archivecleanup:  removing file "archive/00000001000000370000000F"
pg_archivecleanup:  removing file "archive/00000001000000370000000E"
```

pg_archivecleanup##*archivelocation*##

#####

pg_archivecleanup#####################

-b
--clean-backup-history

####################### ####################25.3.2##########

-d
--debug

stderr################

-n
--dry-run

#############stdout######(##########).

`-V`
`--version`

pg_archivecleanup################

`-x` *`extension`*
`--strip-extension=`*`extension`*

#################################### ####################################
############### #: `-x.gz`

`-?`
`--help`

pg_archivecleanup###########################

##

####`PG_COLOR`####################### #####`always#auto#never###`

##

pg_archivecleanup##PostgreSQL 8.0 ## # ##PostgreSQL 9.0 ##

pg_archivecleanup#C###

#

Linux#Unix##################

```
archive_cleanup_command = 'pg_archivecleanup -d /mnt/standby/archive %r
 2>>cleanup.log'
```

####################################`archive_command`#NFS################## ##################################### #######

- #######`cleanup.log`#######
- ################################

pg_checksums

pg_checksums — PostgreSQL####################################

##

pg_checksums [*option*...] [[-D | --pgdata]*datadir*]

##

pg_checksums#PostgreSQL############################## pg_checksums############# ############### ## ##

################################## ### ########### ################pg_control########

#####

#####################

-D *directory*
--pgdata=*directory*

##############################

-c
--check

############ #########################

-d
--disable

##############

-e
--enable

##############

-f *filenode*
--filenode=*filenode*

###########*filenode*######################

-N
--no-sync

pg_checksums################################## #########pg_checksums### ### #### ### --check################### #####

-P
--progress

########### #####################################

`--sync-method=`*`method`*

###########fsync######pg_checksums############################## #### ####WAL##################################

######Linux###syncfs###############WAL################################ ######################## recovery_init_sync_method##########

#############--no-sync#################

`-v`
`--verbose`

############ ####################

`-V`
`--version`

pg_checksums################

`-?`
`--help`

pg_checksums########################

##

`PGDATA`

############################ ######-D###############

`PG_COLOR`

###################### #######always#auto#never###

##

##################################

##############################pg_rewind## ## ########################## ## ############################## ####################

######################pg_checksums########kill################################## ##pg_checksums#############

pg_controldata

pg_controldata — PostgreSQL###################

##

pg_controldata [*option*] [[-D | --pgdata]*datadir*]

##

pg_controldata############initdb################# ###WAL################ ################## ##

##########PGDATA################ ##########pg_controldata##############-V###--version############# ###############-?###--help#############

##

PGDATA

###################

PG_COLOR

###################### #####always#auto#never###

pg_createsubscriber

pg_createsubscriber — convert a physical replica into a new logical replica

##

pg_createsubscriber [*option*...] { -d | --database }*dbname* { -D | --pgdata }*datadir* { -P | --publisher-server }*connstr*

##

######pg_createsubscriber################################ ## ## ## ####

###

`-t` *seconds*
`--recovery-timeout=`*seconds*

###########################0#############6###

`-U` *username*
`--subscriber-username=`*username*

##

`-v`
`--verbose`

##################pg_createsubscriber################################
#####################################

`--config-file=`*filename*

##pg_createsubscriber#####pg_ctl#####
#####################postgresql.conf#######################
#########################

`--publication=`*name*

##############################--publication#########################
###
##

`--replication-slot=`*name*

##################################--replication-slot################
##
###
############################

`--subscription=`*name*

##############################--subscription######################
###
###
########

`-V`
`--version`

######pg_createsubscriber################

`-?`
`--help`

######pg_createsubscriber##########################

##

Prerequisites

######pg_createsubscriber###
#########pg_createsubscriber#####################pg_replication_origin_advance()#####################

#########
#max_replication_slots#max_logical_replication_workers################################
##########max_worker_processes##################################
################

##
#####wal_level#`logical`########## ##
#max_replication_slots############ #########################max_wal_senders########
#####

Warnings

##################pg_createsubscriber## #####
##############################

######pg_createsubscriber############################## #########################
##

######DDL############################pg_createsubscriber####################DDL
############### ############################DDL##############################
########

##########pg_createsubscriber###
################################
#######################

###############primary_slot_name##

#########################pg_createsubscriber#######################################
#####

######pg_createsubscriber############################## #########################
########################### ############`two_phase`##########################
##############

######pg_createsubscriber#pg_resetwal################## #####################WAL##
##################### ##

How It Works

###

1. #################################### ##################pg_createsubscriber##
   ########

2. ################################## ########################## ##############
   ######pg_createsubscriber##########

3. ## #########FOR
   ALL TABLES########### publication-name############################
   ## #pg_createsubscriber_%u_%x######: ######*oid*,####*int*#replication-
   slot-name################################## #pg_createsubscriber_
   %u_%x######: ######*oid*,####*int*############ ############################
   ## ##############LSN##recovery_target_lsn##################################
   ############ ###########################

4. ### ##################
   ####LSN(recovery_target_lsn)###### ##############################`promote`##
   ##### ##########################WAL#####################################

    ## ######
    #######################/######################### --recovery-timeout######
    ####################################pg_createsubscriber#######

5. ## ######################
   ################### #pg_createsubscriber_%u_%x#######: ######*oid*,###
   #*int*######################## ##################### ##################### #
   ######### ############################## ################################
   ##########################

6. ## #############
   ##

7. ## ######################
   ########################### ######################################LSN#
   ## ################,##############################
   ##pg_replication_origin_advance()####################

8. ## ##########################
   ################

9. ###############primary_slot_name################################

10. ##

11. ############################ pg_resetwal#################### ########
    #pg_resetwal############

# #

    ###################hr#finance#######################

```
$ pg_createsubscriber -D /usr/local/pgsql/data -P "host=foo" -d hr -d
 finance
```

####

pg_basebackup

pg_ctl

pg_ctl — PostgreSQL#################

##

`pg_ctl` `init[db]` [`-D` *`datadir`*] [`-s`] [`-o` *`initdb-options`*]

`pg_ctl` `start` [`-D` *`datadir`*] [`-l` *`filename`*] [`-W`] [`-t` *`seconds`*] [`-s`] [`-o` *`options`*] [`-p` *`path`*] [`-c`]

`pg_ctl` `stop` [`-D` *`datadir`*] [`-m` `s[mart]` | `f[ast]` | `i[mmediate]`] [`-W`] [`-t` *`seconds`*] [`-s`]

`pg_ctl` `restart` [`-D` *`datadir`*] [`-m` `s[mart]` | `f[ast]` | `i[mmediate]`] [`-W`] [`-t` *`seconds`*] [`-s`] [`-o` *`options`*] [`-c`]

`pg_ctl` `reload` [`-D` *`datadir`*] [`-s`]

`pg_ctl` `status` [`-D` *`datadir`*]

`pg_ctl` `promote` [`-D` *`datadir`*] [`-W`] [`-t` *`seconds`*] [`-s`]

`pg_ctl` `logrotate` [`-D` *`datadir`*] [`-s`]

`pg_ctl` `kill` *`signal_name`* *`process_id`*

Microsoft Windows######

`pg_ctl` `register` [`-D` *`datadir`*] [`-N` *`servicename`*] [`-U` *`username`*] [`-P` *`password`*] [`-S` `a[uto]` | `d[emand]`] [`-e` *`source`*] [`-W`] [`-t` *`seconds`*] [`-s`] [`-o` *`options`*]

`pg_ctl` `unregister` [`-N` *`servicename`*]

##

pg_ctl#PostgreSQL##############PostgreSQL##########postgres#################### ######################## ####################pg_ctl############################### ############### ##############################

`init`###`initdb`####PostgreSQL## ###### `initdb` ########### ####initdb#########

`start`################# ########################/`dev/null`#Windows####`nul`# ####### Unix##############################pg_ctl###########(################)# pg_ctl#######################rotatelogs#################################### ##### ##`postgres`## Windows## ################################ ############`-l`################################ ##### `-l`############################

`stop`## `-m`########3################ #smart## ########################### ################################### #fast############################## ####### ## #immediate#################### ################ ####################################

`restart`#########stop#########start####### ####`postgres`######################## ################################### ### #######pg_ctl########`restart`#############

reload######postgres#######SIGHUP#######
#postgresql.conf#pg_hba.conf#### ### ###

-p *path*

postgres################# ######postgres########pg_ctl#############
##################
#postgres################################

init################initdb################

-s
--silent

###########################

-t *seconds*
--timeout=*seconds*

############################-w##### #########PGCTLTIMEOUT#############
###60####

-V
--version

pg_ctl################

-w
--wait

############# ###start#stop#restart#promote#register#################
###################

#########pg_ctl########PID############################ #############
###########PID##################### #######PID#####################
pg_ctl################################

###############-t###############pg_ctl################ ##################
##############################

-W
--no-wait

############## #####-w######

##################
##

PostgreSQL##########stop#################

-?
--help

pg_ctl######################

#######################################pg_ctl#########

Windows######

-e *source*

Windows##########################pg_ctl################# ######PostgreSQL#
###pg_ctl############################ ##########event_source##############
##PostgreSQL###
#########

-N *servicename*

################# ########################## ######PostgreSQL###

-P *password*

####################

-S *start-type*

############### *start-type*#auto#demand#######2############### ##########
####auto###########

-U *username*

################### ##########DOMAIN\username###########

##

PGCTLTIMEOUT

################################ #################60####

PGDATA

####################

#####pg_ctl############################PGDATA###########-D##########

##################postgres#########

####

postmaster.pid

pg_ctl##

postmaster.opts

####################pg_ctl##restart######-o############################
postgres###### ############status##########

#

######

################################

```
$ pg_ctl start
```

###5433###fsync############

```
$ pg_ctl -o "-F -p 5433" start
```

######

####################

```
$ pg_ctl stop
```

-m##################################

```
$ pg_ctl stop -m smart
```

#######

##pg_ctl################################
#################### ##############################

```
$ pg_ctl restart
```

####-o########################### ######5433##############fsync##########

```
$ pg_ctl -o "-F -p 5433" restart
```

########

###pg_ctl############

```
$ pg_ctl status
```

```
pg_ctl: server is running (PID: 13718)
/usr/local/pgsql/bin/postgres "-D" "/usr/local/pgsql/data" "-p" "5433" "-
B" "128"
```

2#######restart####################

####

initdb, postgres

pg_resetwal

pg_resetwal — PostgreSQL################################

##

pg_resetwal [-f | --force] [-n | --dry-run] [*option*...] [-D | --pgdata] *datadir*

##

pg_resetwal##########WAL##############pg_control#######################
##
########

######--wal-segsize################initdb##############################
################ ##

####################################pg_resetwal############################
#WAL## ############################
####### pg_resetwal##-f######
###############

#####[78.516624]################################ ##############################
############################## ######################initdb############## ###
############################## #########WAL################################
###
########initdb################ ###########################

pg_resetwal#pg_control##################-f#force,#######################
##########################OID###########ID#####
#############ID########WAL############################ ####################
############# ####################-f#####################################
########### ######################### ############################### #######
################

################################/#######################################

#####

datadir
-D *datadir*
--pgdata=*datadir*

   ########################### ######################################
   pg_resetwal#####PGDATA#######

-f
--force

   ########################pg_resetwal###### ############################
   ##pg_resetwal#pg_control############################

-n
--dry-run

   -n/--dry-run###########pg_resetwal#pg_control#######################
   ########### ######################pg_resetwal#########################

-V
--version

##################

-?
--help

##############

#####[76.600985]#########pg_resetwal#pg_control##########################
###################### ################0x##########16########### ##########
#####pg_resetwal#pg_control######################### ################# ###
##########16###0x################# ##########8kB#########################

-c *xid,xid*
--commit-timestamp-ids=*xid,xid*

#######################ID###########ID##########

#######################ID#######1##################pg_commit_ts#######
######################### #########################ID#######2###########
############################## ######16#########

-e *xid_epoch*
--epoch=*xid_epoch*

##########ID#################

pg_resetwal######################ID############################## #####
################################## Slony-I#Skytools##########################
######################### ####################################

-l *walfile*
--next-wal-file=*walfile*

##WAL################WAL#############

##WAL#######################pg_wal########WAL#######################
#####16####3############ #########ID#################### ####pg_wal#
########0000000100000320000004A#######-l
000000010000003200000 04B###########

#######WAL##################WAL##########################LSN######
############ #######LSN####WAL#############

##
pg_resetwal###pg_wal################################-l######## ## ###########-l##################pg_wal#####WAL############## ##########pg_wal#########################

-m *mxid,mxid*
--multixact-ids=*mxid,mxid*

#############ID##############ID##########

#############ID#######1##################pg_multixact/offsets#######
################1#####65536(0x10000)############# ################ID######

##-m#2##################################65536############# ######16#######
###################16########4#########

-o *oid*
--next-oid=*oid*

##OID##########

#######OID###########OID######################### ##########OID########
###################

-O *mxoff*
--multixact-offset=*mxoff*

############################

#################pg_multixact/members#########################1#####
##52352(0xCC80)############# ######16##### ##########0####################

--wal-segsize=*wal_segment_size*

###WAL#################### ###1##1024#######2################ ####initdb###
##############

############################WAL######################## ######initdb#####
###########

##
pg_resetwal#########WAL############WAL######################### ###WAL######################## ###########WAL#################### ############WAL###########-l###########

-u *xid*
--oldest-transaction-id=*xid*

###################ID##########

#################pg_xact##########################1048576(0x100000)#######
####### ######16################ ###########16################# ##
##0007#pg_xact##############-u 0x700000##############5################
#######

-x *xid*
--next-transaction-id=*xid*

#########ID##########

#################pg_xact##########################1#####
##1048576(0x100000)############# ######16############### ###########16######
########### ####0011#pg_xact##############-x 0x1200000##############5#
######################

##

PG_COLOR

###################### #####always#auto#never###

##

```
########################## pg_resetwal################################### ###
################################## ##################pg_resetwal############
#################################################

pg_resetwal###########################
```

####

pg_controldata

pg_rewind

pg_rewind — PostgreSQL##################################

##

pg_rewind [*option*...] { -D | --target-pgdata } *directory* { --source-pgdata=*directory* | --source-server=*connstr* }

##

pg_rewind##PostgreSQL## ########### ##

####(rewind)## ######### #########rsync################pg_rewind## ##WAL## ######################## ## ##############

pg_rewind## ############## ###################pg_wal###############WAL########## ## ################################## ######################## ##################################WAL############# ###### #####WAL#########pg_wal############## #####-c########pg_rewind### ##WAL####################### pg_rewind################################## ##

pg_rewind#################################WAL############ ##WAL########### pg_rewind#######WAL### ##############pg_rewind##################################WAL############### ###### recovery.signal#####################postgresql.conf### #restore_command##################

pg_rewind#################postgresql.conf#wal_log_hints###############initdb### ##################################### ###########on############## full_page_writes#on############################

> **##: ########**
>
> ####pg_rewind## ################ #############
>
> pg_rewind## ################################### ################################## ######################
>
> pg_rewind########################### ###########################SSL###### ######################### ##################################pg_rewind## ############ ### ################ ##################################

#####

pg_rewind#######################

-D *directory*
--target-pgdata=*directory*

##################################### ##########pg_rewind#######################
########

--source-pgdata=*directory*

##################################
############

--source-server=*connstr*

################PostgreSQL########libpq############ #####################
pg_rewind## ####
####################################

-R
--write-recovery-conf

#########standby.signal#####postgresql.auto.conf############ ########
#--source-server######

-n
--dry-run

##################################

-N
--no-sync

########pg_rewind############################## ###########pg_rewind######
######
##

-P
--progress

###

-c
--restore-target-wal

WAL#####pg_wal#####################################restore_command####
WAL#####WAL#############

--config-file=*filename*

################################## ####pg_rewind####################postgres##
##############(-c/--restore-target-wal#########restore_command#####
########################)#######

--debug

######pg_rewind###########################

--no-ensure-shutdown

pg_rewind## ##########################
#########pg_rewind## ############
##pg_rewind## ###################
#######

--sync-method=*method*

###########fsync######pg_rewind############################## #######
#WAL##################################

######Linux###syncfs###############WAL################################
######################## recovery_init_sync_method##########

#############--no-sync##################

-V
--version

#################

-?
--help

#############

##

--source-server############pg_rewind# libpq###############(32.15###)#

####PG_COLOR######################## #####always#auto#never###

##

####################pg_rewind############################pg_rewind############
#################### rewind_user#########################

```
CREATE USER rewind_user LOGIN;
GRANT EXECUTE ON function pg_catalog.pg_ls_dir(text, boolean, boolean) TO
 rewind_user;
GRANT EXECUTE ON function pg_catalog.pg_stat_file(text, boolean) TO
 rewind_user;
GRANT EXECUTE ON function pg_catalog.pg_read_binary_file(text) TO
 rewind_user;
GRANT EXECUTE ON function pg_catalog.pg_read_binary_file(text, bigint,
 bigint, boolean) TO rewind_user;
```

#########

###

1. ##WAL#######
   #### ###WAL########################### ################################
   ######################## WAL##############################WAL##########-
   c########pg_rewind#########

2. ################--source-pgdata##SQL #--source-server###############
   ################################ ########################WAL#############
   ##

3. #############WAL######pg_xact##
   ######### ##################pg_dynshmem/#pg_notify/#pg_replslot/
   #pg_serial/#pg_snapshots/#pg_stat_tmp/#####pg_subtrans/############
   ################## pgsql_tmp######################## ###
   #backup_label#tablespace_map#pg_internal.init#postmaster.opts#postmaster.pid#
   ##.DS_Store######

4. ######################WAL###########backup_label#####################
   ####pg_current_wal_insert_lsn()##################LSN###############
   #############LSN#pg_control##########

5. ###########PostgreSQL####WAL##

pg_test_fsync

pg_test_fsync — PostgreSQL######`wal_sync_method`#####

##

`pg_test_fsync` [*`option`*...]

##

pg_test_fsync################wal_sync_method############################## ####### #I/O################## ###pg_test_fsync## ## ## pg_test_fsync##`wal_sync_method`#### ######################### ###commit_delay#########################

#####

pg_test_fsync#######################

`-f`
`--filename`

> ########################## #######`pg_wal`################################## ############### #`pg_wal`##WAL########### ############### #`pg_test_fsync.out`###

`-s`
`--secs-per-test`

> ################ ## ######### ########2############5####

`-V`
`--version`

> pg_test_fsync################

`-?`
`--help`

> pg_test_fsync#######################

##

`####PG_COLOR#######################` `#####always#auto#never###`

####

postgres

pg_test_timing

pg_test_timing — ################

##

pg_test_timing [*option*...]

##

pg_test_timing## ###########
###########EXPLAIN ANALYZE#################

#####

pg_test_timing#######################

-d *duration*
--duration=*duration*

################ ## ###
########3####

-V
--version

pg_test_timing################

-?
--help

pg_test_timing############################

####

#####

##################(90%##)#1############ ################(per loop time)########
##100########### #####Intel i7-860#####TSC#######################

```
Testing timing overhead for 3 seconds.
Per loop time including overhead: 35.96 ns
Histogram of timing durations:
  < us   % of total      count
     1     96.40465   80435604
     2      3.59518    2999652
     4      0.00015        126
     8      0.00002         13
    16      0.00000          2
```

#########(per loop time)#######(histogram)#################### ########(ns)###########
##################1#####(us)##############

#######################

############EXPLAIN ANALYZE## #
#psql####################################

```
CREATE TABLE t AS SELECT * FROM generate_series(1,100000);
\timing
SELECT COUNT(*) FROM t;
EXPLAIN ANALYZE SELECT COUNT(*) FROM t;
```

######i7-860######count#######9.8###########EXPLAIN ANALYZE#######16.6#######
################10########### ##6.8#########################68#################
pg_test_timing#######2#### ###################count##############70%######### ###
##

#########

######Linux#################################### 2#################acpi_pm######
##

```
# cat /sys/devices/system/clocksource/clocksource0/available_clocksource
tsc hpet acpi_pm
# echo acpi_pm > /sys/devices/system/clocksource/clocksource0/
current_clocksource
# pg_test_timing
Per loop time including overhead: 722.92 ns
Histogram of timing durations:
  < us   % of total      count
     1     27.84870    1155682
     2     72.05956    2990371
     4      0.07810       3241
     8      0.01357        563
    16      0.00007          3
```

#########EXPLAIN ANALYZE######115.9######### #########1061##################
#################################### ####################################
########################EXPLAIN
ANALYZE##################################

FreeBSD###

```
# dmesg | grep "Timecounter"
Timecounter "ACPI-fast" frequency 3579545 Hz quality 900
Timecounter "i8254" frequency 1193182 Hz quality 0
Timecounters tick every 10.000 msec
Timecounter "TSC" frequency 2531787134 Hz quality 800
# sysctl kern.timecounter.hardware=TSC
kern.timecounter.hardware: ACPI-fast -> TSC
```

################################## ##Linux######"clock"##########################
######Linux##################"jiffies"############## jiffies#Linux##################
##

```
$ cat /sys/devices/system/clocksource/clocksource0/available_clocksource
jiffies
$ dmesg | grep time.c
time.c: Using 3.579545 MHz WALL PM GTOD PIT/TSC timer.
time.c: Detected 2400.153 MHz processor.
$ pg_test_timing
Testing timing overhead for 3 seconds.
Per timing duration including loop overhead: 97.75 ns
Histogram of timing durations:
  < us   % of total      count
     1     90.23734   27694571
```

```
     2      9.75277    2993204
     4      0.00981       3010
     8      0.00007         22
    16      0.00000          1
    32      0.00000          1
```

##################

######################
################################## ##
###################
##############################

################################### ################################## #######
############################
################# ##
###########

###########(TSC)#######CPU#################### ############TSC##############
################TSC############ TSC################################ #######
##TSC####CPU########################### #########CPU#TSC##################
############# ##1#### ##############
####################TSC##################

####################TSC## ###
##TSC## ###########
##TSC################TSC#############
##

###########(HPET)###########TSC######################### ############100#######
##

ACPI(Advanced Configuration and Power Interface)#################Linux####acpi_pm#####
acpi_pm###########300##############

##PC###################8254################(PIT)###########(RTC)#APIC(Advanced Programmable Interrupt Controller)#########(Cyclone)####### ######################
##

####

EXPLAIN

pg_upgrade

pg_upgrade — PostgreSQL##################

##

pg_upgrade -b *oldbindir* [-B *newbindir*] -d *oldconfigdir* -D *newconfigdir* [*option*...]

##

pg_upgrade(#####pg_migrator#########)##############################12.14##13.10 ##14.9##15.5#######################/###########PostgreSQL#####################
#PostgreSQL################ #######12.7##12.8#14.1##14.5#########################
###

PostgreSQL## ####################
pg_upgrade###
##pg_upgrade###########
############ #########################

pg_upgrade##32########64###############
######################### ################################pg_upgrade###########
###

pg_upgrade#9.2.X########PostgreSQL#########(##########β#######)##################

#####

pg_upgrade####################

-b *bindir*
--old-bindir=*bindir*

##PostgreSQL###############PGBINOLD####

-B *bindir*
--new-bindir=*bindir*

###PostgreSQL#####################pg_upgrade##########PGBINNEW####

-c
--check

#########################

-d *configdir*
--old-datadir=*configdir*

##################PGDATAOLD####

-D *configdir*
--new-datadir=*configdir*

###################PGDATANEW####

-j *njobs*
--jobs=*njobs*

##################

-k
--link

##################################

-N
--no-sync

########pg_upgrade## #######
#pg_upgrade###
################### ##

-o *options*
--old-options *options*

##postgres################################

-O *options*
--new-options *options*

###postgres################################

-p *port*
--old-port=*port*

############PGPORTOLD####

-P *port*
--new-port=*port*

#############PGPORTNEW####

-r
--retain

#############SQL#################

-s *dir*
--socketdir=*dir*

#########postmaster###PGSOCKETDIR

-U *username*
--username=*username*

#################PGUSER####

-v
--verbose

##############

-V
--version

#################

`--clone`

#################################(############reflink############)###### ######
#############################-k/--link##################

##################
##pg_upgrade############ #######Btrfs#XFS(#reflink##################)#Linux(###
#4.5##)##APFS#macOS###########

`--copy`

################### ########## #--link###--clone##########

`--copy-file-range`

#########################copy_file_range###### #####################
######--clone### #
##Linux#FreeBSD###########

`--sync-method=`*`method`*

###########fsync######pg_upgrade#####################################
######### ########WAL##################################

######Linux###syncfs##############################WAL###################
#################################### recovery_init_sync_method##########

##############--no-sync##################

`-?`
`--help`

###############

####

pg_upgrade#####################

1. ###############

   #########################(###/opt/PostgreSQL/17)########################
   ######## ##

   ################################/usr/local/pgsql##########PostgreSQL####
   ##################PostgreSQL############################ #####PostgreSQL###
   ##########PostgreSQL###################### #########/usr/local/pgsql###
   #####################

   ```
   mv /usr/local/pgsql /usr/local/pgsql.old
   ```

2. ##########################

   ################configure###########PostgreSQL############ pg_upgrade#####
   ################################pg_controldata#######

3. **###PostgreSQL###########**

   ################################ pg_upgrade#######################

   ##prefix##########

   ```
   make prefix=/usr/local/pgsql.new install
   ```

4. **###PostgreSQL#######**

   initdb#################### ####################initdb############## ################################## ####################

5. ####################

   contrib##pgcrypto.so######## ##########(###DLL)###### ## ## ###CREATE EXTENSION pgcrypto## ############## ##pg_upgrade##################################

6. ##############

   ###########(##################)#########################

7. #####

   pg_upgrade#################### #####pg_hba.conf##peer######## ##~/.pgpass####(32.16##)####################

8. **Prepare for publisher upgrades**

   ######pg_upgrade############### ### ### ####################17.0################ #####17.0################### #####

   ##############################ALTER SUBSCRIPTION ... DISABLE######## ############################## ##############################

   ######pg_upgrade## ################### ######

   - #############wal_level#logical##########
   - ####################################max_replication_slots########## ##
   - ##################################PostgreSQL#########################
   - ##
   - ################################## ###pg_replication_slots#conflicting#### ## true######
   - ################################## ####pg_replication_slots########### temporary#false###

9. **Prepare for subscriber upgrades**

   ################################## pg_upgrade##pg_subscription_rel############# ## ############# ## ############################## 17.0################ ##### 17.0 ################################

   ######pg_upgrade## ############### #########

   - ##################################i########r################### ## #pg_subscription_rel############ srsubstate

- ## ## ##pg_subscription#pg_replication_origin##########################
- ##max_replication_slots####### ######

10. #######

    ######################### ###Unix###########

    ```
    pg_ctl -D /opt/PostgreSQL/12 stop
    pg_ctl -D /opt/PostgreSQL/17 stop
    ```

    Windows####

    ```
    NET STOP postgresql-12
    NET STOP postgresql-17
    ```

    ##

11. ###################

    ############# 13#####################pg_controldata########################### ############################## #########Latest checkpoint location################ ################## ################postgresql.conf#### #wal_level#minimal##################

12. **pg_upgrade###**

    ######################pg_upgrade############## pg_upgrade#################### ###################bin########### ### #########################

    ##############################(##############)############################### ## ################################### ################### #########pg_wal######################## ############# ### ###################### ##################### ##

    --jobs###########CPU## ## ####################CPU################## ############################## ##############################

    #####[56.953642]################################pg_upgrade########## #### ##Windows ##pg_upgrade########### ##

    ```
    pg_upgrade.exe
            --old-datadir "C:/Program Files/PostgreSQL/12/data"
            --new-datadir "C:/Program Files/PostgreSQL/17/data"
            --old-bindir "C:/Program Files/PostgreSQL/12/bin"
            --new-bindir "C:/Program Files/PostgreSQL/17/bin"
    ```

    ####pg_upgrade#2##################################### #######pg_upgrade --check############ ######################## ##pg_upgrade --check##### ##################################### ##################################### ##--check####--link###--clone############## pg_upgrade################ ###########

#################################### ##########################pg_upgrade #50432############# ### ### ##

##########################pg_upgrade############### 19################ ####### ##pg_upgrade#######pg_upgrade################################## ##### ###contrib## #contrib###

13. ##

    ################################26.2.5#############26.2######################## ######################### #########pg_upgrade################rsync########### ## ###################

    #################rsync## ##pg_upgrade##

    1. **PostgreSQL#######################**

       ##

    2. ############################

       ## initdb######################## ##################

    3. ####################

       ##

    4. ###########

       ##################################

    5. #########

       ####################################postgresql.conf(################ ##)#postgresql.auto.conf#pg_hba.conf######### ################### ########

    6. **rsync###**

       ##############rsync################################## ################### ## ################

       ```
       rsync --archive --delete --hard-links --size-only --no-inc-
       recursive old_cluster new_cluster remote_dir
       ```

       ###old_cluster###new_cluster#########################remote_dir### ## ############################## ################## ######################rsync################ #######

       ```
       rsync --archive --delete --hard-links --size-only --no-inc-
       recursive /opt/PostgreSQL/12 \
             /opt/PostgreSQL/17 standby.example.com:/opt/PostgreSQL
       ```

rsync#`--dry-run`################################## rsync######1########
##rsync######
#######################

#########pg_upgrade###
############# ### #
##############
######### #####rsync## ####
##########################

#######################################rsync################ #######

```
rsync --archive --delete --hard-links --size-only --no-inc-
recursive /vol1/pg_tblsp/PG_12_201909212 \
      /vol1/pg_tblsp/PG_17_202307071 standby.example.com:/vol1/
pg_tblsp
```

`pg_wal`####################################rsync#############

7. ####################################

   ###################### ######################
   ##`pg_backup_start`(##`pg_backup_stop()`############################
   #######)# ############ 17.0 ##
   ######################## ############ 17.0 ##############################
   ##

14. **`pg_hba.conf`**###

    `pg_hba.conf`######################### #########################
    #`postgresql.conf`(################)#`postgresql.auto.conf`############
    #############

15. ########

    ################### ####rsync#####################

16. ######

    ##################pg_upgrade############## ################################
    ### ## ########################
    ##

    ```
    psql --username=postgres --file=script.sql postgres
    ```

    ##

    > ##
    >
    > ## #
    > ################################ ####################################
    > ###

17. ####

    #############pg_upgrade###
    #####################################

```
#####vacuumdb       --all       --analyze-only#################### 
#####  --jobs#################### --analyze-in-stages############
################  vacuum_cost_delay#0##############PGOPTIONS#######
#PGOPTIONS='-c vacuum_cost_delay=0' vacuumdb ...#################
#########
```

18. ```
    ########
    ```

    ```
    ####################pg_upgrade########################################
    ##### ############################################ #bin#share##########
    ################
    ```

19. ```
    #########
    ```

    ```
    pg_upgrade################################
    ```

    - ```
      --check################################
      ```
    - ```
      --link####################################
      ```
    - ```
      --link##############################################
      ```
      - ```
        ###########pg_upgrade##################################
        ```
      - ```
        ############################$PGDATA/global/pg_control#.old######
        #################### ################$PGDATA/global/pg_control#
        #.old############################
        ```
      - ```
        ################################################### ################
        ################
        ```

##

```
############################################
PGBINOLD
    ########PostgreSQL############### #####-b/--old-bindir#
PGBINNEW
    #########PostgreSQL########### #####-B/--new-bindir#
PGDATAOLD
    ########################### #####-d/--old-datadir#
PGDATANEW
    ########################### #####-D/--new-datadir#
PGPORTOLD
    ################# #####-p/--old-port#
PGPORTNEW
    ################## #####-P/--new-port#
PGSOCKETDIR
    ###############postmaster############### #####-s/--socketdir#
```

PGUSER

######################## #####-U/--username#

##

pg_upgrade#######################################pg_upgrade_output.d#######
#######ISO 8601####################(%Y%m%dT%H%M%S)###########################
############################## pg_upgrade##########pg_upgrade_output.d######
################# ##

pg_upgrade################postmaster####### ##postmaster########Unix##############
##################### ## #####-s##
####################################
########## (###Windows##############)

#####################################pg_upgrade########## ######################
###################### ################################## ####################
############################
##############

###

pg_upgrade###OID#####reg*###

regcollation
regconfig
regdictionary
regnamespace
regoper
regoperator
regproc
regprocedure

(regclass#regrole#regtype############)

###############
##################################### ############################rsync###
##rsync   --checksum###############
################# #rsync############1#######--checksum####### ###25.3.4####
#postmaster.pid########################### ################################
###
##############################

####

initdb, pg_ctl, pg_dump, postgres

pg_waldump

pg_waldump — PostgreSQL################################

##

`pg_waldump` [`option`...] [`startseg` [`endseg`]]

##

`pg_waldump`#########WAL####### ################

###

#####

#############################

startseg

####WAL#################### ####################################

endseg

####WAL####################

-b
--bkp-details

#######################

-B *block*
--block=*block*

########################### ########--relation###-R##############

-e *end*
--end=*end*

##########################WAL##############

-f
--follow

###WAL#############WAL######1###1###########

-F *fork*
--fork=*fork*

#####[79.389313]## ##############main##
#######fsm########vm#init#####init### ####################################
################main###########fsm##########vm##########init###

-n *limit*
--limit=*limit*

#################

-p *path*
--path=*path*

WAL##########################WAL#############pg_wal#################### ###### ##############################pg_wal########PGDATA#pg_wal######## ########

-q
--quiet

################## ########WAL## ##

-r *rmgr*
--rmgr=*rmgr*

##################################### ##################################### list#######################################

#####[78.571429]##############################pg_waldump#################### ######################## ####################custom############# ###"###"# 3###########ID### ################### ################################### #####pg_waldump################# ################################## ##### ##custom################################## ########3###########ID### ###### #############

-R *tblspc/db/rel*
--relation=*tblspc/db/rel*

################################## ########1234/12345/12345############## #######OID#######OID###relfilenode######## ##############################

-s *start*
--start=*start*

########WAL##### ##############################WAL###############

-t *timeline*
--timeline=*timeline*

WAL################### ########*startseg*###########*startseg*########### ## ############1### ##10#####16#############17#0x11###

-V
--version

pg_waldump################

-w
--fullpage

##########################

-x *xid*
--xid=*xid*

#############ID###################

-z
--stats[=record]

######rmgr##################

pg_waldump#####SIGINT(**Control**+**C**)################################ ##### Windows##############

`--save-fullpage=`*`save_path`*

WAL####################*`save_path`*############ ############################ ############

############################ *`TIMELINE-LSN.RELTABLESPACE.DATOID.RELNODE.BLKNOFORK`* ##################

####	##
TIMELINE	8###16##1#%08X##############WAL### ############
LSN	8###16##2#%08X-%08X############### ##LSN
RELTABLESPACE	###########OID
DATOID	###########OID
RELNODE	############
BLKNO	###########
FORK	_main#_fsm#_vm#_init############# ############

`-?`
`--help`

pg_waldump##########################

##

`PGDATA`

##########-p###############

`PG_COLOR`

###################### #####always#auto#never###

##

#############################

#################################### ####################

pg_waldump####.partial#WAL############### ####################.partial#### ######

####

28.6

pg_walsummary

pg_walsummary — print contents of WAL summary files

##

`pg_walsummary` [*option*...] [*file*...]

##

######pg_walsummary##WAL###################### ###################### #`pg_wal/summaries`################################ ############## WAL####### ##

######WAL##############OID#######OID############################## ###fork### ##WAL##################################### ####WAL######################### ####0## ####WAL####fork############## ###############limit################################

#####

`-i`
`--individual`

##############`pg_walsummary`######1##################1########### ######0## 999#########################1000##########1################################ ######################################

`-q`
`--quiet`

#################### ####WAL##

`-V`
`--version`

#######################

`-?`
`--help`

######pg_walsummary###########################

##

##########`PG_COLOR`###################### #####`always#auto#never`###

####

pg_basebackup, pg_combinebackup

postgres

postgres — PostgreSQL#########

##

postgres [*option*...]

##

postgres##PostgreSQL############ ################################postgres# ###################################### ####postgres#########################

1##postgres########1#################### ##############################
######################### 1##############postgres#########################
################################ postgres########################### #####-
D###########PGDATA############################ ###-D###PGDATA##initdb######
################### ####################19.2#########

#######postgres################################## ######postgres##########
##########################

##postgres######################## #########initdb################## ###########
################### ##
################### ##################################### #################
######################## ######################ID#1###########################
######### ######################## ##
##########

#####

postgres###################### #############19########## ####################
################################## ##################################### ######
################################## #########PGOPTIONS#######libpq##############
########## ########postgres####################

######

-B *nbuffers*

########################## ###############initdb############# ################
######shared_buffers#############

-c *name=value*

################## PostgreSQL################19######## ####################
########################## -c##############################

-C *name*

############################## #########-c########## ###postgresql.conf##
################################## ############################

################################## ####################(##
#shared_memory_size#shared_memory_size_in_huge_pages#wal_segment_size##)########
##################

#########pg_ctl## ###########
#########SHOW#pg_settings#############

-d *debug-level*

############# ## #############1##5###
#########-d 0########### ##########postgres#############################

-D *datadir*

################################ ###19.2##########

-e

##########################DMY######## ######################################
###8.5##########

-F

########fsync########### ################################### ##############
####fsync############# ################################

-h *hostname*

postgres################################IP########### ####################
##################################*############ ##########IP################
####### ###############Unix################## #######localhost######### ########
##############listen_addresses#############

-i

#############TCP/IP######################### ##############################
######### #########postgresql.conf##listen_addresses######-h#*########
##########

########listen_addresses################################ #####
#listen_addresses##############

-k *directory*

postgres########################Unix###################### ###############
################ ##########Unix####################################TCP/IP###
############ ##########/tmp######################## ####################
#unix_socket_directories############

-l

SSL################ #################PostgreSQL########SSL############## SSL
##########18.9##########

-N *max-connections*

################################ ###############initdb############## ###########
###########max_connections#################

-p *port*

####################postgres#####TCP/IP###########Unix####################
###########PGPORT#PGPORT###########################5432#########
################################PGPORT##################################
###########

-s

##

-S *work-mem*

#######19.4.1#work_mem###

-V
--version

postgres################

--*name=value*

################## -c#######

--describe-config

###COPY########### ################ ###########

-?
--help

postgres##########################

##########

##################PostgreSQL################ ################

-f { s | i | o | b | t | n | m | h }

##################### s############i################## o#b#t############ ########################TID########### n############m#######h############ ##

-fs######-fn####################

-O

#################### ###initdb#########

-P

################################ ############################## ######### #################

-t pa[rser] | pl[anner] | e[xecutor]

########-s##################

-T

#################################### ##########################SIGQUIT#### ############################## #############SIGABRT######################## ##

-v *protocol*

###################/########################## ########################

-W *seconds*

##

##################

##

--single

################ #########################

database

#################### ########################## ####################

-E

############################

-j

##################################2###########

-r *filename*

###########*filename*###### ######################################

##

PGCLIENTENCODING

########################### ######################## #####################

PGDATA

####################

PGDATESTYLE

########DateStyle######### ##################

PGPORT

############## ########################

##

semget#shmget## ### ###18.4######### ####shared_buffers########PostgreSQL############ ##max_connections###

##

```
$ ps ax | grep postgres
```

####

```
$ ps -ef | grep postgres
```

######################### ##
####

#################################PostgreSQL################################ #
##postgres## ######################
#################### ### ##
##Unix###########1024#################trusted###########Unix####################

###########pg_ctl#####postgres#############################

postgres########SIGKILL################### SIGKILL#######postgres#######
###################################### ####################postgres########
#############

postgres###############SIGTERM#SIGINT#SIGQUIT############ SIGTERM#######
##################### SIGINT######################### SIGQUIT###############
##########################

SIGHUP########################## ###SIGHUP#################################
##

##################################SIGINT############# ###################
###########SIGTERM######### ################SQL##################9.28.2#
#pg_cancel_backend###pg_terminate_backend#########

postgres##SIGQUIT####### #############
#######SIGKILL#################### ####postgres#####################
##################################postgres#############

--######FreeBSD#OpenBSD########## ####-c######### ######################## ##
###########PostgreSQL#################

###########

##

```
postgres --single -D /usr/local/pgsql/data other-options my_database
```

###################-D#######################PGDATA############ #############
###############

#################################### ##########psql################### #######
##############################
####################################

###############-j########1###
####### ################################## ################################
##

#############
################1###################

#############EOF###**Control**+**D**######## ##################################EOF###
#######################EOF#############

###############
##

# #

###########postgres###########################

```
$ nohup postgres >logfile 2>&1 </dev/null &
```

###1234###########postgres###################

```
$ postgres -p 1234
```

psql#######################-p####################

```
$ psql -p 1234
```

#######PGPORT##########

```
$ export PGPORT=1234
$ psql
```

################################

```
$ postgres -c work_mem=1234
$ postgres --work-mem=1234
```

#########postgresql.conf#########work_mem########## #######################
#_###################-#################### ##################################
###postgresql.conf#########################

####

initdb, pg_ctl

VII.

#####PostgreSQL#####################

##

50. PostgreSQL##### 2010
50.1. ######## 2010
50.2. ##### 2010
50.3. ###### 2010
50.3.1. ### 2011
50.3.2. ######## 2011
50.4. PostgreSQL####### 2011
50.5. ####/####### 2012
50.5.1. ########## 2012
50.6. ####### 2013
51. ######## 2014
51.1. ## 2014
51.2. pg_aggregate 2015
51.3. pg_am 2017
51.4. pg_amop 2017
51.5. pg_amproc 2018
51.6. pg_attrdef 2019
51.7. pg_attribute 2019
51.8. pg_authid 2021
51.9. pg_auth_members 2022
51.10. pg_cast 2022
51.11. pg_class 2023
51.12. pg_collation 2025
51.13. pg_constraint 2026
51.14. pg_conversion 2028
51.15. pg_database 2028
51.16. pg_db_role_setting 2029
51.17. pg_default_acl 2030
51.18. pg_depend 2030
51.19. pg_description 2032
51.20. pg_enum 2032
51.21. pg_event_trigger 2033
51.22. pg_extension 2033
51.23. pg_foreign_data_wrapper 2034
51.24. pg_foreign_server 2034
51.25. pg_foreign_table 2035
51.26. pg_index 2035
51.27. pg_inherits 2037
51.28. pg_init_privs 2037
51.29. pg_language 2037
51.30. pg_largeobject 2038
51.31. pg_largeobject_metadata 2038
51.32. pg_namespace 2039
51.33. pg_opclass 2039
51.34. pg_operator 2040
51.35. pg_opfamily 2041
51.36. pg_parameter_acl 2041
51.37. pg_partitioned_table 2041
51.38. pg_policy 2042
51.39. pg_proc 2043
51.40. pg_publication 2045
51.41. pg_publication_namespace 2045
51.42. pg_publication_rel 2046
51.43. pg_range 2046
51.44. pg_replication_origin 2047

51.45. pg_rewrite 2047
51.46. pg_seclabel 2047
51.47. pg_sequence 2048
51.48. pg_shdepend 2048
51.49. pg_shdescription 2049
51.50. pg_shseclabel 2050
51.51. pg_statistic 2050
51.52. pg_statistic_ext 2051
51.53. pg_statistic_ext_data 2052
51.54. pg_subscription 2053
51.55. pg_subscription_rel 2054
51.56. pg_tablespace 2054
51.57. pg_transform 2055
51.58. pg_trigger 2055
51.59. pg_ts_config 2056
51.60. pg_ts_config_map 2057
51.61. pg_ts_dict 2057
51.62. pg_ts_parser 2058
51.63. pg_ts_template 2058
51.64. pg_type 2059
51.65. pg_user_mapping 2062
52. ####### 2063
52.1. ## 2063
52.2. pg_available_extensions 2064
52.3. pg_available_extension_versions 2064
52.4. pg_backend_memory_contexts 2065
52.5. pg_config 2065
52.6. pg_cursors 2066
52.7. pg_file_settings 2066
52.8. pg_group 2067
52.9. pg_hba_file_rules 2067
52.10. pg_ident_file_mappings 2068
52.11. pg_indexes 2069
52.12. pg_locks 2069
52.13. pg_matviews 2071
52.14. pg_policies 2072
52.15. pg_prepared_statements 2072
52.16. pg_prepared_xacts 2073
52.17. pg_publication_tables 2073
52.18. pg_replication_origin_status 2074
52.19. pg_replication_slots 2074
52.20. pg_roles 2076
52.21. pg_rules 2077
52.22. pg_seclabels 2077
52.23. pg_sequences 2077
52.24. pg_settings 2078
52.25. pg_shadow 2080
52.26. pg_shmem_allocations 2081
52.27. pg_stats 2081
52.28. pg_stats_ext 2082
52.29. pg_stats_ext_exprs 2083
52.30. pg_tables 2085
52.31. pg_timezone_abbrevs 2085
52.32. pg_timezone_names 2086
52.33. pg_user 2086
52.34. pg_user_mappings 2087
52.35. pg_views 2087
52.36. pg_wait_events 2088

53. #######/########### 2089
53.1. ## 2089
53.1.1. ######### 2089
53.1.2. ######### 2089
53.1.3. ######## 2090
53.2. ######## 2090
53.2.1. ## 2090
53.2.2. ####### 2092
53.2.3. ####### 2094
53.2.4. ####### 2096
53.2.5. ###### 2097
53.2.6. COPY## 2097
53.2.7. ##### 2098
53.2.8. ############## 2099
53.2.9. ## 2099
53.2.10. SSL######## 2099
53.2.11. GSSAPI######## 2100
53.3. SASL## 2101
53.3.1. SCRAM-SHA-256 ## 2101
53.4. ################### 2102
53.5. ######################## 2111
53.5.1. ######################## 2111
53.5.2. ######################## 2111
53.5.3. ######################## 2112
53.6. ######### 2112
53.7. ######## 2113
53.8. ################## 2128
53.9. ################## 2130
53.10. #####2.0######### 2138
54. PostgreSQL######## 2140
54.1. ## 2140
54.2. ############## 2140
54.3. ################ 2143
54.4. ########### 2147
55. ####### 2149
55.1. #### 2149
55.1.1. #### 2149
55.1.2. ## 2149
55.1.3. ############### 2150
55.1.4. PO####### 2150
55.2. ###### 2151
55.2.1. ### 2151
55.2.2. ######### 2152
56. ############ 2154
57. ############ 2156
57.1. ########### 2156
57.2. ################### 2156
57.2.1. ##############FDW#### 2156
57.2.2. ###################FDW#### 2158
57.2.3. ####/#################FDW#### 2159
57.2.4. ############FDW#### 2159
57.2.5. TRUNCATE####FDW#### 2164
57.2.6. ########FDW#### 2165
57.2.7. EXPLAIN####FDW#### 2166
57.2.8. ANALYZE####FDW#### 2166
57.2.9. IMPORT FOREIGN SCHEMA####FDW#### 2167
57.2.10. ##########FDW#### 2167
57.2.11. #########FDW#### 2168

57.2.12. ##############FDW#### 2169
57.3. ############## 2169
57.4. ################### 2170
57.5. ############### 2172
58. ################## 2174
58.1. ################ 2174
59. ################ 2177
59.1. ############# 2177
59.1.1. ################# 2178
59.2. ############# 2178
59.2.1. ################# 2179
59.3. ########## 2179
59.3.1. ################# 2179
60. ############### 2182
60.1. ################### 2182
60.2. ######### 2182
60.3. PostgreSQL#############GEQO# 2183
60.3.1. GEQO############ 2184
60.3.2. PostgreSQL GEQO######## 2184
60.4. ######### 2184
61. ###################### 2185
62. ######################## 2186
62.1. ##########API## 2186
62.2. ################ 2188
62.3. ########## 2193
62.4. ################## 2194
62.5. ########## 2195
62.6. ############# 2196
63. Write Ahead Logging for Extensions 2199
63.1. ##WAL#### 2199
63.2. ####WAL######### 2200
64. Built-in Index Access Methods 2203
64.1. B-Tree###### 2203
64.1.1. #### 2203
64.1.2. B-Tree########### 2203
64.1.3. B-Tree###### 2204
64.1.4. ## 2206
64.2. GiST###### 2208
64.2.1. #### 2208
64.2.2. ########## 2209
64.2.3. ### 2211
64.2.4. ## 2222
64.2.5. # 2222
64.3. SP-GiST###### 2223
64.3.1. #### 2223
64.3.2. ########## 2223
64.3.3. ### 2225
64.3.4. ## 2232
64.3.5. # 2233
64.4. GIN###### 2233
64.4.1. #### 2233
64.4.2. ########## 2234
64.4.3. ### 2234
64.4.4. ## 2236
64.4.5. GIN### 2237
64.4.6. #### 2238
64.4.7. # 2238
64.5. BRIN###### 2238

64.5.1. #### 2239
64.5.2. ######### 2239
64.5.3. ### 2247
64.6. ######### 2250
64.6.1. ## 2250
64.6.2. ## 2251
65. ############# 2252
65.1. ################ 2252
65.2. TOAST 2253
65.2.1. #######TOAST## 2254
65.2.2. #######TOAST## 2255
65.3. ###### 2255
65.4. ##### 2256
65.5. ###### 2256
65.6. ################ 2256
65.6.1. ########### 2258
65.7. ########(HOT) 2258
66. ######### 2260
66.1. ############ 2260
66.2. ############ 2260
66.3. ######### 2260
66.4. 2######## 2261
67. ################ 2262
67.1. ############## 2262
67.2. ############# 2263
67.2.1. ######## 2263
67.2.2. OID### 2264
67.2.3. OID#### 2265
67.2.4. ####### 2265
67.2.5. ############ 2265
67.3. BKI###### 2267
67.4. BKI#### 2267
67.5. BKI################ 2268
67.6. BKI## 2268
68. ################### 2269
68.1. ##### 2269
68.2. ###### 2273
68.2.1. ##### 2273
68.2.2. ###N##### 2274
68.2.3. MCV### 2275
68.3. ################ 2276
69. ############## 2278
69.1. ######################### 2278
69.2. ######################## 2278
69.3. #############WAL######## 2279

#50# PostgreSQL#####

##

###[sim98]#### #########O.Univ.Prof.Dr. Georg Gottlob#Univ.Ass.Mag. Katrin Seyr.## ####Stefan Simkovics#####################

####PostgreSQL#################### ## ## ##

50.1.

##################################

1. ##############PostgreSQL#################### ############################## ##################

2. ##

3. ## ## ######################

   ############1################### ## ##

4. ####/###

   ################################ ##################################2####### ### 1###################1################### ############################## ############# ##

5. ### ################################ ##################################

#########PostgreSQL##

50.2.

PostgreSQL##1#####1##########/############### ######################1######### ######## ### ## ##############*postmaster*#########TCP/IP################### ################### ############ ##*instance*################### ###################

###########53#######PostgreSQL########################## ##########libpq C###### ##########Java JDBC##############################

########

50.3.

######2###############

- `gram.y#scan.l`############Unix####bison#flex###########
- ################################

50.3.1.

######################
########Unix##########bison#flex#############

########`scan.l`#########*SQL*################ ################################
######

#######`gram.y`## ############C##
#######################

####`scan.l`#####flex####C#######`scan.c`####### ###`gram.y`#bison####`gram.c`####
########################C################ #####C#####################
#######flex####bison #################

##
###################PostgreSQL############*makefile*##############

bison###`gram.y`########################## flex#bison################### `gram.y`#
##########bison########### ##################################

50.3.2.

#######SQL################################ ################################
####### ##
###################### ################################

##
########################### ############################BEGIN#ROLLBACK######
###########SELECT####UPDATE#######
####################
######

####
#######`FuncCall`################################ ###########################
###`FuncExpr`####`Aggref`############### ######################################
####

50.4. PostgreSQL#######

PostgreSQL## ##PostgreSQL#######
#2##############

- ###1#################################### ############################### ###
  ##1995#####Berkeley Postgres#############Postgres95##############

- #######2######################### ####################/################### ###
  ##############

################39########################### ################################
######################

50.5. ####/########

####/######################## #######SQL#################################### ###################### ##

> ##
>
> ## ######## ############### ##PostgreSQL####### ########################60###########geqo_threshold#########

#################################### ## ## ############################## ############################## ########################

50.5.1.

####/## ############ ######################### ## ####### #################B-tree#################relation.attribute OPR constant############## ##relation.attribute#B-tree##############OPR####### ##################1#####################B-tree##################### ######## ## ####### ##ORDER BY###

###

- ########## ################################1######### ######################### #### ## ######################### ##########################
- ###### ################################## ####2################################ ##### ###############1########################## ############################ ##
- ####### ### ################ ##

######3####################2#################################### ############### ############################

######geqo_threshold## #### #WHERE######################where rel1.attr1=rel2.attr2################### ###################### ## ################ ##

geqo_threshold####################60######################## ################

###1######WHERE##

50.6.

############/## ##############
#demand-pull########### ##############1##################################

###################MergeJoin################ #########################1####2
################## ######################################lefttree############
#######################Sort##
Sort#######################SeqScan############
###################### Sort#################################### ###########
########NULL#######Sort#################################### ##############
##

MergeJoin############################## ###2##################################
############## ##
################################## ########################MergeJoin########
###############NULL##########

############################
##

##########5######SQL######################### 5##SQL########
#SELECT#INSERT#UPDATE#DELETE####MERGE### SELECT##########################
INSERT ...
SELECT#UPDATE#DELETE#MERGE######ModifyTable###################
#SELECT###

INSERT ... SELECT######ModifyTable######## UPDATE######################
#############*TID*####ID#####ID####### ######ModifyTable####################
############################## DELETE###############1####TID##ModifyTable##
##########################TID###### MERGE##############################WHEN#
#######################TID###### ######ModifyTable#############WHEN########
##################################

###INSERT ... VALUES######1##Result################################1#####
###############ModifyTable#######

#51#

PostgreSQL# ################## ## ################## ######################### #######SQL#############CREATE DATABASE###pg_database#####1########################## ################### ##################SQL##

51.1.

51.1###############

########################

#51.1 ########

#####	##
pg_aggregate	####
pg_am	##############
pg_amop	###########
pg_amproc	##############
pg_attrdef	#######
pg_attribute	###########
pg_authid	##########
pg_auth_members	############
pg_cast	############
pg_class	###############################
pg_collation	############
pg_constraint	######################
pg_conversion	###########
pg_database	###################
pg_db_role_setting	################
pg_default_acl	################
pg_depend	##################
pg_description	####################
pg_enum	############
pg_event_trigger	#######
pg_extension	###########
pg_foreign_data_wrapper	############
pg_foreign_server	########
pg_foreign_table	###########
pg_index	##########
pg_inherits	########
pg_init_privs	###########

#####	##
pg_language	######
pg_largeobject	#################
pg_largeobject_metadata	###############
pg_namespace	####
pg_opclass	##############
pg_operator	###
pg_opfamily	###########
pg_parameter_acl	##############
pg_partitioned_table	####################
pg_policy	#############
pg_proc	########
pg_publication	###################
pg_publication_namespace	#################
pg_publication_rel	##################
pg_range	######
pg_replication_origin	###############
pg_rewrite	###########
pg_seclabel	######################
pg_sequence	###########
pg_shdepend	############
pg_shdescription	################
pg_shseclabel	########################
pg_statistic	#######
pg_statistic_ext	##############
pg_statistic_ext_data	##################
pg_subscription	###################
pg_subscription_rel	######################
pg_tablespace	##################
pg_transform	#################
pg_trigger	###
pg_ts_config	######
pg_ts_config_map	############
pg_ts_dict	#####
pg_ts_parser	######
pg_ts_template	#########
pg_type	####
pg_user_mapping	################

51.2. pg_aggregate

pg_aggregate#################### ################################1#######
################################ ###########sum#count###max###

pg_aggregate######pg_proc######### pg_proc#############################
############

#51.2 pg_aggregate##

# #	##
aggfnoid regproc #### pg_proc.oid#	pg_proc#####OID
aggkind char	####### n = ######### o = ########### h = ##########
aggnumdirectargs int2	##1########## #####pronargs## ## ########### #####
aggtransfn regproc #### pg_proc.oid#	####
aggfinalfn regproc #### pg_proc.oid#	###########
aggcombinefn regproc #### pg_proc.oid#	###########
aggserialfn regproc #### pg_proc.oid#	############
aggdeserialfn regproc #### pg_proc.oid#	##############
aggmtransfn regproc #### pg_proc.oid#	#######################
aggminvtransfn regproc #### pg_proc.oid#	####################
aggmfinalfn regproc #### pg_proc.oid#	###################
aggfinalextra bool	aggfinalfn############true
aggmfinalextra bool	aggmfinalfn############true
aggfinalmodify char	aggfinalfn############### r = ######### s = aggfinalfn## #aggtransfn######### w = #########
aggmfinalmodify char	aggmfinalfn#########aggfinalmodify###
aggsortop oid #### pg_operator.oid#	#################
aggtranstype oid #### pg_type.oid#	####################
aggtransspace int4	#####################################
aggmtranstype oid #### pg_type.oid#	#####################################
aggmtransspace int4	##

##
agginitval text ######## ######################### ######NULL##########NULL#######
aggminitval text ## ########NULL########### NULL########

########CREATE AGGREGATE########### ######################36.12##########

51.3. pg_am

pg_am############################## ####################1########## ########## ##################### ########################61###62##############

#51.3 pg_am##

##
oid oid ####
amname name #########
amhandler regproc #### pg_proc.oid# ##############################OID
amtype char t = ##################### i = ######

> ##
>
> PostgreSQL9.6#####pg_am## ######### C################## ####pg_index_column_has_property()############ ##########################SQL################## # 9.74##########

51.4. pg_amop

pg_amop## ########################## ##################################### ####1################################## ############ ##

#51.4 pg_amop##

##
oid oid ####
amopfamily oid #### pg_opfamily.oid# ##########
amoplefttype oid #### pg_type.oid# ############
amoprighttype oid #### pg_type.oid# ############

##
amopstrategy int2 #######
amoppurpose char ###### s = #### o = #####
amopopr oid #### pg_operator.oid# ####OID
amopmethod oid #### pg_am.oid# ##################
amopsortfamily oid #### pg_opfamily.oid# #######################B-tree################

##########################WHERE *indexed_column operator constant*#########
#################### ##
############

############################ORDER BY *indexed_column operator constant*#
############################### ##
######################### ORDER BY##################B-tree############
#amopsortfamily##########

##
##################################ASC NULLS LAST############ ########

###amopmethod###########opfmethod################amopmethod###########
######################## ###amoplefttype#amoprighttype#######
#pg_operator###oprleft#oprright############

51.5. pg_amproc

pg_amproc## ########################
###

#51.5 pg_amproc##

##
oid oid ####
amprocfamily oid #### pg_opfamily.oid# #########
amproclefttype oid #### pg_type.oid# ##################
amprocrighttype oid #### pg_type.oid# ################
amprocnum int2 #######
amproc regproc #### pg_proc.oid# ###OID

amproclefttype#amprocrighttype##
####### ## ############### #####
######################### ###amproclefttype#amprocrighttype#############
###opcintype############

51.6. pg_attrdef

pg_attrdef###################### #######pg_attribute######### ##########
##################

#51.6 pg_attrdef##

# #	##
oid oid	####
adrelid oid #### pg_class.oid#	###########
adnum int2 #### pg_attribute.attnum#	###
adbin pg_node_tree	nodeToString()########## pg_get_expr(adbin, adrelid)####SQL### #####

51.7. pg_attribute

pg_attribute#################### ##########################pg_attribute## ##### ###########pg_class##############################

#################################

#51.7 pg_attribute##

# #	##
attrelid oid #### pg_class.oid#	###########
attname name	##
atttypid oid #### pg_type.oid#	###################
attlen int2	######pg_type.typlen####
attnum int2	#### ######1############### ctid############################
attcacheoff int4	######-1##
atttypmod int4	atttypmod##########################varchar############ ############ ############ atttypmod#################-1###
attndims int2	############################ ### #################

##
attbyval bool   ######pg_type.typbyval####
attalign char   ######pg_type.typalign####
attstorage char   ########pg_type.typstorage##### TOAST###########################   #######
attcompression char   ########### ###############default_toast_compression#######'\0'### #####   ###'p'#pglz#######'l'#LZ4######## ####attstorage##################   #######
attnotnull bool   #NULL#######
atthasdef bool   ####################### #############pg_attrdef################ ##   ################attgenerated#########
atthasmissing bool   ################################## ################(non-volatile)   #DEFAULT################## ########attmissingval##########
attidentity char   ######''######IDENTITY######## IDENTITY#### a = GENERATED   ALWAYS# d = GENERATED BY DEFAULT######
attgenerated char   ######''############## ###### s = stored### ###################
attisdropped bool   ####################### ##SQL#   #########
attislocal bool   ####################### #####################################
attinhcount int2   ############### ########################
attcollation oid #### pg_collation.oid#   ###################################
attstattarget int2   #####[87.364621]#attstattarget#ANALYZE###########################   ######## ################### ######################################   ######################### ##########attstattarget###############   ################################## ######attstattarget############   #################ANALYZE####### ####################### NULL#####   ############################## ######################### ##########   ##attstattarget##
attacl aclitem[]   ##############################
attoptions text[]   #keyword=value###################
attfdwoptions text[]   #keyword=value###########################
attmissingval anyarray

##
################(non-volatile) #DEFAULT################## ####atthasmissing#true############ ###### ####NULL######

#######pg_attribute#####atttypid############ ###attlen#pg_type####### ################# ############################pg_type################### attlen##################################

51.8. pg_authid

pg_authid############################## ########################## ####### #rolcanlogin############### ########rolcanlogin########################## ################ pg_auth_members#########

pg_roles##pg_authid######

#21###########################

#########################pg_authid######################## ################# ###1###pg_authid#######

#51.8 pg_authid##

##
oid oid ####
rolname name ####
rolsuper bool ###################
rolinherit bool #########################
rolcreaterole bool ############
rolcreatedb bool ###############
rolcanlogin bool ##
rolreplication bool ################### ## #######
rolbypassrls bool ############################5.9##########
rolconnlimit int4 ###################################### -1###########
rolpassword text ######################NULL### ###################
rolvaliduntil timestamptz ######################## NULL###############

MD5##############rolpassword#####md5#######32###16#MD5########## MD5## ################################ ###joe######xyzzy###PostgreSQL#xyzzyjoe# MD5##########

######SCRAM-SHA-256##################

SCRAM-SHA-256$<*iteration count*>:<*salt*>$<*StoredKey*>:<*ServerKey*>

####*salt*#*StoredKey*#*ServerKey*#Base64########### #####RFC 5803[1]############## ##

##

51.9. pg_auth_members

pg_auth_members######################### ###########################

#########################pg_auth_members######################### pg_auth_members##################################

#51.9 pg_auth_members##

# #	##
oid oid	####
roleid oid #### pg_authid.oid#	#############ID
member oid #### pg_authid.oid#	roleid##########ID
grantor oid #### pg_authid.oid#	################ID
admin_option bool	member#roleid#####################true
inherit_option bool	##########################true
set_option bool	####SET ROLE#################true

51.10. pg_cast

pg_cast##################### ###########################

pg_cast## ########## ########################### ####################pg_cast########## ###########I/O######### #########text##############################I/O################### ####pg_cast#############

#51.10 pg_cast##

# #	##
oid oid	####

[1] https://datatracker.ietf.org/doc/html/rfc5803

# #
##
castsource oid #### pg_type.oid#
########OID
casttarget oid #### pg_type.oid#
#######OID
castfunc oid #### pg_proc.oid#
####################OID# ##############################
castcontext char
###################### e = ########################CAST###::######
a = ################################### i =
######
castmethod char
##################### f = castfunc########################### i = ##
################## b = ##############################

pg_cast#################1################################# ################# ############ #######3############### ########2######integer########### ##### ######################### 2###########-1### 3################boolean######## #### ########################true#############false#######

####################################pg_cast############### ##########length coercion functions######## #length coercion functions###########################

pg_cast##1################1########### ############ ##############################2########## 1##############2######## ########

51.11. pg_class

pg_class## ###########pg_index# #########pg_sequence##########################TOAST########### relkind#### ###### ## pg_class################ ####################

#51.11 pg_class##

# #
##
oid oid
####
relname name
###################
relnamespace oid #### pg_namespace.oid#
################OID
reltype oid #### pg_type.oid#
####################OID#######pg_type#####################TOAST###
##
reloftype oid #### pg_type.oid#
####################OID# ##############
relowner oid #### pg_authid.oid#
##########
relam oid #### pg_am.oid#

##
##################################_####_####_#############
relfilenode oid   ######################## ###   #########
reltablespace oid #### pg_tablespace.oid#   #################### ################################### ###########   ###################################### ##############################   ##############################
relpages int4   ####################BLCKSZ###### #####################   VACUUM#ANALYZE###CREATE INDEX#########DDL###########
reltuples float4   ############# ##################### VACUUM#ANALYZE#CREATE INDEX##   ####DDL########### ######VACUUM#ANALYZE##########reltuples###   ###########-1######
relallvisible int4   ############################## ####################   VACUUM#ANALYZE#CREATE INDEX######DDL###########
reltoastrelid oid #### pg_class.oid#   ############TOAST#####OID# ######### TOAST########################   #######
relhasindex bool   ##true
relisshared bool   ####################################true# #pg_database##############   ###########
relpersistence char   p = ######/###### u = ##########/###### t = ######/#####
relkind char   r = ####### i = ###### S = ###### t = TOAST##### v = #### m = ###########   c = #### f = ###### p = ########### I = ############
relnatts int2   ########################## pg_attribute####################   pg_attribute.attnum#########
relchecks int2   ######CHECK##### pg_constraint############
relhasrules bool   ########################true# pg_rewrite############
relhastriggers bool   ########################true# pg_trigger############
relhassubclass bool   ##true
relrowsecurity bool   ####################true# pg_policy############
relforcerowsecurity bool   ##true# pg_policy############
relispopulated bool

##
##
####################true##########################true###
relreplident char
###replica identity########## d = ############### n = ### f = ##### i = ###### # indisreplident##################################
relispartition bool
############################true
relrewrite oid #### pg_class.oid#
############DDL##################################OID###### ####### ##### ############### ##
relfrozenxid xid
###############ID################################ID########### ###### ################ID########pg_xact################################ ###### ####################InvalidTransactionId####
relminmxid xid
###############ID#########################ID# ##############ID#ID## ######pg_multixact## ### ################InvalidMultiXactId####
relacl aclitem[]
####### ###5.8########
reloptions text[]
#keyword=value########################
relpartbound pg_node_tree
###############relispartition#################

pg_class######################## ##########true####################true###### ###false############### ###relhasindex#CREATE INDEX########DROP INDEX##### ######## ####VACUUM##########################relhasindex####### ########## ################

51.12. pg_collation

pg_collation#########SQL## ### 23.2#########

#51.12 pg_collation##

##
##
oid oid
####
collname name
#########################
collnamespace oid #### pg_namespace.oid#
#############OID
collowner oid #### pg_authid.oid#
#######
collprovider char
#####[80.468750]######### d = ############ c = libc# i = icu ############ #:d=###########b=builtin#c=libc# i = icu
collisdeterministic bool

##	##
	##########
collencoding int4	####################################-1
collcollate text	##################LC_COLLATE# ######libc#####collcollate#null## ######colllocale########
collctype text	##################LC_CTYPE# ######libc#####collctype#null####### #colllocale########
colllocale text	################################## ######libc####colllocale#null### ####collcollate#collctype########
collicurules text	##########ICU####
collversion text	######################## ## ################

###########(collname, collnamespace)#####(collname,collencoding, collnamespace)### PostgreSQL####collencoding####################-1########## ########## ###collencoding = -1########################### ################### ############################SQL##(*schema.name*)############# ################## ##########initdb ### ################### ####################################

##template0##################################template0################### ################################ ##################

51.13. pg_constraint

#####[83.296214]#pg_constraint### ### ############# ######################## #NULL#########pg_attribute########## ##########pg_##null######### ########### ##### ##################### ########null###pg_##################### ####

#CREATE CONSTRAINT TRIGGER##################################

####################

#51.13 pg_constraint##

##	##
oid oid	####
conname name	##################
connamespace oid #### pg_namespace.oid#	###########OID
contype char	

##
#####[72.360248]#c = ##### f = ####### p = ###### u = ###### t = ###### x = ##   ## ######c=#####f=#######n=#null########p=######u=######t=######x=###   ##
condeferrable bool   ###########
condeferred bool   ##################
convalidated bool   ########### ####################false###########
conrelid oid #### pg_class.oid#   ############### #############
contypid oid #### pg_type.oid#   ############### ############
conindid oid #### pg_class.oid#   ## #########
conparentid oid #### pg_constraint.oid#   ############################### #########
confrelid oid #### pg_class.oid#   ################## #########
confupdtype char   ############## a = no action, r = restrict, c = cascade, n = set null, d = set default
confdeltype char   ############## a = no action, r = restrict, c = cascade, n = set null, d = set default
confmatchtype char   ######## f = full, p = partial, s = simple
conislocal bool   ##
coninhcount int2   ############## #########################
connoinherit bool   #####################################
conkey int2[] #### pg_attribute.attnum#   ###
confkey int2[] #### pg_attribute.attnum#   ##################
conpfeqop oid[] #### pg_operator.oid#   #########PK = FK###############
conppeqop oid[] #### pg_operator.oid#   #########PK = PK###############
conffeqop oid[] #### pg_operator.oid#   #########FK = FK###############
confdelsetcols int2[] #### pg_attribute.attnum#   #####SET NULL###SET DEFAULT################### NULL##############   #########
conexclop oid[] #### pg_operator.oid#   ####################
conbin pg_node_tree

##
###############pg_get_constraintdef()###########################

#######################conkey##### ######conkey########################### ############# #############conkey#pg_index.indkey##############

##
pg_class.relchecks##

51.14. pg_conversion

pg_conversion##################### ###CREATE CONVERSION#########

#51.14 pg_conversion##

##
oid oid ####
conname name #############
connamespace oid #### pg_namespace.oid# ###########OID
conowner oid #### pg_authid.oid# ######
conforencoding int4 ##########ID#pg_encoding_to_char()########################
contoencoding int4 #################ID#pg_encoding_to_char()########################
conproc regproc #### pg_proc.oid# ####
condefault bool ###############true

51.15. pg_database

pg_database########################## #######CREATE DATABASE########### ################22##########

##################pg_database############################ #######1######### ######1###pg_database##########

#51.15 pg_database##

##
oid oid ####
datname name ######

##	##
datdba oid	#### pg_authid.oid# ########################
encoding int4	################pg_encoding_to_char()#######################
datlocprovider char	#######################:b=builtin, c = libc, i = icu
datistemplate bool	true##################CREATEDB########## false#################### ##########
datallowconn bool	false#################### ###template0#########################
dathasloginevt bool	### ################pg_### #_################################## ########################### ######################## PostgreSQL
datconnlimit int4	############################## -1########-2###################
datfrozenxid xid	##########################ID#####################ID############## # ########################ID########pg_xact########################## ############ ########pg_class.relfrozenxid##########
datminmxid xid	################ID#######################ID# ###########ID######### ##pg_multixact## ####### ##pg_class.relminmxid#######
dattablespace oid	#### pg_tablespace.oid# ################### #########pg_class.reltablespace############## ################# ###################################
datcollate text	########LC_COLLATE
datctype text	########LC_CTYPE
datlocale text	############################## ######libc####datlocale#null### ### #datcollate#datctype#######
daticurules text	########ICU####
datcollversion text	####################### ## #################
datacl aclitem[]	####### ###5.8########

51.16. pg_db_role_setting

pg_db_role_setting##

##############pg_db_role_setting########################### ######## #pg_db_role_setting#####1######## ########1########

#51.16 pg_db_role_setting##

# #	##
setdatabase oid #### pg_database.oid#	################OID###############
setrole oid #### pg_authid.oid#	#############OID############
setconfig text[]	################

51.17. pg_default_acl

pg_default_acl#######################################

#51.17 pg_default_acl##

# #	##
oid oid	####
defaclrole oid #### pg_authid.oid#	############OID
defaclnamespace oid #### pg_namespace.oid#	#############OID#########
defaclobjtype char	############# r = ################ S = ###### f = ### T = ## n = ####
defaclacl aclitem[]	################################

pg_default_acl####################################### ##2###########
defaclnamespace = ##################################### ###################
############################ ##
#####

#######ACL###NULL#######pg_default_acl################################
######## pg_default_acl###################

51.18. pg_depend

pg_depend############################# #######DROP################DROP
CASCADE##############DROP RESTRICT##################

pg_shdepend######### ##

#51.18 pg_depend##

# #	##
classid oid #### pg_class.oid#	#######################OID
objid oid ######OID##	

##	##
	#############OID
objsubid int4	#################objid#classid############## #################### #####
refclassid oid #### pg_class.oid#	########################OID
refobjid oid ######OID##	##############OID
refobjsubid int4	#################refobjid#refclassid############## ############### ###########
deptype char	###########################

##########pg_depend## ### ##deptype##############################

DEPENDENCY_NORMAL (n)

####################### ################################## ##########
#CASCADE################### ###################### ######################
########

DEPENDENCY_AUTO (a)

################################## ########################RESTRICT###
#CASCADE#################################### ##########################
######################

DEPENDENCY_INTERNAL (i)

#############DROP###
############ ######################DROP############### ###########DROP#
######CASCADE################################### ########################
################NORMAL#AUTO#
#################################### ######ON SELECT##################
###################### ##

DEPENDENCY_PARTITION_PRI (P)
DEPENDENCY_PARTITION_SEC (S)

####INTERNAL##
####################### ##
########################CASCADE##############################
######## ###INTERNAL##
################ ##
##
######################### ##
############ ##################################1#######################
#####
#ATTACH/DETACH PARTITION########## ################################ ##
#######
################################### ####################################
##

DEPENDENCY_EXTENSION (e)

##############################pg_extension#### ######################### #DROP EXTENSION############ ##############INTERNAL################## pg_dump##################

DEPENDENCY_AUTO_EXTENSION (x)

##pg_dump######################### ############################## ############################### ############ ###AUTO##################pg_dump##################

################################

2###########pg_depend######################## ################################ ## ########################### #### ##################################CASCADE######### ##################### ##################################

initdb#####################(pinned)######## ################################ # ########################### ## #####pg_depend#################### ########numeric##########numeric#### #NORMAL###################pg_depend#######

51.19. pg_description

#####################################pg_description########### #### #COMMENT#########psql#\d########### ######################pg_description# ##############

pg_shdescription######### ## ##

#51.19 pg_description##

##	##
objoid oid	######OID## ################OID
classoid oid	#### pg_class.oid# ####################OID
objsubid int4	########################objoid####classoid################### ###### #####################
description text	#####################

51.20. pg_enum

pg_enum################################ ####################pg_enum########## #OID###

#51.20 pg_enum##

##	##
oid oid	

##	##
	####
enumtypid oid	#### pg_type.oid# ###########pg_type###OID
enumsortorder float4	#################
enumlabel name	############

pg_enum##OID################ ####################OID######### #########OID ############OID##########enumsortorder############ ###OID############## ##### ## #########################OID# ##############

################1..*n*################ ###################enumsortorder###### ########### ######################################

51.21. pg_event_trigger

pg_event_trigger################## ####38##########

#51.21 pg_event_trigger##

##	##
oid oid	####
evtname name	#################
evtevent name	######################
evtowner oid	#### pg_authid.oid# ##########
evtfoid oid	#### pg_proc.oid# ########
evtenabled char	##session_replication_role ######################## O = #####origin### ##local######## D = ####### R = #####replica######## A = #########
evttags text[]	################## NULL##################################

51.22. pg_extension

pg_extension############################## #########36.17#########

#51.22 pg_extension##

##	##
oid oid	

##	
	##
	####
extname name	
	#####
extowner oid #### pg_authid.oid#	
	######
extnamespace oid #### pg_namespace.oid#	
	###################
extrelocatable bool	
	####################true
extversion text	
	#########
extconfig oid[] #### pg_class.oid#	
	########regclass#OID#######NULL
extcondition text[]	
	###########WHERE##############NULL

##########################extnamespace############################## ##### ################ extnamespace#################################### extrelocatable#true##

51.23. pg_foreign_data_wrapper

pg_foreign_data_wrapper####################### ########################### ########

#51.23 pg_foreign_data_wrapper##

##	
	##
oid oid	
	####
fdwname name	
	############
fdwowner oid #### pg_authid.oid#	
	#############
fdwhandler oid #### pg_proc.oid#	
	################################### ###################
fdwvalidator oid #### pg_proc.oid#	
	## #
	###################
fdwacl aclitem[]	
	####### ###5.8#########
fdwoptions text[]	
	###########keyword=value#############

51.24. pg_foreign_server

pg_foreign_server################### ########################### ########## ################

#51.24 pg_foreign_server##

# #	##
oid oid	####
srvname name	#######
srvowner oid #### pg_authid.oid#	########
srvfdw oid #### pg_foreign_data_wrapper.oid#	###############OID
srvtype text	###########
srvversion text	###############
srvacl aclitem[]	###### ###5.8#########
srvoptions text[]	######keyword=value############

51.25. pg_foreign_table

pg_foreign_table########################## ########pg_class########### pg_foreign_table##

#51.25 pg_foreign_table##

# #	##
ftrelid oid #### pg_class.oid#	###########pg_class###OID
ftserver oid #### pg_foreign_server.oid#	################OID
ftoptions text[]	#keyword=value###################

51.26. pg_index

pg_index###################### ###########pg_class######

#51.26 pg_index##

# #	##
indexrelid oid #### pg_class.oid#	###########pg_class###OID
indrelid oid #### pg_class.oid#	####################pg_class###OID
indnatts int2	##########pg_class.relnatts####
indnkeyatts int2	

##
################################*included columns*############*key columns*# ##
indisunique bool true############
indnullsnotdistinct bool ######################### false################NULL############# #PostgreSQL#######################NULL############ NULL########## #####true###########1##NULL##############
indisprimary bool true######################### #####true####indisunique###true######### ##
indisexclusion bool true#########################
indimmediate bool true##################### #indisunique#true##########
indisclustered bool true###############################
indisvalid bool true############################ false############################ INSERT/UPDATE##################################### ################ #########
indcheckxmin bool true####pg_index##xmin#TransactionXmin########################### ######## ########################HOT###########################
indisready bool true########################## false##########INSERT/UPDATE########## ##############
indislive bool false######################################HOT#####################
indisreplident bool true############ALTER TABLE ... REPLICA IDENTITY USING INDEX ...#####replica identity########
indkey int2vector #### pg_attribute.attnum# ###############################indnatts###### ####1 3#1###3######## ################### ######INCLUDE################# ############## ##
indcollation oidvector #### pg_collation.oid# ########indnkeyatts##############################OID####### ###### ###############
indclass oidvector #### pg_opclass.oid# ########indnkeyatts#########################OID####### pg_opclass#########
indoption int2vector #############indnkeyatts###### ##################################
indexprs pg_node_tree ##################nodeToString()############ indkey############1# ################ ###################NULL######
indpred pg_node_tree ###########nodeToString()############ ############NULL#

51.27. pg_inherits

pg_inherits############################ ################################
###1################################

#51.27 pg_inherits##

##	##
inhrelid	oid #### pg_class.oid# ###############OID
inhparent	oid #### pg_class.oid# ###############OID
inhseqno	int4 ### 1###### ##
inhdetachpending	bool ##################true#######false#

51.28. pg_init_privs

pg_init_privs################################## ######################
#NULL############1#########

#############initdb#####################CREATE EXTENSION################
#GRANT## ##############################
##########################GRANT#REVOKE##########################
privtype#######initdb##############CREATE EXTENSION###################

initdb######################privtype#'i'##CREATE EXTENSION###############
#######privtype#'e'#####

#51.28 pg_init_privs##

##	##
objoid	oid #####OID## #########OID
classoid	oid #### pg_class.oid# ##################OID
objsubid	int4 ################objoid#classoid############## ################### #######
privtype	char ####################### ############
initprivs	aclitem[] ######## ###5.8########

51.29. pg_language

pg_language################################### #########CREATE
LANGUAGE##40##########

#51.29 pg_language##

# #	##
oid oid	####
lanname name	####
lanowner oid #### pg_authid.oid#	######
lanispl bool	#SQL##########false##########true### ###pg_dump##
lanpltrusted bool	##########true# ############SQL#################################### ###### ############################
lanplcallfoid oid #### pg_proc.oid#	################### ###################################### ######## ###
laninline oid #### pg_proc.oid#	##################DO#################### ################################
lanvalidator oid #### pg_proc.oid#	## ################################
lanacl aclitem[]	###### ###5.8########

51.30. pg_largeobject

pg_largeobject############################## ########################OID#### #### ##############pg_largeobject### #############BLCKSZ/4#######2#######LOBLKSIZE#########

PostgreSQL 9.0################################## ####pg_largeobject########## ##################OID############### ############## ##########OID########## ##pg_largeobject_metadata#########

#51.30 pg_largeobject##

# #	##
loid oid #### pg_largeobject_metadata.oid#	#####################
pageno int4	##############################
data bytea	################### LOBLKSIZE############ ##############

pg_largeobject#######################pageno * LOBLKSIZE###############1 ############## ##################################LOBLKSIZE################ ########################### ################################

51.31. pg_largeobject_metadata

pg_largeobject_metadata########################## ##############
#pg_largeobject#######

#51.31 pg_largeobject_metadata##

# #	##
oid oid	####
lomowner oid #### pg_authid.oid#	############
lomacl aclitem[]	####### ###5.8#########

51.32. pg_namespace

pg_namespace############### #####SQL############# ########################
###################

#51.32 pg_namespace##

# #	##
oid oid	####
nspname name	#######
nspowner oid #### pg_authid.oid#	########
nspacl aclitem[]	####### ###5.8#########

51.33. pg_opclass

pg_opclass############################## ##################################
###################### ##
##

##########36.16############

#51.33 pg_opclass##

# #	##
oid oid	####
opcmethod oid #### pg_am.oid#	######################
opcname name	##########
opcnamespace oid #### pg_namespace.oid#	############
opcowner oid #### pg_authid.oid#	

##	
	##
	#########
opcfamily	oid #### pg_opfamily.oid#
	###############
opcintype	oid #### pg_type.oid#
	#####################
opcdefault	bool
	######opcintype###########true
opckeytype	oid #### pg_type.oid#
	##################opcintype#######

######opcmethod#################opfmethod############# #####
#opcmethod#opcintype#########opcdefault#true#####pg_opclass###########
###

51.34. pg_operator

pg_operator################# CREATE OPERATOR#36.14##########

#51.34 pg_operator##

##	
	##
oid	oid
	####
oprname	name
	####
oprnamespace	oid #### pg_namespace.oid#
	############OID
oprowner	oid #### pg_authid.oid#
	#######
oprkind	char
	b = ############ l = ###### (####)
oprcanmerge	bool
	##################
oprcanhash	bool
	###################
oprleft	oid #### pg_type.oid#
	######################
oprright	oid #### pg_type.oid#
	#########
oprresult	oid #### pg_type.oid#
	###############shell###########
oprcom	oid #### pg_operator.oid#
	#########################
oprnegate	oid #### pg_operator.oid#
	########################
oprcode	regproc #### pg_proc.oid#
	######################shell###########
oprrest	regproc #### pg_proc.oid#

##	##
	#####################
oprjoin regproc	#### pg_proc.oid# #####################

51.35. pg_opfamily

pg_opfamily############### ## ################################ ## #### ## ##########

#########36.16##########

#51.35 pg_opfamily##

##	##
oid oid	####
opfmethod oid	#### pg_am.oid# ###################
opfname name	#######
opfnamespace oid	#### pg_namespace.oid# #########
opfowner oid	#### pg_authid.oid# ########

##################pg_opfamily##########pg_amop#pg_amproc#pg_opclass#### ###

51.36. pg_parameter_acl

####pg_parameter_acl######1########################## ##################### #######

#################pg_parameter_acl######################## ################ #pg_parameter_acl#####1########

#51.36 pg_parameter_acl Columns

##	##
oid oid	####
parname text	#################
paracl aclitem[]	####### ###5.8#########

51.37. pg_partitioned_table

####pg_partitioned_table#####################################

#51.37 pg_partitioned_table##

# #	##
partrelid oid #### pg_class.oid#	#############pg_class######OID
partstrat char	########### h = ############### l = ############## r = ############
partnatts int2	############
partdefid oid #### pg_class.oid#	######################pg_class#####OID# ########################## ######
partattrs int2vector #### pg_attribute.attnum#	###partnatts##################################### ######1 3####### ##1###3########################## ################################### ##############
partclass oidvector #### pg_opclass.oid#	##############################OID###### #######pg_opclass#########
partcollation oidvector #### pg_collation.oid#	####################################OID###### ##################### ##
partexprs pg_node_tree	#########################nodeToString()########### partattrs##### #######1################ ######################NULL######

51.38. pg_policy

####pg_policy########################## ####################################
###
######WITH CHECK####################

#51.38 pg_policy##

# #	##
oid oid	####
polname name	#######
polrelid oid #### pg_class.oid#	##############
polcmd char	################# r = SELECT# a = INSERT# w = UPDATE# d = DELETE# * = ###
polpermissive bool	##(permissive)########(restrictive)#####
polroles oid[] #### pg_authid.oid#	#################PUBLIC##############
polqual pg_node_tree	

# #
##
######################################
polwithcheck pg_node_tree
################WITH CHECK###############

##

pg_policy##################pg_class.relrowsecurity################
###

51.39. pg_proc

pg_proc######################WINDOW################################
CREATE FUNCTION#CREATE PROCEDURE#36.3#########

prokind#########################pg_aggregate#############

#51.39 pg_proc##

# #
##
oid oid
####
proname name
###
pronamespace oid #### pg_namespace.oid#
###########OID
proowner oid #### pg_authid.oid#
######
prolang oid #### pg_language.oid#
######################
procost float4
#######cpu_operator_cost###### proretset###################
prorows float4
#######proretset############
provariadic oid #### pg_type.oid#
##
prosupport regproc #### pg_proc.oid#
########################### #36.11###
prokind char
f = ###### p = ####### a = ##### w = WINDOW##
prosecdef bool
################setuid####
proleakproof bool
##################
proisstrict bool
#########NULL#####NULL###### ######################## ########### NULL######################

proretset bool
provolatile char provolatile## i = #immutable## ############################ s = #stable############################### ##### v = #volatile################################## ############ v #### ##################################
proparallel char proparallel######################## s = ########################### #### r = ## ################### ######### u = ################### ##############################
pronargs int2 #######
pronargdefaults int2 ############
prorettype oid #### pg_type.oid# #######
proargtypes oidvector #### pg_type.oid# ################# #######INOUT#VARIADIC########################## ######
proallargtypes oid[] #### pg_type.oid# ################# ####OUT#INOUT################ ##########IN###### ######NULL###### #######proargtypes#0################1########### #######
proargmodes char[] ################ ################ i = IN####### o = OUT####### b = INOUT###### # v = VARIADIC###### t = TABLE###### ########IN####### ######NULL### ###proargtypes####proallargtypes################## ###
proargnames text[] ############# ######################## ####################NULL### ###proargtypes####proallargtypes###################
proargdefaults pg_node_tree ##########nodeToString()########## ###pronargdefaults########## #*N*###################*N* proargtypes########### ################### #NULL######
protrftypes oid[] #### pg_type.oid# ####TRANSFORM########################## ####NULL#
prosrc text ################### ## #######
probin text ################ ################
prosqlbody pg_node_tree ########SQL###### ###########SQL##############SQL########### ##### ##NULL###
proconfig text[] ###################
proacl aclitem[]

##
####### ###5.8#########

##################################prosrc####C################## SQL##########
##################prosrc################ #######SQL##############prosrc#
####################prosqlbody#################### ####################
##prosrc################## probin#########C##############################
#########NULL###

51.40. pg_publication

####pg_publication################################ ################29.1#####
#####

#51.40 pg_publication##

##
oid oid ####
pubname name ###########
pubowner oid #### pg_authid.oid# ############
puballtables bool true###
pubinsert bool true######################INSERT##########
pubupdate bool true######################UPDATE##########
pubdelete bool true######################DELETE##########
pubtruncate bool true######################TRUNCATE##########
pubviaroot bool true### ############

51.41. pg_publication_namespace

####pg_publication_namespace#################################### #########
######

#51.41 pg_publication_namespace Columns

##
oid oid ####
pnpubid oid #### pg_publication.oid# ############
pnnspid oid #### pg_namespace.oid#

##	##
	#######

51.42. pg_publication_rel

####pg_publication_rel################################### ############### ################################52.17#########

#51.42 pg_publication_rel##

##	##
oid oid	####
prpubid oid #### pg_publication.oid#	###########
prrelid oid #### pg_class.oid#	#########
prqual pg_node_tree	####################nodeToString()######### ##################NULL ###
prattrs int2vector #### pg_attribute.attnum#	################################### #####1 3#############3######### ############# NULL#########################

51.43. pg_range

pg_range###################### ###pg_type################

#51.43 pg_range##

##	##
rngtypid oid #### pg_type.oid#	####OID
rngsubtype oid #### pg_type.oid#	################OID
rngmultitypid oid #### pg_type.oid#	###############OID
rngcollation oid #### pg_collation.oid#	################OID#########
rngsubopc oid #### pg_opclass.oid#	########################OID
rngcanonical regproc #### pg_proc.oid#	###############OID#########
rngsubdiff regproc #### pg_proc.oid#	###########double precision#######OID######

rngsubopc##################rngcollation#################### ##rngcanonical################### rngsubdiff################GiST######### ####################

51.44. pg_replication_origin

pg_replication_origin################################ ##################48 ##########

##################pg_replication_origin#################### ######## #pg_replication_origin####1###############1########

#51.44 pg_replication_origin##

##	
roident oid	######################## ################
roname text	###################

51.45. pg_rewrite

pg_rewrite############################

#51.45 pg_rewrite##

##	
oid oid	####
rulename name	####
ev_class oid #### pg_class.oid#	###########
ev_type char	############## 1 = SELECT# 2 = UPDATE# 3 = INSERT# 4 = DELETE
ev_enabled char	######session_replication_role################ O = #####origin#####local###### ## D = ####### R = #####replica######## A = #########
is_instead bool	####INSTEAD########true
ev_qual pg_node_tree	############nodeToString()###########
ev_action pg_node_tree	#############nodeToString()#############

##

####################pg_class.relhasrules#true##########

51.46. pg_seclabel

pg_seclabel################################## ##########SECURITY LABEL### ########### ########################52.22##########

pg_shseclabel######### ##

#51.46 pg_seclabel##

# #	##
objoid oid	#####OID## ######################OID
classoid oid	#### pg_class.oid# ####################OID
objsubid int4	##########################objoid###classoid############## ####### ##################
provider text	#####################
label text	#########################

51.47. pg_sequence

####pg_sequence################## ######################pg_class######

#51.47 pg_sequence##

# #	##
seqrelid oid	#### pg_class.oid# #######pg_class######OID
seqtypid oid	#### pg_type.oid# #########
seqstart int8	########
seqincrement int8	########
seqmax int8	########
seqmin int8	########
seqcache int8	#############
seqcycle bool	#############

51.48. pg_shdepend

pg_shdepend## ##### PostgreSQL###

pg_depend######### pg_depend##

##############pg_shdepend######################## ###################1# #pg_shdepend#########

#51.48 pg_shdepend##

# #	##
dbid oid #### pg_database.oid#	##################OID# ##############
classid oid #### pg_class.oid#	########################OID
objid oid ######OID##	##############OID
objsubid int4	##################objid#classid############## ##################### ########
refclassid oid #### pg_class.oid#	#########################OID##################
refobjid oid ######OID##	###############OID
deptype char	############################

###########pg_shdepend## ### ##deptype##############################

SHARED_DEPENDENCY_OWNER (o)

##

SHARED_DEPENDENCY_ACL (a)

#######################################ACL############################## ## ###SHARED_DEPENDENCY_OWNER#######SHARED_DEPENDENCY_ACL############ ##############

SHARED_DEPENDENCY_INITACL (i)

#####[52.808989]############pg_class###OID ################################ #########pg_init_privs###########

SHARED_DEPENDENCY_POLICY (r)

###

SHARED_DEPENDENCY_TABLESPACE (t)

###

############################ ## #####

pg_depend########initdb#####################(pinned)######## pg_shdepend#### ##

51.49. pg_shdescription

pg_shdescription################################## #####COMMENT########## ####psql#\d###############

pg_description######### ##

##############pg_shdescription########################## ###############
##1######pg_shdescription#1########

#51.49 pg_shdescription##

##	
	##
objoid oid	######OID##
	################OID
classoid oid	#### pg_class.oid#
	####################OID
description text	
	###################

51.50. pg_shseclabel

pg_shseclabel#################################### #########SECURITY LABEL########### ##################52.22#########

pg_seclabel######### ##

##############pg_shseclabel######################## pg_shseclabel######## ############1########

#51.50 pg_shseclabel##

##	
	##
objoid oid	######OID##
	#######################OID
classoid oid	#### pg_class.oid#
	####################OID
provider text	
	#####################
label text	
	#########################

51.51. pg_statistic

pg_statistic############################ ###ANALYZE#################### ## ##

################stainherit = false###1########## ########################## ##stainherit = true###2############ ########################SELECT *column* FROM *table**##################### ###stainherit = false###SELECT *column* FROM ONLY *table*########

pg_statistic######################### ############################## # #starelid############# ## # ##############stainherit = false######

##############################pg_statistic################################ ####NULL#################pg_statistic########## ##############1########## ################################## src/include/catalog/pg_statistic.h### #######

pg_statistic##

##
stxnamespace oid #### pg_namespace.oid# ##################OID
stxowner oid #### pg_authid.oid# ############
stxkeys int2vector #### pg_attribute.attnum# ################################### ######1 3########1###3########### #########
stxstattarget int2 #####[86.717268]#stxstattarget#ANALYZE############################ ######### ################### ##################################### ################################ ####stxstattarget################ ############ ######stxstattarget################################## ANALYZE####### ######################## NULL######################## ### stxstattarget########### ##############
stxkind char[] ##################### ###### d = N######## f = ########## m = ####### (MCV)#### e = ########
stxexprs pg_node_tree ###########################nodeToString()############ ############ ############## ########################NULL###

CREATE STATISTICS#####pg_statistic_ext################################ ## #######ANALYZE##############pg_statistic_ext_data###############

51.53. pg_statistic_ext_data

####pg_statistic_ext_data##pg_statistic_ext########################## ## ###########CREATE STATISTICS#####################

#####################stxdinherit = false###1########## ################### #######stxdinherit = true###2############ ########################SELECT * FROM *table**##################### ###stxdinherit = false###SELECT * FROM ONLY *table*########

pg_statistic###pg_statistic_ext_data################################## ########## ################################## pg_stats_ext############## #pg_statistic_ext#####pg_statistic_ext_data########################## ##########

#51.53 pg_statistic_ext_data##

##
stxoid oid #### pg_statistic_ext.oid# ######################
stxdinherit bool true###
stxdndistinct pg_ndistinct pg_ndistinct##########N#####
stxddependencies pg_dependencies

##	##
	pg_dependencies################
stxdmcv pg_mcv_list	pg_mcv_list##########MCV##############
stxdexpr pg_statistic[]	pg_statistic######################

51.54. pg_subscription

####pg_subscription################################## ################## ##29##########

##################pg_subscription#################### ######## #pg_subscription####1###############1########

#subconninfo##

#51.54 pg_subscription##

##	##
oid oid	####
subdbid oid #### pg_database.oid#	####################OID
subskiplsn pg_lsn	###LSN#######################LSN# #######0/0#
subname name	###########
subowner oid #### pg_authid.oid#	############
subenabled bool	true################################
subbinary bool	true##
substream char	#####[85.249458]################################ f = ################ ####### t = ## ############ p = ##u##### ### f=######################## ##t=## #######p=##t#####
subtwophasestate char	2########## d = ### p = ##### e = ##
subdisableonerr bool	true##
subpasswordrequired bool	true###################################
subrunasowner bool	true####################################
subfailover bool	

##	##
##	##################
subconninfo text	###############
subslotname name	### NULL#NONE#####
subsynccommit text	################synchronous_commit####
subpublications text[]	######################## ########################### ############## #29.1#########
suborigin text	#######none###any############### ######any### none############### ################################ any##############################

51.55. pg_subscription_rel

####pg_subscription_rel################################ ###############

#######CREATE SUBSCRIPTION####ALTER SUBSCRIPTION ... REFRESH PUBLICATION#####################################

#51.55 pg_subscription_rel##

##	##
srsubid oid #### pg_subscription.oid#	###########
srrelid oid #### pg_class.oid#	########
srsubstate char	###### i = #### d = ######## f = ########## s = ##### r = ################
srsublsn pg_lsn	s #### r ########################LSN# #######NULL

51.56. pg_tablespace

pg_tablespace############################ ################################ #########

#################pg_tablespace######################### #################pg_tablespace#####1########

#51.56 pg_tablespace##

##	##
oid oid	####
spcname name	######

##	##
spcowner oid	### pg_authid.oid# ##############################
spcacl aclitem[]	###### ###5.8#########
spcoptions text[]	#keyword=value######################

51.57. pg_transform

####pg_transform################ ########################## ####CREATE TRANSFORM##########

#51.57 pg_transform##

##	##
oid oid	####
trftype oid	#### pg_type.oid# ############OID
trflang oid	#### pg_language.oid# ###########OID
trffromsql regproc	#### pg_proc.oid# ######################################OID# ########################## ##
trftosql regproc	#### pg_proc.oid# ##################################OID# ###########################

51.58. pg_trigger

pg_trigger######################### CREATE TRIGGER##########

#51.58 pg_trigger##

##	##
oid oid	####
tgrelid oid	#### pg_class.oid# ##############
tgparentid oid	#### pg_trigger.oid# ## ###### ######
tgname name	#########################
tgfoid oid	#### pg_proc.oid# ########
tgtype int2	##################

##
tgenabled char ##session_replication_role#################### O = ######################### ### D = ######## R = #replica############## A = ###########
tgisinternal bool ######tgconstraint#############################true
tgconstrrelid oid #### pg_class.oid# ################# ###################
tgconstrindid oid #### pg_class.oid# ############################### ###################
tgconstraint oid #### pg_constraint.oid# #######pg_constraint#### ###############
tgdeferrable bool ###############true
tginitdeferred bool #########################true
tgnargs int2 #############
tgattr int2vector #### pg_attribute.attnum# #########################
tgargs bytea #########################
tgqual pg_node_tree ####WHEN######nodeToString()##############NULL
tgoldtable name OLD TABLE####REFERENCING########NULL
tgnewtable name NEW TABLE####REFERENCING########NULL

############UPDATE##################tgattr################## tgtype#####
###########################tgattr##########################

##
tgconstraint####### ##tgconstrrelid#tgconstrindid#tgdeferrable#tginitdeferred##### #pg_constraint############## ################################## ###### ##############################

##
pg_class.relhastriggers##########################true##########

51.59. pg_ts_config

pg_ts_config######################## ####################################
############### ####pg_ts_config########################
##pg_ts_config_map################

PostgreSQL##############12##########

#51.59 pg_ts_config##

# #	##
oid oid	####
cfgname name	##########
cfgnamespace oid #### pg_namespace.oid#	###########OID
cfgowner oid #### pg_authid.oid#	#######
cfgparser oid #### pg_ts_parser.oid#	#################OID

51.60. pg_ts_config_map

pg_ts_config_map################################ ########################## ########################

PostgreSQL##############12##########

#51.60 pg_ts_config_map##

# #	##
mapcfg oid #### pg_ts_config.oid#	###########pg_ts_config###OID
maptokentype int4	###################
mapseqno int4	##############mapseqno#####
mapdict oid #### pg_ts_dict.oid#	############OID

51.61. pg_ts_dict

pg_ts_dict######################## ##################################### # ## ################################ ######dictinitoption############## ####################

PostgreSQL##############12##########

#51.61 pg_ts_dict##

# #	##
oid oid	####
dictname name	##########

##	
dictnamespace oid	#### pg_namespace.oid# ###########OID
dictowner oid	#### pg_authid.oid# ######
dicttemplate oid	#### pg_ts_template.oid# ##################OID
dictinitoption text	###################

51.62. pg_ts_parser

pg_ts_parser######################### ##################################### ########## ####C###

PostgreSQL##############12##########

#51.62 pg_ts_parser##

##	
oid oid	####
prsname name	###########
prsnamespace oid	#### pg_namespace.oid# ############OID
prsstart regproc	#### pg_proc.oid# ########OID
prstoken regproc	#### pg_proc.oid# ############OID
prsend regproc	#### pg_proc.oid# #########OID
prsheadline regproc	#### pg_proc.oid# ##########OID########
prslextype regproc	#### pg_proc.oid# ##########OID

51.63. pg_ts_template

pg_ts_template############################ ############################## ## #####C###

PostgreSQL##############12##########

#51.63 pg_ts_template##

##	
oid oid	####

##	##
tmplname name	##############
tmplnamespace oid #### pg_namespace.oid#	###############OID
tmplinit regproc #### pg_proc.oid#	############OID#######
tmpllexize regproc #### pg_proc.oid#	###########OID

51.64. pg_type

pg_type################## ##############CREATE TYPE###########CREATE DOMAIN####### ## ####CREATE TYPE AS############

#51.64 pg_type##

##	##
oid oid	####
typname name	#####
typnamespace oid #### pg_namespace.oid#	###########OID
typowner oid #### pg_authid.oid#	#####
typlen int2	######typlen################ #########typlen##### -1##varlena######4 ##################-2######C###########
typbyval bool	typbyval########################### typlen#1#2#4########Datum# 8#########8###########typbyval#false########## ################ typbyval##########false################
typtype char	typtype### b = #### c = ################ d = ########### e = #### p = #### r = #### m = ####### typrelid###typbasetype#########
typcategory char	typcategory## # 51.65# ########
typispreferred bool	##typcategory##################true
typisdefined bool	###########true# ####################false# typisdefined#false####### ####OID##############
typdelim char	##################2########### ### ###
typrelid oid #### pg_class.oid#	

##
########typtype#########################pg_class######## ######## ##pg_class##########################pg_attribute################ ##############
typsubscript regproc #### pg_proc.oid# ########OID########################### ###########typsubscript = array_subscript_handler###### ################################### #######
typelem oid #### pg_type.oid# typelem##############################pg_type########### typsubscript################## ########################typelem#### ######typsubscript#################### typelem################## ######################### #############DDL######################
typarray oid #### pg_type.oid# typarray########typarray#pg_type############### ################ ################
typinput regproc #### pg_proc.oid# ##############
typoutput regproc #### pg_proc.oid# ##############
typreceive regproc #### pg_proc.oid# ####################
typsend regproc #### pg_proc.oid# ####################
typmodin regproc #### pg_proc.oid# ##############################
typmodout regproc #### pg_proc.oid# #########################
typanalyze regproc #### pg_proc.oid# ###ANALYZE################
typalign char typalign#################### ##############PostgreSQL############## ## ### ##### ################################### ############## • c = char################### • s = short##########2######### • i = int##########4######### • d = double##########8###############################
typstorage char typstorage##varlena##typlen = -1################################### ############################ ######### • p (plain)#####plain###############varlena############## • e (external)## #pg_class.reltoastrelid########## • m (main)###################

##	##
	• x (extended)## ############ x ######## m ##################################e # x ##### #########
typnotnull bool	typnotnull#####NULL####### #############
typbasetype oid #### pg_type.oid#	#########typtype########typbasetype################ ########### ###
typtypmod int4	#####typtypmod##############typmod###### #####typmod########-1## # ############-1###
typndims int4	typndims################## #####typbasetype####### ############## ####
typcollation oid #### pg_collation.oid#	typcollation############ ######################## ################ #################DEFAULT_COLLATION_OID### ######################## ###########################OID###########
typdefaultbin pg_node_tree	typdefaultbin#NULL#################nodeToString()##### ######## #####
typdefault text	###################typdefault#NULL### typdefaultbin#NULL#### ##typdefault##typdefaultbin################################ typdefaultbin#NULL#typdefault#NULL#####typdefault########### ##### ##################################
typacl aclitem[]	####### ###5.8########

##

################pg_type##

51.65##########typcategory####### ######################ASCII######## ###### ASCII#########################

#51.65 typcategory####

###	####
A	###
B	####
C	###
D	#####
E	###
G	####
I	###########

###	####
N	###
P	###
R	###
S	####
T	#####
U	######
V	#####
X	unknown#
Z	######

51.65. pg_user_mapping

pg_user_mapping################################## ########################### ### ####pg_user_mappings#########

#51.66 pg_user_mapping##

##
oid oid   ####
umuser oid #### pg_authid.oid#   ##################OID# ###################
umserver oid #### pg_foreign_server.oid#   ################OID
umoptions text[]   #keyword=value########################

#52#

##########PostgreSQL################### ################################## ################## #########################

#######35################################## #############PostgreSQL########## ########SQL###

52.1######################## ################################## ################## ##################### ##### 27.2#########

52.1.

52.1############## ################################## ##########################

#52.1 #######

####	##
pg_available_extensions	#######
pg_available_extension_versions	#############
pg_backend_memory_contexts	###############
pg_config	##############
pg_cursors	########
pg_file_settings	###########
pg_group	###############
pg_hba_file_rules	####################
pg_ident_file_mappings	###########################
pg_indexes	######
pg_locks	####################
pg_matviews	##########
pg_policies	####
pg_prepared_statements	######
pg_prepared_xacts	############
pg_publication_tables	###################
pg_replication_origin_status	###############################
pg_replication_slots	###############
pg_roles	########
pg_rules	###
pg_seclabels	########
pg_sequences	#####
pg_settings	#######
pg_shadow	#########
pg_shmem_allocations	#######
pg_stats	########
pg_stats_ext	##########

####	##
pg_stats_ext_exprs	##############
pg_tables	####
pg_timezone_abbrevs	######
pg_timezone_names	####
pg_user	#########
pg_user_mappings	########
pg_views	###
pg_wait_events	wait events

52.2. pg_available_extensions

pg_available_extensions######################### ################ #pg_extension#############

#52.2 pg_available_extensions##

##
name name ###
default_version text #######################NULL
installed_version text ##################################NULL
comment text ##################

pg_available_extensions############

52.3. pg_available_extension_versions

pg_available_extension_versions################################## ##### ############pg_extension#############

#52.3 pg_available_extension_versions##

##
name name ###
version text #######
installed bool true##############################
superuser bool true##############################trusted########
trusted bool true##################################
relocatable bool

##	##
	true####################
schema name	########################## ###########################NULL
requires name[]	################NULL
comment text	##################

pg_available_extension_versions############

52.4. pg_backend_memory_contexts

pg_backend_memory_contexts##
##

pg_backend_memory_contexts##1###################

#52.4 pg_backend_memory_contexts Columns

##	##
name text	###########
ident text	######################1024##########
parent text	###############
level int4	############TopMemoryContext#####
total_bytes int8	#####################
total_nblocks int8	#####################
free_bytes int8	#########
free_chunks int8	########
used_bytes int8	#########

#######pg_backend_memory_contexts############pg_read_all_stats########
####################

52.5. pg_config

pg_config################PostgreSQL########################## ####PostgreSQL
###
PostgreSQL################pg_config################

#######pg_config####################

#52.5 pg_config##

##
name text #####
setting text #####

52.6. pg_cursors

pg_cursors#################### #######################

- SQL##DECLARE####
- 53.2.3############/############Bind#######
- 45.1#######################SPI####

pg_cursors############################## ######WITH HOLD################### ################# ##

> ##
>
> ##########PostgreSQL########################## ######pg_cursors###### ###########################

#52.6 pg_cursors##

##
name text ######
statement text #######################
is_holdable bool true## #### #false
is_binary bool true#########BINARY######## #####false
is_scrollable bool true## #####false
creation_time timestamptz ############

pg_cursors############

52.7. pg_file_settings

pg_file_settings########################## ##########name = value############# ####1############################## ##############name = value################## ################

###########pg_settings###########

#######pg_file_settings###################

#52.7 pg_file_settings##

# #	##
sourcefile text	############
sourceline int4	###############
seqno int4	############1..*n*#
name text	#######
setting text	###########
applied bool	true#############
error text	NULL##################################

###applied######false###### ## #######error#######NULL#########1######### ########################## ##### ######################################error################## #####applied = false#################################### ###################error######## ########

##########################19.1#########

52.8. pg_group

pg_group################### #####8.1###PostgreSQL############## ##### ##rolcanlogin################################## ########################## ##

#52.8 pg_group##

# #	##
groname name #### pg_authid.rolname#	#######
grosysid oid #### pg_authid.oid#	#####ID
grolist oid[] #### pg_authid.oid#	#########ID#####

52.9. pg_hba_file_rules

pg_hba_file_rules##################pg_hba.conf############ ############## #####################1############################

##

#################pg_hba_file_rules###############

#52.9 pg_hba_file_rules##

rule_number int4 #####################NULL#
file_name text ##############
line_number int4 file_name##########
type text #####
database text[] #####################
user_name text[] ########################
address text #####IP########all#samehost#samenet###############NULL#
netmask text IP#################NULL
auth_method text ####
options text[] ######################
error text NULL##############################

#################line_number#####error##############

##################20##########

52.10. pg_ident_file_mappings

pg_ident_file_mappings#######################pg_ident.conf############
##################################1##############################

##########
##

#######pg_ident_file_mappings####################

#52.10 pg_ident_file_mappings Columns

map_number int4 ##############################NULL
file_name text ##############
line_number int4 file_name##########
map_name text ######

##	
sys_name text	################
pg_username text	#####PostgreSQL####
error text	NULL##############################

################line_number######error##############

##################20##########

52.11. pg_indexes

pg_indexes####################################

#52.11 pg_indexes##

##	
schemaname name #### pg_namespace.nspname#	####################
tablename name #### pg_class.relname#	##################
indexname name #### pg_class.relname#	########
tablespace name #### pg_tablespace.spcname#	##############################NULL#
indexdef text	############CREATE INDEX#####

52.12. pg_locks

pg_locks## ############## ##13##########

pg_locks##1######### ############## ### ####### #######################

########################### ## #ID##########ID#####################pg_description#pg_depend######OID### ####OID######## ##################pg_database.datfrozenxid################ ################## ######################################

#52.12 pg_locks##

##	
locktype text	############# relation# extend# frozenid# page# tuple# transactionid# virtualxid# spectoken# object# userlock# advisory# applytransaction####### ## 27.11##########
database oid #### pg_database.oid#	

# # # #

32) | objid::bigint############ #############classid###2#####
#objid######objsubid#2### ################### ############################
###database#########

############ID
#transactionid######## objsubid################# ########################
####0##1###

pg_locks##
relation####################pg_class.oid##########################
#database############OID###########################

#######################################pg_stat_activity####pid##pid######
#############

```
SELECT * FROM pg_locks pl LEFT JOIN pg_stat_activity psa
    ON pl.pid = psa.pid;
```

##
##virtualtransaction###pg_prepared_xacts####transaction######## #######
##############

```
SELECT * FROM pg_locks pl LEFT JOIN pg_prepared_xacts ppx
    ON pl.virtualtransaction = '-1/' || ppx.transaction;
```

pg_locks###
############ ## #######
##pg_locks###
################################## ##
##pg_blocking_pids()#### 9.69##################

pg_locks### ##############
###################### ######################## ###################fastpath
= true## #################
#################### ##
##
################# ##
########################
###

############
##

52.13. pg_matviews

pg_matviews##

#52.13 pg_matviews##

##
schemaname name #### pg_namespace.nspname# ###################
matviewname name #### pg_class.relname# ##############
matviewowner name #### pg_authid.rolname# #################
tablespace name #### pg_tablespace.spcname#

##	
	##
	##NULL#
hasindexes bool	
	true##
ispopulated bool	
	true############################
definition text	
	####################SELECT######

52.14. pg_policies

pg_policies##

#52.14 pg_policies##

##	
	##
schemaname name #### pg_namespace.nspname#	
	#########################
tablename name #### pg_class.relname#	
	##################
policyname name #### pg_policy.polname#	
	#######
permissive text	
	##(permissive)########(restrictive)#####
roles name[]	
	##############
cmd text	
	################
qual text	
	#####################################
with_check text	
	##################WITH CHECK###########

52.15. pg_prepared_statements

pg_prepared_statements################################ ############ #PREPARE##########

pg_prepared_statements###1################### ############################ #######DEALLOCATE#############

#52.15 pg_prepared_statements##

##	
	##
name text	
	########
statement text	
	################################## SQL############################
	#PREPARE#### #######/##

##	##
prepare_time timestamptz	############
parameter_types regtype[]	regtype###################### ########OID##regtype##oid######### ##############
result_types regtype[]	regtype############### ########OID##regtype##oid############### ###### ##################DML###########NULL######
from_sql bool	true##########PREPARE SQL############ false###########/########## #########
generic_plans int8	###########
custom_plans int8	#############

pg_prepared_statements############

52.16. pg_prepared_xacts

pg_prepared_xacts#######2#####################################PREPARE TRANSACTION##########

pg_prepared_xacts################1######## ############################## ########

#52.16 pg_prepared_xacts##

##	##
transaction xid	##############################
gid text	################################
prepared timestamptz	########################
owner name #### pg_authid.rolname#	################
database name #### pg_database.datname#	###################

pg_prepared_xacts## ############### ## ## ######################### ################

52.17. pg_publication_tables

#####[93.617021]#pg_publication_tables#################################### ###### ##########pg_publication_rel###########FOR ALL TABLES#FOR TABLES IN SCHEMA###1#######

#52.17 **pg_publication_tables##**

##
pubname name #### pg_publication.pubname# ##########
schemaname name #### pg_namespace.nspname# ##############
tablename name #### pg_class.relname# #######
attnames name[] #### pg_attribute.attname# ##################### ##
rowfilter text ##################

52.18. pg_replication_origin_status

pg_replication_origin_status############################ ###############
###48##########

#52.18 **pg_replication_origin_status##**

##
local_id oid #### pg_replication_origin.roident# ########
external_id text #### pg_replication_origin.roname# ########
remote_lsn pg_lsn ######################LSN
local_lsn pg_lsn ##remote_lsn#################LSN# ################################ ########################

52.19. pg_replication_slots

pg_replication_slots##

###################26.2.6##47##########

#52.19 **pg_replication_slots##**

##
slot_name name ########################
plugin name ################################NULL################
slot_type text ########physical###logical
datoid oid #### pg_database.oid# ###################OID####NULL############################
database name #### pg_database.datname#

##
########################NULL###############################
temporary bool   true######################### ### ###########
active bool   true###########################
active_pid int4   ##ID###########NULL#
xmin xid   ##################################### VACUUM ######################### #######
catalog_xmin xid   ### VACUUM############ #########################
restart_lsn pg_lsn   ####################LSN####LSN##max_slot_wal_keep_size################ ######################LSN#### #######LSN###########NULL###
confirmed_flush_lsn pg_lsn   ##############################LSN## ##LSN############################## ############ ##########NULL#
wal_status text   ###########WAL########### ##########   • reserved###########max_wal_size############   • extended#max_wal_size############################ #wal_keep_size###################   • unreserved###############WAL#################################### ############ #####reserved###extended###########   • lost####WAL###   ###2######max_slot_wal_keep_size############## restart_lsn#NULL##### ######NULL###
safe_wal_size int8   ########################WAL################## ############### #max_slot_wal_keep_size#-1##NULL###
two_phase bool   true##################################### ##########false#
inactive_since timestamptz   ########################### ##############NULL# ######################### ###########synced######true#####inactive_since##############47.2.3# ####
conflicting bool   ############################(true)################### ############# #invalidation_reason######## ############NULL###
invalidation_reason text   ################### ######################### ###############NULL# ##### ###########

##
• ######wal_removed#####WAL############## • ######rows_removed################### #################### • ######wal_level_insufficient######################wal_level##### ########## ###################
failover bool ## ############true# ############false#
synced bool ############# ################### true# ################### true ###### ################################## ############ ############# ######## ############### false #######################true ##########

52.20. pg_roles

pg_roles######################## ##############pg_authid################
#######

#52.20 pg_roles##

##
rolname name ####
rolsuper bool ###################
rolinherit bool ###########################
rolcreaterole bool ############
rolcreatedb bool ###############
rolcanlogin bool ##
rolreplication bool ################## ## ######
rolconnlimit int4 ##################################### -1###########
rolpassword text #############********#########
rolvaliduntil timestamptz ######################## NULL###############
rolbypassrls bool ##############################5.9##########
rolconfig text[] #####################
oid oid #### pg_authid.oid# ####ID

52.21. pg_rules

pg_rules#####################################

#52.21 pg_rules##

##
schemaname name #### pg_namespace.nspname# ##############
tablename name #### pg_class.relname# ############
rulename name #### pg_rewrite.rulename# ####
definition text ##################

pg_rules#########################ON SELECT######### ####pg_views## #pg_matviews######

52.22. pg_seclabels

pg_seclabels######################### ###pg_seclabel#####################

#52.22 pg_seclabels##

##
objoid oid ######OID## ######################OID
classoid oid #### pg_class.oid# ####################OID
objsubid int4 ###########################objoid###classoid############## ####### ##################
objtype text ###########################
objnamespace oid #### pg_namespace.oid# ########################OID########NULL
objname text #########################
provider text #### pg_seclabel.provider# #####################
label text #### pg_seclabel.label# #########################

52.23. pg_sequences

pg_sequences##

#52.23 pg_sequences##

schemaname name #### pg_namespace.nspname#
sequencename name #### pg_class.relname# #######
sequenceowner name #### pg_authid.rolname# ###########
data_type regtype #### pg_type.oid# #########
start_value int8 ########
min_value int8 ########
max_value int8 ########
increment_by int8 ########
cycle bool #############
cache_size int8 #############
last_value int8 ################### ## ###

last_value#########true######NULL#########

- ##################
- ###############USAGE####SELECT#########
- #######################

52.24. pg_settings

pg_settings########################## ####SHOW#SET################### ### ###########SHOW##

#52.24 pg_settings##

##
name text ##########
setting text ########
unit text ###########
category text ###########

##
short_desc text   ##########
extra_desc text   ###################
context text   ######################
vartype text   #######bool#enum#integer#real####string#
source text   ############
min_val text   ########################NULL#
max_val text   ########################NULL#
enumvals text[]   ########################NULL#
boot_val text   #####################################
reset_val text   ###########RESET##############
sourcefile text   ##   #pg_read_all_settings###################NULL#### #######   #include################
sourceline int4   ###   #pg_read_all_settings###################NULL####
pending_restart bool   true########################### #####false

context############ ###########################

internal

################ ###################### ############################initdb### ###################

postmaster

#################### ################################ ##########
#postgresql.conf## ###########
#context###################

sighup

####################postgresql.conf##################
postgresql.conf####################postmaster#SIGHUP############# ######
################postmaster#########SIGHUP##########

superuser-backend

####################postgresql.conf################## ################
######################libpq#PGOPTIONS##########################SET######

############## ##
postgresql.conf#############postgresql.conf############postmaster#
SIGHUP############## ###########################

backend

####################postgresql.conf################## ################
######################libpq#PGOPTIONS###### ############################
postgresql.conf########
#####postgresql.conf############postmaster#SIGHUP############## ########
###################

superuser

#######postgresql.conf############SET################### ###SET#######
######################SET################# postgresql.conf##############
##SET##############################

user

#######postgresql.conf############SET################### ###############
################### postgresql.conf###############SET#################
##############

############################19.1##########

############################## pg_settings########UPDATE##SET##############
############ pg_settings#######UPDATE##############SET################ ###
######################### ####################UPDATE###########################
#UPDATE############ ################################UPDATE####SET########
####################

###shared_preload_libraries#######
###########C########LOAD#################################### ################
###
###

52.25. pg_shadow

pg_shadow################### #####8.1###PostgreSQL#################
pg_authid##rolcanlogin######################

pg_user##pg_shadow##############
######################

#52.25 pg_shadow##

# #
usename name #### pg_authid.rolname#
usesysid oid #### pg_authid.oid#
usecreatedb bool
usesuper bool
userepl bool

##	
usebypassrls bool	########################## ####5.9##########
passwd text	################ #######NULL### ###################pg_authid#### ######
valuntil timestamptz	######################
useconfig text[]	################

52.26. pg_shmem_allocations

pg_shmem_allocations################################## ###PostgreSQL##### ######36.10.10##################################

##

#52.26 pg_shmem_allocations##

##	
name text	############ NULL################<anonymous>###
off int8	############# ##############NULL######
size int8	##########
allocated_size int8	################### ########################### #size#allocated_size######## ################################### #####

######ShmemInitStruct()####ShmemInitHash()#####ShmemAlloc()######### ##

######pg_shmem_allocations#########pg_read_all_stats################## #####

52.27. pg_stats

pg_stats####pg_statistic##################### ########################## #pg_statistic################### ############################

pg_stats## ######### #pg_statistic###

#52.27 pg_stats##

##	
schemaname name	#### pg_namespace.nspname# ##############

##	##
tablename name	#### pg_class.relname# ######
attname name	#### pg_attribute.attname# #########
inherited bool	true##
null_frac float4	NULL###########
avg_width int4	##############
n_distinct float4	##################### ########################## ################### #####ANALYZE################### ################################### ##-1#########################
most_common_vals anyarray	######################################NULL####
most_common_freqs float4[]	######################################most_common_vals#NULL### NULL####
histogram_bounds anyarray	############################### most_common_vals################### ####### #######<################most_common_vals################### ########NULL####
correlation float4	####[########]########################### ####-1##+1###### ##-1####+1# ### ####### #######<############NULL####
most_common_elems anyarray	##############NULL###################NULL####
most_common_elem_freqs float4[]	################################1################## 2#####3#######1# ############### ##############################NULL############ #most_common_elems#NULL###NULL####
elem_count_histogram float4[]	####NULL#######################NULL#######################NULL# ###
range_length_histogram anyarray	##################################### #########NULL## ######################################subtype_diff###############
range_empty_frac float4	################## #########NULL#
range_bounds_histogram anyarray	################################## ##########NULL## ########2##

########ALTER TABLE SET STATISTICS###################default_statistics_target# ########################

52.28. pg_stats_ext

pg_stats_ext####pg_statistic_ext#pg_statistic_ext_data##############
########## ####################
#pg_statistic_ext#pg_statistic_ext_data################### ###########
##################

pg_stats_ext## ######
#pg_statistic_ext###

#52.28 pg_stats_ext##

##	##
schemaname name	#### pg_namespace.nspname# #############
tablename name	#### pg_class.relname# #######
statistics_schemaname name	#### pg_namespace.nspname# #####################
statistics_name name	#### pg_statistic_ext.stxname# ##############
statistics_owner name	#### pg_authid.rolname# ##############
attnames name[]	#### pg_attribute.attname# ###################
exprs text[]	###############
kinds char[]	#######################
inherited bool	#### pg_statistic_ext_data.stxdinherit# true##
n_distinct pg_ndistinct	###########N######## ## ##################### ########ANALYZE############################ ######### ##################################### ####-1################ ###########################
dependencies pg_dependencies	########
most_common_vals text[]	##################### ############################NULL##
most_common_val_nulls bool[]	################NULL####### #most_common_vals#NULL##NULL##
most_common_freqs float8[]	##################################### #most_common_vals#NULL## NULL##
most_common_base_freqs float8[]	################################## #most_common_vals#NULL###NULL ####

##############ALTER TABLE SET STATISTICS#####################
#default_statistics_target#######################

52.29. pg_stats_ext_exprs

pg_stats_ext_exprs#####pg_statistic_ext#pg_statistic_ext_data#######
######################
#pg_statistic_ext#pg_statistic_ext_data#################### ###########
##################

##pg_stats_ext_exprs#################################### #######
#pg_statistic_ext##

#52.29 pg_stats_ext_exprs Columns

##
schemaname name #### pg_namespace.nspname# ###########
tablename name #### pg_class.relname# ######################
statistics_schemaname name #### pg_namespace.nspname# ###################
statistics_name name #### pg_statistic_ext.stxname# #############
statistics_owner name #### pg_authid.rolname# ##############
expr text ###############
inherited bool #### pg_statistic_ext_data.stxdinherit# true##
null_frac float4 NULL########
avg_width int4 ###########
n_distinct float4 ## ####### ##ANALYZE###################################### ################### ################### ####-1##
most_common_vals anyarray ###################### ############################NULL##
most_common_freqs float4[] ################################## #most_common_vals#NULL## NULL##
histogram_bounds anyarray ############################## most_common_vals################### ########### ###########<#########most_common_vals############### #NULL######
correlation float4 ########################### ###-1##+1### ##-1####+1################## ## ########<############# NULL######
most_common_elems anyarray ##############NULL##################NULL####
most_common_elem_freqs float4[]

##	##
	################################1################## 2#####3#######1# ############### ################################NULL############# #most_common_elems#NULL###NULL####
elem_count_histogram float4[]	####NULL#########################NULL#########################NULL# ###

ALTER TABLE SET STATISTICS################################## ### #default_statistics_target######################

52.30. pg_tables

pg_tables##

#52.30 pg_tables##

##	##
schemaname name	#### pg_namespace.nspname# ##############
tablename name	#### pg_class.relname# #######
tableowner name	#### pg_authid.rolname# ########
tablespace name	#### pg_tablespace.spcname# ################################NULL#
hasindexes bool	#### pg_class.relhasindex# true####################################
hasrules bool	#### pg_class.relhasrules# true###########################
hastriggers bool	#### pg_class.relhastriggers# true###########################
rowsecurity bool	#### pg_class.relrowsecurity# true###################

52.31. pg_timezone_abbrevs

pg_timezone_abbrevs## ######## ##timezone_abbreviations#####################

#52.31 pg_timezone_abbrevs##

##	##
abbrev text	######
utc_offset interval	UTC#########################
is_dst bool	

##
true##########

##############UTC###B.4### ####### ########################

52.32. pg_timezone_names

pg_timezone_names#####SET TIMEZONE#################### ######## #########UTC################### #PostgreSQL###################UTC######## pg_timezone_abbrevs################################## #################DST ########### ###########CURRENT_TIMESTAMP##############

#52.32 pg_timezone_names##

##
name text ####
abbrev text ######
utc_offset interval UTC#########################
is_dst bool true############

52.33. pg_user

pg_user############################## #################pg_shadow########### ######

#52.33 pg_user##

##
usename name ####
usesysid oid ###ID
usecreatedb bool #################
usesuper bool #############
userepl bool ##
usebypassrls bool ########################## ####5.9##########
passwd text #############********##########
valuntil timestamptz

##
######################
useconfig text[]   #################

52.34. pg_user_mappings

pg_user_mappings############################## ############################ ########################pg_user_mapping######

#52.34 pg_user_mappings##

##
umid oid #### pg_user_mapping.oid#   ########OID
srvid oid #### pg_foreign_server.oid#   ###############OID
srvname name #### pg_foreign_server.srvname#   #######
umuser oid #### pg_authid.oid#   ##################OID# ##########################
usename name   ##################
umoptions text[]   #keyword=value#######################

##################################umoptions############NULL#########

- ##USAGE#######
- ################PUBLIC############
- ################

52.35. pg_views

pg_views##

#52.35 pg_views##

##
schemaname name #### pg_namespace.nspname#   ##########
viewname name #### pg_class.relname#   ####
viewowner name #### pg_authid.rolname#   ######
definition text   ###########SELECT######

52.36. pg_wait_events

#########pg_wait_events#####################

#52.36 pg_wait_events Columns

##	##
type	text ################
name	text ##############
description	text #####[54.545455]### ################

#53# ########/############

PostgreSQL## #######TCP/IP## ##Unix################ #####5432####################TCP######IANA######### ## ##############################

#####PostgreSQL 7.4############3.0############## ######################PostgreSQL ##################### ############################## ############################ ######################### ##4.0############## ################################## #################3.1########3.0## #################NegotiateProtocolVersion############### ######################### ################################

##############################

53.1.

############################## ## ## ## ########################### ################## ##

#NOTIFY###

####################################

#######SQL###################################### ############################## ##################################### ### #################### ################################

###########COPY########################

53.1.1.

###################### ### ## ##################################### ######################### ############## ##

##

###########

53.1.2.

###############SQL#################### ##################################### ### ################################## ############################## ########### ################ ### #SELECT# ###################### #########SELECT##################################

##################################*parse*################################*bind*#####
#############*execute*######### ##########`SELECT`#`SHOW`##############################
##

################################## #####1#####################################
###########
############
##########################

53.1.3.

################################## PostgreSQL 7.4################################
######################### ######################## ############################
##########0###########1###
############

########C
################################

#################################### ##
######################

53.2.

################################### ####################53.7######## ##########
######`COPY`################################## ################################
##############################

53.2.1.

##############################
##
###################`pg_hba.conf`##
################

########################### ##
GSSAPI#SSPI#SASL################################## ########################
############################ GSSAPI#SSPI#SASL################################
###

#########################ErrorResponse#######AuthenticationOk##############

##################################

ErrorResponse

> ############# ##################

AuthenticationOk

> ##################

AuthenticationKerberosV5

> ################Kerberos V5###################Kerberos################### #####
> ########AuthenticationOk######## ######ErrorResponse######## ##################

AuthenticationCleartextPassword

> ########################PasswordMessage############## ###################
> AuthenticationOk######## ######ErrorResponse########

AuthenticationMD5Password

###########MD5#####################AuthenticationMD5Password################ ####################PasswordMessage############ ################## AuthenticationOk####### #####ErrorResponse####### ###PasswordMessage #concat('md5', md5(concat(md5(concat(password, username)), random-salt)))###SQL############ #md5()############################

AuthenticationGSS

###########GSSAPI################ ###########GSSAPI#################### ##GSSResponse####### ####################AuthenticationGSSContinue#######

AuthenticationSSPI

###########SSPI############### ###########SSPI######################## GSSResponse####### ####################AuthenticationGSSContinue#######

AuthenticationGSSContinue

##########GSSAPI###SSPI########AuthenticationGSS#AuthenticationSSPI###### AuthenticationGSSContinue################# ########GSSAPI###SSPI########### ######################################GSSResponse################### ###### ##GSSAPI###SSPI##############AuthenticationOk###################### ErrorResponse################

AuthenticationSASL

#########################SASL###1#####SASL############### ######SASL### ####################SASLInitialResponse################ ################## AuthenticationSASLContinue####### #######53.3#########

AuthenticationSASLContinue

##########SASL###########AuthenticationSASL######AuthenticationSASLContinue## ############## ########SASLResponse##################

AuthenticationSASLFinal

#####################SASL######## ############AuthenticationOk############ ###ErrorResponse####### ########SASL## #######

NegotiateProtocolVersion

####################################_pq_.############ ##################### ##########ErrorResponse############################

###

AuthenticationOk## ##################### ######################### ########ErrorResponse################################ #NegotiateProtocolVersion#########################ParameterStatus#### ##BackendKeyData#######ReadyForQuery#######

##########ErrorResponse#########

#####################################

BackendKeyData

    ## #########
    ##################ReadyForQuery####################

ParameterStatus

    ########################client_encoding#DateStyle######################### #####
    ### ###53.2.7########## ################
    ############ReadyForQuery####################

ReadyForQuery

    ############ ############################

ErrorResponse

    ############ #####################

NoticeResponse

    ################ ####################ReadyForQuery####ErrorResponse###########
    ##########

ReadyForQuery## ######################
##ReadyForQuery#########################ReadyForQuery##########################
#######

53.2.2.

########Query## ######################SQL
################### ################################1################
ReadyForQuery####### ReadyForQuery################################## ############
###############ReadyForQuery############# ######################################
##############################

##################################

CommandComplete

    SQL###############

CopyInResponse

    ## 53.2.6##########

CopyOutResponse

    ## 53.2.6##########

RowDescription

    SELECT#FETCH################################## ###########################
    #### ############################DataRow###########

DataRow

    SELECT#FETCH#################1####

EmptyQueryResponse

    ###################

ErrorResponse

###########

ReadyForQuery

################### ############SQL### ##### #CommandComplete###########1##SQL################### ################ ##########ReadyForQuery##########

NoticeResponse

######################### ########################### ######################### #####

SELECT###########EXPLAIN#SHOW############################RowDescription#0#### DataRow#############CommandComplete######## #########COPY############# #COPY#53.2.6#################### #############CommandComplete##############

ReadyForQuery###

##EmptyQueryResponse#####ReadyForQuery## ####

###########ErrorResponse#####ReadyForQuery######## ############################## ##############ErrorResponse########### ### #########

################################## ##############BINARY################### #FETCH############ ######################### RowDescription#################### #################

################################ErrorResponse#NoticeResponse###################### ##### ##53.2.7##########

####################

53.2.2.1.

##Query#######################SQL## ######################### ################

```
INSERT INTO mytable VALUES(1);
SELECT 1/0;
INSERT INTO mytable VALUES(2);
```

SELECT#############INSERT############## ################################ #INSERT###########

####################

```
BEGIN;
INSERT INTO mytable VALUES(1);
COMMIT;
INSERT INTO mytable VALUES(2);
SELECT 1/0;
```

###INSERT#####COMMIT################ ####INSERT#SELECT#################### ### ################INSERT##############INSERT#############

#######################Query###
##Query######################
################################### ##
#################

############BEGIN################################Query#####################
####################### #######Query######################COMMIT#ROLLBACK##
################################ ########Query#####BEGIN########Query########
#####################COMMIT#ROLLBACK############################ BEGIN##
####################
################

##################COMMIT#ROLLBACK#########################BEGIN##
#COMMIT#ROLLBACK###################### ####################################
#####

###

#######################Query################################### #######

```
BEGIN;
SELECT 1/0;
ROLLBACK;
```

#####Query#####################ROLLBACK#####################################
######################ROLLBACK########

#################
################################### ##
################## ## #
######################

```
BEGIN;
INSERT INTO mytable VALUES(1);
COMMIT;
INSERT INTO mytable VALUES(2);
SELCT 1/0;
```

#######################INSERT####################### #######################
##########################

53.2.3.

#######
###

###
######################Parse########### #####ParseComplete###ErrorResponse### #####
#######OID############ ##

##
##$*n*##############OID###### ##################### ###########void####void####OID################## ## ###############OUT################################### #####void#### ### ####$3#$4#void## ##########foo($1,$2,$3,$4)###########2##IN#2##OUT#############

> ##
>
> Parse####################SQL############# ################## ########### ### ###

#############Parse################## ####Query################################### ####### ##############Parse############################ ######################### ## #############SQL########PREPARE#EXECUTE#####################

#############Bind############################## Bind############################## ## # ## #Parse#######void######1##########Bind## ##########NULL####### ###Bind########################### ######################### ########### ###BindComplete####ErrorResponse###

> ##
>
> #######################SQL##########Bind################## ############## #############BINARY###########

############Bind#################### ## #######################Bind######################## ################################ ## ##########################

##################Bind################### ####Query############################## ### ############Bind############################## ######################### ######## ####SQL########DECLARE CURSOR#FETCH#####################

###########Execute######################### Execute############################## ##############fetch all rows############### ################################## ## ################################### Execute##########Execute##ReadyForQuery# RowDescription##

Execute##PortalSuspended####### ########## ####################################Execute#################### #####SQL### ###########CommandComplete######################### ######Execute##### CommandComplete#EmptyQueryResponse##########################ErrorResponse#### PortalSuspended############################

####################################Sync################### ################### #BEGIN/COMMIT## ################### ########################### ####ReadyForQuery########## Sync#################### ############ ##################################ErrorResponse#####Sync############ ########### ####ReadyForQuery#################### #####Sync#################### ############## #######Sync####ReadyForQuery#1################

> ##
>
> Sync#####BEGIN############################## ReadyForQuery############## ################################

##

Describe## ####################
####RowDescription######## ####################NoData######## ################
ErrorResponse###

Describe### ##############
########ParameterDescription################################RowDescription#########
############NoData######### ###################ErrorResponse######## Bind#########
RowDescription########################
######

> ###
>
> #################Execute##################################Describe#######
> #########

Close## ##############Close##############
#####CloseComplete#########################ErrorResponse######
################################

Flush########################### ### ###
##################################Flush#Sync############################## Flush#
###

> ##
>
> ##Parse#Bind######
> Describe#Execute#Close#Sync######## ################SQL#########bind/
> describe/execute########################### #######ParseComplete#
> BindComplete#CloseComplete#NoData#################

53.2.4.

##########################*(pipelining)*######### ##################################
######## ### ############################
################# #########################

`#############################################` `#####BEGIN … COMMIT######` ######
#######################################

#################################### ####################Sync####################
################sync###
`#BEGIN#COMMIT`## ######################
##Sync##################

`###########BEGIN`#########################Sync`######COMMIT`###########
`#ROLLBACK`######## ############################DDL`#####CREATE DATABASE`######
######################
################# ########################Sync##ReadyForQuery#################
####

#######################ReadyForQuery############Sync##########################
###

53.2.5.

##############################pg_proc#################################### ## ############################

##
######SELECT function($1, ...)########################## ################Bind/Execute#############

##################FunctionCall######################### #######1################ ##################ReadyForQuery########### ReadyForQuery######################## ########################

##################################

ErrorResponse

###########

FunctionCallResponse

################################ ## ##########

ReadyForQuery

################ #########################ReadyForQuery##########

NoticeResponse

######################### ######################### ######################### #####

53.2.6. COPY##

COPY################################### ## ################

##############################COPY FROM STDIN SQL############## ## ############CopyInResponse########### ################CopyData############## ############## ### ####### ##CopyDone####################CopyFail######COPY SQL######################### ################ ###########COPY### # #####CommandComplete#######ErrorResponse################

#########################CopyFail##########################ErrorResponse########## ## ##############COPY##################Sync############################ ####### Sync###########ReadyForQuery############## ############COPY# ######################ReadyForQuery########### ############################ CopyData#CopyDone#CopyFail#########

####################Flush#Sync########### ################################### ########### ############Execute#################COPY FROM STDIN########### ####Flush###Sync################Flush#Sync#######

##############################COPY TO STDOUT SQL###########
#######CopyOutResponse##########################CopyData###########1######
##CopyDone####### ###########COPY##################CommandComplete####### ####
###########Cancel########################### #######CopyData#CopyDone#########
########

#############################ErrorResponse################### ########
ErrorResponse#########################

NoticeResponse###ParameterStatus######CopyData################## ##################
########## ################53.2.7############### ######CopyData###CopyDone######
####################################

#########
##walsender###########START_REPLICATION################ #######
CopyBothResponse################### ################################CopyData##
################ ########CopyDone####################################CopyData
############# #####CopyDone## ######
#####################################ErrorResponse##########Sync################
#############ReadyForQuery############### ########################ErrorResponse#
############ ####CopyDone############## #########################53.4##########

CopyInResponse#CopyOutResponse#CopyBothResponse###############1################
######################## #########COPY######################### ###############
###############

53.2.7.

########################
################################## ####################################

##########NoticeResponse################### ##################################
##########################NoticeResponse####### ################################
#####NoticeResponse#######################

ParameterStatus###
####################SET SQL####################
#####SIGHUP### ###SET####################
#############ParameterStatus#############

######ParameterStatus#################### ###########

```
application_name                  is_superuser
client_encoding                   scram_iterations
DateStyle                         server_encoding
default_transaction_read_only     server_version
in_hot_standby                    session_authorization
integer_datetimes                 standard_conforming_strings
IntervalStyle                     TimeZone
```

#######default_transaction_read_only#in_hot_standby#14#################
scram_iterations#16####################
##server_version#server_encoding#integer_datetimes###################
##
#####ParameterStatus#############

########LISTEN#####################NOTIFY##################
NotificationResponse######NoticeResponse#################

##

###NotificationResponse########################## ##########################
####ReadyForQuery################ ################################### #
###########NotificationResponse####################

53.2.8.

################################### ######################################
#############
##

##################################StartupMessage#########CancelRequest###########
############# ##################### ######################################

CancelRequest####################################PID###################### ######
#############PID################################## ############################
##############################

########################### ## ##########
################################

################
######################## ##
###############

###
########################### ################################### ############
################################### ##
#############

53.2.9.

####################Terminate###################### ############################
######

####################
################################

####################
################################### ##
############# ##

###########SELECT##
####################
#BEGIN … COMMIT##

53.2.10. SSL########

PostgreSQL#SSL################################SSL############## ##############
################################## SSL#####PostgreSQL################18.9#######
###

SSL#######################StartupMessage####SSLRequest####### ###########SSL#
##############S#N####1############ ##
S##################SSL##############SSL##################### ##############

StartupMessage# # # # # # # #

GSSAPI#############`gss_wrap()`#######StartupMessage#################### ####### #########`gss_wrap()`#################4################## ####16kB########### ######################### #######`gss_wrap_size_limit()`################### ##############################`gss_wrap()`############ ###############8kB#### ###########8kB################16kB############## ####16kB################### ##################

###########GSSAPI## ###########

53.3. SASL##

SASL############################ #####PostgreSQL#2##SASL#####SCRAM-SHA-256# SCRAM-SHA-256-PLUS######## ##################### #######SASL################ #################SCRAM-SHA-256#SCRAM-SHA-256-PLUS#################

SASL###########

1. SASL################AuthenticationSASL########### ###################SASL# ##################################
2. ####################1########SASLInitialResponse########### ############## ##################Initial Client Response####################################
3. ##################################1######## ############## AuthenticationSASLContinue###########################SASLResponse######### ## ####################
4. #######################AuthenticationSASLFinal###############AuthenticationOk## ######### AuthenticationSASLFinal## ## ############################AuthenticationSASLFinal#############

############################ErrorMessage############

53.3.1. SCRAM-SHA-256

###########SASL###`SCRAM-SHA-256`#################`SCRAM-SHA-256-PLUS`### ## #RFC 7677[6]###RFC 5802[7]#########

PostgreSQL#SCRAM-SHA-256##############`client-first-message`############## ##### ############################## SCRAM#######UTF-8########### ##PostgreSQL####################PostgreSQL#####UTF-8##############

SCRAM###########UTF-8####*SASLprep*####################### ###PostgreSQL###### #UTF-8################## ######################################UTF-8######## SASLprep####### #########UTF-8##############SASLprep#####UTF-8############## #########SASLprep################### ###########UTF-8################UTF-8# ##

SSL######PostgreSQL####################### #############SCRAM####SASL# ###`SCRAM-SHA-256-PLUS`### PostgreSQL####################`tls-server-end-point`###

###############SCRAM## ## ## #######################

[6] https://datatracker.ietf.org/doc/html/rfc7677
[7] https://datatracker.ietf.org/doc/html/rfc5802

##############SCRAM## ###### ##SSL#########

#

1. ####AuthenticationSASL########### ###################SASL############### #### SSL##################SCRAM-SHA-256-PLUS#SCRAM-SHA-256############### #####

2. #######SASLInitialResponse################### #############SCRAM-SHA-256# ##SCRAM-SHA-256-PLUS###### ### ########################### Initial Client################SCRAM#client-first-message####### client-first-message########################### #####

3. ####AuthenticationSASLContinue########### #####SCRAM#server-first-message###

4. #######SASLResponse########### #####SCRAM#client-final-message###

5. ####SCRAM#server-final-message###AuthenticationSASLFinal############### AuthenticationOk###########

53.4.

###replication########### #####true### #on#yes#1############SQL##walsender #############

replication############database#######dbname############################ #############walsende############### ############walsender######################### ###SQL###############

############################walsender##########################

######################replication###################psql#####libpq############## ############### #######

```
psql "dbname=postgres replication=database" -c "IDENTIFY_SYSTEM;"
```

###################pg_receivewal####################pg_recvlogical################## ###

log_replication_commands################################

################################

IDENTIFY_SYSTEM

- ################### ####################################

  systemid(text)

  - ###################### ## ###########

  timeline(int8)

  - #########ID### ####################################

xlogpos (text)

###WAL########### ####################################

dbname (text)

##############NULL###

SHOW *name*

############################### ###SQL####SHOW######

name

############## ############19###########

TIMELINE_HISTORY *tli*

tli################################### ####2############### #######text## ################################

filename (text)

################00000002.history

content (text)

###############

CREATE_REPLICATION_SLOT *slot_name* [TEMPORARY] { PHYSICAL | LOGICAL *output_plugin* } [(*option* [, ...])]

############################ ################26.2.6####

slot_name

########## ######################### #26.2.6.1#####

output_plugin

######################### #47.6#####

TEMPORARY

################################ ## #######

###################

TWO_PHASE [*boolean*]

true####################2################### ##############PREPARE TRANSACTION#COMMIT PREPARED###ROLLBACK PREPARED###2############## ############## #########PREPARE TRANSACTION############### ######false ###

RESERVE_WAL [*boolean*]

true#######################WAL############## ########WAL############## ###################### ######false###

SNAPSHOT { 'export' | 'use' | 'nothing' }

################################### ######'export'################## ############### ############################### 'use'################

####################### ##########################
##CREATE_REPLICATION_SLOT################################# ##
##'nothing'##

FAILOVER [*boolean*]

######true ### ###### false ###

#############################1#############

slot_name (text)

#########################

consistent_point (text)

###################WAL##### ####################################

snapshot_name (text)

############################## ### ########### ################NULL######

output_plugin (text)

##################################### #################NULL######

CREATE_REPLICATION_SLOT *slot_name* [TEMPORARY] { PHYSICAL [RESERVE_WAL] | LOGICAL *output_plugin* [EXPORT_SNAPSHOT | NOEXPORT_SNAPSHOT | USE_SNAPSHOT | TWO_PHASE] }

#################CREATE_REPLICATION_SLOT##########################

ALTER_REPLICATION_SLOT *slot_name* (*option* [, ...])

#####[50.000000]########################### ################26.2.6#### ####################### ####################26.2.6########## ################### #################

slot_name

#####[81.343284]########### ######################### #26.2.6.1##### ######### ##########################26.2.6.1#####

The following option is supported:

FAILOVER [*boolean*]

######true ###

READ_REPLICATION_SLOT *slot_name*

########################### #####################NULL############## ################################

#############################1#############

slot_type (text)

################physical###NULL#

restart_lsn (text)

############restart_lsn#

`restart_tli` (`int8`)

###############`restart_lsn`#############ID#

`START_REPLICATION` [`SLOT` *`slot_name`*] [`PHYSICAL`] *`XXX/XXX`* [`TIMELINE` *`tli`*]

#######WAL########*`XXX/XXX`* WAL############### `TIMELINE`############## #######*`tli`*############### ########################### #############WAL ############################### #######CopyBothResponse################### WAL###########

`slot_name`### #############WAL##### #############`hot_standby_feedback`###################

###WAL ########### ##################################COPY#############

###############WAL###############COPY###################### #######COPY ######################2################################### ############ ID(`int8`#)############WAL###(`text`#)### #########WAL################### ###############WAL################## ###########CommandComplete######## CopyData#########`START_REPLICATION`################################

#####[69.053118]#WAL####CopyData############### ######################## ###########################ErrorResponse################## ############## CopyData########################## ######WAL######CopyData############## ### ###53.6#53.7######### ##################### ######################### ErrorResponse################# #############CopyData######################## ###########

XLogData (B)

Byte1('w')

######WAL###########

Int64

########WAL#####

Int64

#######WAL####

Int64

################ 2000#1#1###0#########

Byte*n*

WAL###########

###WAL#####2##XLogData################## ###WAL#####WAL######## ## ###########WAL######## #######XLogData###################

#################(B)

Byte1('k')

########################

Int64

########WAL###

Int64

################# 2000#1#1###0##########

Byte1

##1## 0######

###########1#####CopyData##

##########(F)

Byte1('r')

#######################

Int64

##########################WAL###+1####

Int64

########################WAL###+1####

Int64

################WAL###+1####

Int64

#################### 2000#1#1###0##########

Byte1

##1#################################### ############################ ####################

####################(F)

Byte1('h')

####################################

Int64

#################### 2000#1#1###0##########

Int32

#####[69.918699]################xmin#####################catalog_xmin### ### ######catalog_xmin#####0## ##### ##################################### ####################xmin# # ##############catalog_xmin#### #######catalog_xmin###0########### ### ############################ #########

Int32

#####[52.238806]#############xmin xid######## ##################xmin xid ######

Int32

#####[64.141414]#######################catalog_xmin###### ####### catalog_xmin#####################################0####### ########### ################catalog_xmin# #######catalog_xmin############### ########################0#######

Int32

#####[52.941176]#######catalog_xmin xid######## ############ #catalog_xminxid######

START_REPLICATION SLOT *slot_name* LOGICAL *XXX/XXX* [(*option_name* [*option_value*] [, ...])]

########*XXX/XXX*WAL#######confirmed_flush_lsn(52.19##)################# ######WAL################## ##############################LSN######## ############### ########LSN#####LSN#################################### ## ############START_REPLICATION#######confirmed_flush_lsn########### #############

################################## #########CopyBothResponse################ ###WAL###############

CopyBothResponse#########2##CommandComplete######## #START_REPLICATION ... PHYSICAL###########

##

SLOT *slot_name*

################ #############LOGICAL###### #CREATE_REPLICATION_SLOT##

XXX/XXX

#########WAL###

option_name

##(pgoutput)################### ####53.5##########

option_value

######## #########

DROP_REPLICATION_SLOT *slot_name* [WAIT]

################################## ########walsender######################## #########################

slot_name

##########

WAIT

####################

UPLOAD_MANIFEST

##

BASE_BACKUP [(*option* [, ...])]

################################## ######################################
####################### ##############

LABEL *'label'*

################ ##########base backup################# ##########
##standard_conforming_strings#######SQL########

TARGET *'target'*

################### ######client################################ ##
####server################TARGET_DETAIL#################### #####
#blackhole##############################

server####################pg_write_server_files################
##

TARGET_DETAIL *'detail'*

#########################

#################################### #########################

PROGRESS [*boolean*]

true############################## ######################################
############################## ###################################### #
####################### ###################################### #######
###
###false###

CHECKPOINT { 'fast' | 'spread' }

################################### ######spread###

WAL [*boolean*]

true##############WAL########## ########tar#####pg_wal###############
######################## ######false###

WAIT [*boolean*]

true###############WAL#################### WAL####################
false## ######true#
##

COMPRESSION *'method'*

################################## ##############gzip#lz4###zstd###

COMPRESSION_DETAIL *detail*

################# ###COMPRESSION################ ###################
#####################################*keyword*###*keyword=value*### #####
###########level#long###workers###

level################ gzip#########1##9#########
#Z_DEFAULT_COMPRESSION###-1##lz4####1##12##################
#0##zstd####ZSTD_minCLevel()###-131072###ZSTD_maxCLevel()###
#22########ZSTD_CLEVEL_DEFAULT###3########

long###################### ############################ #######zstd### ############

workers########################### #####zstd##############

MAX_RATE *rate*

########kB/s######/##### #### ##################32 kB##1 GB############ ############################

TABLESPACE_MAP [*boolean*]

true#########pg_tblspc##################tablespace_map############ ### ######################pg_tblspc/################################## ######## ######false###

VERIFY_CHECKSUMS [*boolean*]

true## false############ ##### #true###

MANIFEST *manifest_option*

#######yes###force-encode## ##################WAL################################ ############## ######################## force-encode################16############ ## ########################UTF-8############################ force-encode## #### ########################MANIFEST 'no'###

MANIFEST_CHECKSUMS *checksum_algorithm*

##NONE#CRC32C#SHA224, SHA256#SHA384#SHA512### ######CRC32C###

INCREMENTAL

########################## ########################UPLOAD_MANIFEST### #############

##CopyOutResponse########

##############1#2######################## ######XLogRecPtr################### #######ID#######

################################ ##################

spcoid (oid)

#######OID### ############NULL###

spclocation (text)

################### ############NULL###

size (int8)

####################################(1024###)#### ###########NULL###

###############CopyOutResponse######## #CopyData############################ ######

new archive (B)

Byte1('n')

######################### ##############1######################1#######
####### #######tar#############POSIX 1003.1-2008###########ustar#######
###

String

##############

String

###################### ##

manifest (B)

Byte1('m')

##############################

archive or manifest data (B)

Byte1('d')

##

Byte*n*

#######

progress report (B)

Byte1('p')

########################

Int64

######################

CopyOutResponse##############################WAL#########################
##############

##################tar######################PostgreSQL#######################
################ ################
- postmaster.pid
- postmaster.opts
- pg_internal.init###############
- PostgreSQL##############################pgsql_tmp####################
  ##############
- ##init##########
- ###########pg_wal# #######wal######################pg_wal####### #######
  ####################################
- pg_dynshmem#pg_notify#pg_replslot#pg_serial#pg_snapshots#pg_stat_tmp#pg_subt
  ##
- ##
  ############ #pg_tblspc##################
###

###############boolean##############*value*############## ###TRUE##########
##

53.5.

############################## ##################START_REPLICATION SLOT *slot_name* LOGICAL##############

###

PostgreSQL############################## ############################pgoutput#
##

53.5.1.

START_REPLICATION##########pgoutput################

proto_version

#############1#2#3####4########### ####################

#####2##########14##

#####3#########15##############2#####################

#####4##########16##

publication_names

############################
################### #####################

binary

######################## ##################################

messages

pg_logical_slot_peek_message################################

streaming

#################################### ##############################parallel##
####### ########################2############# #parallel####################4
############

two_phase

2##################### ########################3############

origin

######################## ######################none####################any##
##

53.5.2.

############################## ############53.9##########

######################################
###

###

53.5.3.

START_REPLICATION##

#######Begin#Commit#################### ######################### ####Begin Prepare#Prepare messages########################## ##### ### #####################

#########################DML######Insert#Update#Delete######## ################ ##Origin################ Origin## ######## ##Origin###### ###################### ####### Origin####################DML###########

####DML############################OID######### ###########OID###DML######### ################################## ## ################### ##

############OID############ ######################OID########################## ## ########OID###OID## ################ ####################################OID############################ ########## #####OID######

53.6.

#############################

Int*n*(*i*)

> ##########################*n*####### ##*i*####################### ########### ### Int16#Int32(42)#####

Int*n*[*k*]

> *n*#########*k*####################### #####*k*######################## ###Int16[M]# ##

String(*s*)

> NULL######C######################### *s*####################### ############ ###String#String("user")#####

##
###############

Byte*n*(*c*)

> ###*n*#### ######*n*################################ *c*####################### ### Byte2#Byte1('\n')#####

53.7.

############################## ##############F#########B########F & B############## ######### ## ############## #CopyData############################## ###CopyData############## #############

AuthenticationOk (B)

Byte1('R')

###################

Int32(8)

######################

Int32(0)

################

AuthenticationKerberosV5 (B)

Byte1('R')

###################

Int32(8)

######################

Int32(2)

Kerberos V5#################

AuthenticationCleartextPassword (B)

Byte1('R')

###################

Int32(8)

######################

Int32(3)

######################

AuthenticationMD5Password (B)

Byte1('R')

###################

Int32(12)

######################

Int32(5)

MD5######################

Byte4

###############

AuthenticationGSS (B)

Byte1('R')

###################

Int32(8)

######################

Int32(7)

GSSAPI#######################

AuthenticationGSSContinue (B)

Byte1('R')

###################

Int32

######################

Int32(8)

######GSSAPI####SSPI###############

Byte*n*

GSSAPI####SSPI######

AuthenticationSSPI (B)

Byte1('R')

###################

Int32(8)

######################

Int32(9)

SSPI###################

AuthenticationSASL (B)

Byte1('R')

###################

Int32

######################

Int32(10)

SASL################

########SASL########################### ################################## ## #############

String

SASL##########

AuthenticationSASLContinue (B)

Byte1('R')

###################

Int32

######################

Int32(11)

########SASL##################

Byte*n*

####SASL######SASL######

AuthenticationSASLFinal (B)

Byte1('R')

###################

Int32

######################

Int32(12)

SASL################

Byte*n*

SASL#################SASL###########

BackendKeyData (B)

Byte1('K')

############################ ##########CancelRequest######################## ###################

Int32(12)

######################

Int32

#############ID###

Int32

###############

Bind (F)

Byte1('B')

######Bind################

Int32

######################

String

################################

String

#####################################

Int16

#######################*C*####### ##
######0##################################1########### ####################
#######

Int16[*C*]

############# ########0##########1#################

Int16

############ ############# ################################

##############################

Int32

############################## ############# #########-1#NULL###########
NULL####################

Byte*n*

############################ *n*#########

##########################

Int16

############## #####*R*####### ##################0#########################
####################1########### ############################

Int16[*R*]

########## #######0##########1######################

BindComplete (B)

Byte1('2')

######Bind#################

Int32(4)

######################

CancelRequest (F)

Int32(16)

#####################

Int32(80877102)

######### ########16####1234####16####5678############## ##############
###########################

Int32

#############ID###

Int32

##############

Close (F)

Byte1('C')

######Close################

Int32

#####################

Byte1

############'S'# #########'P'###

String

#################### ################################

CloseComplete (B)

Byte1('3')

######Close################

Int32(4)

#####################

CommandComplete (B)

Byte1('C')

#########################

Int32

#####################

String

######### #######SQL####################

INSERT##########INSERT *oid rows*### ###*rows*########## ###*oid*##*rows*#
1##########OID##################ID########OID############## ####*oid*###
0###

DELETE###########DELETE *rows*### ###*rows*###########

UPDATE###########UPDATE *rows*### ###*rows*###########

MERGE###########MERGE *rows*### ###*rows*################

SELECT###CREATE TABLE AS#######SELECT *rows*###### ###*rows*###########

MOVE###########MOVE *rows*### ###*rows*#######################

FETCH###########FETCH *rows*### ###*rows*#######################

COPY###########COPY *rows*### ###*rows*####################PostgreSQL 8.2## ###########

CopyData (F & B)

Byte1('d')

##########COPY###########

Int32

######################

Byte*n*

COPY###################### ######################1############# ######### ##

CopyDone (F & B)

Byte1('c')

######COPY################

Int32(4)

######################

CopyFail (F)

Byte1('f')

######COPY################

Int32

######################

String

####################

CopyInResponse (B)

Byte1('G')

######Start Copy In############### ################################## ######## ########CopyFail################

Int32

######################

Int8

0#COPY## 1##############
#DataRow################# #######COPY##########

Int16

############## #####N########

Int16[N]

################ #######0#########1################# ####################
0###########

CopyOutResponse (B)

Byte1('H')

######Start Copy Out############## ########################

Int32

######################

Int8

0#COPY## 1##############
#DataRow################# #######COPY##########

Int16

############## #####N########

Int16[N]

################ #######0#########1################# ####################
0###########

CopyBothResponse (B)

Byte1('W')

######Start Copy Both############## ################################

Int32

######################

Int8

0#COPY## 1##############
#DataRow################# #######COPY##########

Int16

############## #####N########

Int16[N]

################ #######0#########1################# ####################
0###########

DataRow (B)

Byte1('D')

######################

Int32

######################

Int16

######### #############

##########################

Int32

############ ################ ############ #########-1#NULL########### NULL####################

Byte*n*

######################### *n*#########

Describe (F)

Byte1('D')

######Describe################

Int32

######################

Byte1

#############'S'# ###########'P'###

String

######################### ################################

EmptyQueryResponse (B)

Byte1('I')

################################## ####CommandComplete#########

Int32(4)

######################

ErrorResponse (B)

Byte1('E')

####################

Int32

######################

##############################1#################### ######################### # ###############

Byte1

################## ################################### 53.8################ ################## ## ################

String

#########

Execute (F)

Byte1('E')

######Execute################

Int32

######################

String

############## ####################

Int32

############################## ################## ##############

Flush (F)

Byte1('H')

######Flush################

Int32(4)

######################

FunctionCall (F)

Byte1('F')

########################

Int32

######################

Int32

#############ID#######

Int16

############## #####C####### ## ####0###########################1########### ########################

Int16[C]

########### ######0#########1##################

Int16

##################

##########################

Int32

############## ############### 0########## #########-1#NULL########## ## NULL###################

Byte*n*

######################### *n*#########

######################

Int16

############## ###0#########1#################

FunctionCallResponse (B)

Byte1('V')

##########################

Int32

######################

Int32

################## ################ ############# #########-1#NULL######## ######## NULL###################

Byte*n*

############################ *n*#########

GSSENCRequest (F)

Int32(8)

######################

Int32(80877104)

GSSAPI########## ########16####1234####16####5680############## ###### ##################################

GSSResponse (F)

Byte1('p')

########GSSAPI###SSPI############## ###SASL############################ ###### ########################

Int32

######################

Byte*n*

GSSAPI/SSPI#############

NegotiateProtocolVersion (B)

Byte1('v')

##################################

Int32

######################

Int32

###

Int32

#######################

####################################

String

#######

NoData (B)

Byte1('n')

###########################

Int32(4)

######################

NoticeResponse (B)

Byte1('N')

####################

Int32

######################

#############################1#################### ######################### # ##############

Byte1

################## ################################### 53.8################ ################# ##

String

#########

NotificationResponse (B)

Byte1('A')

######################

Int32

######################

Int32

##############ID###

String

##################

String

#########################

ParameterDescription (B)

Byte1('t')

#########################

Int32

######################

Int16

############### ###############

######################

Int32

#################ID#######

ParameterStatus (B)

Byte1('S')

##############################

Int32

######################

String

###################

String

##############

Parse (F)

Byte1('P')

######Parse################

Int32

######################

String

############# #######################

String

################

Int16

################## ############### ######################################
################################

#######################

Int32

#################ID####### ##########################

ParseComplete (B)

Byte1('1')

######Parse#################

Int32(4)

######################

PasswordMessage (F)

Byte1('p')

######################### ###GSSAPI#SSPI###SASL#######################
########################

Int32

######################

String

####### #################

PortalSuspended (B)

Byte1('s')

########################### ####Execute##################################

Int32(4)

######################

Query (F)

Byte1('Q')

#########################

Int32

######################

String

#############

ReadyForQuery (B)

Byte1('Z')

################ ################################ReadyForQuery########

Int32(5)

######################

Byte1

########################## ################################'I'##############
####'T'##'E'###

RowDescription (B)

Byte1('T')

######################

Int32

######################

Int16

############## ##############

######################

String

#########

Int32

####################################ID### ##########

Int16

################################### ##########

Int32

#################ID###

Int16

########pg_type.typlen####### ########################

Int32

#####pg_attribute.atttypmod####### ##############

Int16

################### ###0#########1################# RowDescription#########
#Describe#########################0######

SASLInitialResponse (F)

Byte1('p')

#########SASL############## ###GSSAPI#SSPI################################ #### ########################

Int32

######################

String

##########SASL########

Int32

#####SASL#######Initial Client Response######################Initial Response### ##-1#

Byte*n*

SASL#######Initial Response#########

SASLResponse (F)

Byte1('p')

######SASL############## ###GSSAPI#SSPI################################ ## ########################

Int32

######################

Byte*n*

SASL##############

SSLRequest (F)

Int32(8)

######################

Int32(80877103)

SSL######## ########16####1234######16####5679############### ########## ####################################

StartupMessage (F)

Int32

######################

Int32(196608)

################ ####16################################3#### ####16###### ##########################0####

################################1######## ############################## ##### ################## user############## ###################

String

##################:

user

################## ###########

database

############# #############

options

#################### ##################################### ##########
##########(\)################################## ##################\\#
########

replication

#############################SQL##################################
##true#false###database##############false### ###53.4##########

########################## _pq_.################################## ###
################################### ####################################
######### ######################

String

##########

Sync (F)

Byte1('S')

######Sync###############

Int32(4)

######################

Terminate (F)

Byte1('X')

####################

Int32(4)

######################

53.8.

#####ErrorResponse###NoticeResponse############################## ################
################ ##################################1####################

S

########ERROR#FATAL#PANIC########
###WARNING#NOTICE#DEBUG#INFO#LOG##################1################ ##
######

V

#########ERROR#FATAL#PANIC######### ###WARNING#NOTICE#DEBUG#INFO#LOG############ ######################### #S########### ###PostgreSQL#####9.6#####################

C

###########SQLSTATE########A##### ######### ########

M

########################### #######################1##### ########

D

#################

H

#################

P

##10#ASCII######### ########## ##1#########################

p

######## ###P## #### ################q######

q

######### ##################### #######PL/pgSQL###########SQL##########

W

########1####1#####################

s

########

t

####################################

c

#####################################

d

####################################

n

##

F

####### #######################

L

####################

R

####### #######################

##
##A########## #### ## ######################### ### ############################

################################## ############################ ################ ##################################

53.9.

#################################### ########################SQL################ ###walsender################ walsender#####53.4#########################WAL####### ####################################

Begin

Byte1('B')

#########################

Int64 (XLogRecPtr)

###########LSN###

Int64 (TimestampTz)

################## ####PostgreSQL######2000-01-01#############

Int32 (TransactionId)

########XID###

Message

Byte1('M')

############################

Int32 (TransactionId)

#########xid########################### ##################2##########

Int8

####0##################################1###

Int64 (XLogRecPtr)

################LSN#

String

####################

Int32

######

Byte*n*

###################

Commit

Byte1('C')

######Commit################

Int8(0)

#############

Int64 (XLogRecPtr)

#####LSN###

Int64 (XLogRecPtr)

##########LSN###

Int64 (TimestampTz)

################## ####PostgreSQL######2000-01-01#############

Origin

Byte1('O')

######Origin################

Int64 (XLogRecPtr)

Origin##########LSN###

String

Origin######

################Origin######################

Relation

Byte1('R')

######Relation################

Int32 (TransactionId)

########xid########################## ##################2##########

Int32 (Oid)

#######OID#

String

#####pg_catalog#########

String

########

Int8

#################pg_class#relreplident#####

Int16

###

##

Int8

###############0#################1###########

String

###

Int32 (Oid)

#######OID#

Int32

#######atttypmod##

Type

Byte1('Y')

######Type################

Int32 (TransactionId)

#########xid########################### ##################2##########

Int32 (Oid)

#####OID#

String

#####pg_catalog#########

String

########

Insert

Byte1('I')

######Insert################

Int32 (TransactionId)

########xid########################## ##################2##########

Int32 (Oid)

Relation#######ID############OID#

Byte1('N')

###TupleData#######################

TupleData

###########TupleData##########

Update

Byte1('U')

######Update################

Int32 (TransactionId)

########xid########################## ##################2##########

Int32 (Oid)

Relation#######ID############OID#

Byte1('K')

#####TupleData#################### ###############UPDATE#REPLICA IDENTITY##################################

Byte1('O')

#####TupleData######################### ###############UPDATE########## REPLICA IDENTITY#FULL##################

TupleData

################TupleData########## ####'O'###'K'###################

Byte1('N')

###TupleData#######################

TupleData

###########TupleData##########

Update######'K'#######'O'##

Delete

Byte1('D')

######Delete################

Int32 (TransactionId)

########xid########################## ##################2##########

Int32 (Oid)

Relation#######ID###########OID#

Byte1('K')

#####TupleData#################### ########DELETE################## REPLICA IDENTITY####################

Byte1('O')

#####TupleData######################## ########DELETE##########REPLICA IDENTITY#FULL###################

TupleData

##############################TupleData##########

Delete######'K'########'O'################################

Truncate

Byte1('T')

######Truncate###########

Int32 (TransactionId)

#########xid########################### ##################2##########

Int32

#######

Int8

TRUNCATE############1#CASCADE#2#RESTART IDENTITY

Int32 (Oid)

############ID############OID# ########################

########Stream Start#Stream Stop#Stream Commit#Stream Abort###########2##########

Stream Start

Byte1('S')

##############################

Int32 (TransactionId)

#########XID###

Int8

##1####XID#####################0#########################

Stream Stop

Byte1('E')

##############################

Stream Commit

Byte1('c')

################################

Int32 (TransactionId)

########XID###

Int8(0)

#############

Int64 (XLogRecPtr)

#####LSN###

Int64 (XLogRecPtr)

##########LSN###

Int64 (TimestampTz)

################## ####PostgreSQL######2000-01-01#############

Stream Abort

Byte1('A')

################################

Int32 (TransactionId)

########XID###

Int32 (TransactionId)

##########xid#################xid##############

Int64 (XLogRecPtr)

LSN###### ##################4##########

Int64 (TimestampTz)

################## ####PostgreSQL######2000-01-01############# ############ ######4##########

########Begin Prepare#Prepare# Commit Prepared#Rollback Prepared#Stream Prepare######### ###3##########

Begin Prepare

Byte1('b')

#######################################

Int64 (XLogRecPtr)

######LSN###

Int64 (XLogRecPtr)

################LSN###

Int64 (TimestampTz)

######################### ####PostgreSQL######2000-01-01#############

Int32 (TransactionId)

########XID###

String

###################GID###

Prepare

Byte1('P')

####################################

Int8(0)

#############

Int64 (XLogRecPtr)

######LSN###

Int64 (XLogRecPtr)

################LSN###

Int64 (TimestampTz)

######################### ####PostgreSQL######2000-01-01#############

Int32 (TransactionId)

########XID###

String

###################GID###

Commit Prepared

Byte1('K')

##

Int8(0)

#############

Int64 (XLogRecPtr)

###################LSN###

Int64 (XLogRecPtr)

#####################LSN###

Int64 (TimestampTz)

################## ####PostgreSQL#####2000-01-01#############

Int32 (TransactionId)

########XID###

String

##################GID###

Rollback Prepared

Byte1('r')

##

Int8(0)

#############

Int64 (XLogRecPtr)

################LSN###

Int64 (XLogRecPtr)

#######################LSN###

Int64 (TimestampTz)

########################## ####PostgreSQL######2000-01-01#############

Int64 (TimestampTz)

################### ####PostgreSQL######2000-01-01############

Int32 (TransactionId)

########XID###

String

##################GID###

Stream Prepare

Byte1('p')

#####################################

Int8(0)

#############

Int64 (XLogRecPtr)

######LSN###

Int64 (XLogRecPtr)

################LSN###

Int64 (TimestampTz)

########################## ####PostgreSQL######2000-01-01#############

Int32 (TransactionId)

########XID###

String

##################GID###

#########################

TupleData

Int16

###

##

Byte1('n')

####NULL#########

Or

Byte1('u')

TOAST##############################

Or

Byte1('t')

#######################

Or

Byte1('b')

##########################

Int32

######

Byte*n*

#################### ####################### *n*#########

53.10. #####2.0##########

#########################3.0##################################

####################################### ### ############# ########options#########################options################ ##

###PasswordMessage

ErrorResponse###NoticeResponse#'E'###'N'########################## ################ ################################## ############################## ########### ######################

ReadyForQuery#'Z'###########################

BinaryRow#DataRow#################### 1##DataRow##############################
DataRow################################ ###################
#######

######################### #####################Parse#Execute#Describe#Close#Flush###
#Sync###############
ParseComplete#BindComplete#PortalSuspended#ParameterDescription#NoData####
CloseComplete######## ##
#######

COPY####CopyData#CopyDone###################### COPY#########################
####\.################COPY OUT######## #COPY IN#######################
################### ####COPY########## CopyInResponse#CopyOutResponse###########
##################

FunctionCall#FunctionCallResponse#################### FunctionCall#####NULL########
########### ## ###################
#############FunctionCall############################

##ParameterStatus#'S'############# #
################################ParameterStatus############

RowDescription#'T'##########################OID############### ################

CursorResponse#'P'########################

NotificationResponse#'A'########NOTIFY###
#####

EmptyQueryResponse#'I'################################## ###########

#54# PostgreSQL########

54.1.

```
###############4################################################## ##############
###1###########

#############BSD######## ##if#while#switch############################

####80##################1############# ####80######################## ########
####################80########################################

#################C++#######//################ pgindent######/* ... */#######

##############################


/*
 * ################
 * ########
 */

###1#############pgindent#######################################################
######## ######################################


    /*----------

     * ################
     * ########
     *----------
     */

############################################## #################pgindent###### #
##################################### ########################################
#######

src/tools/editors################################Emacs#xemacs#vim########
#################

pgindent##################################src/tools/pgindent###############

#########more#less###########


more -x4
less -x4

##################
```

54.2.

```
##############################ereport##########elog############# ###########
############

#####[87.081340]###################DEBUG##PANIC####################2########
### ################ ###############SQL###SQLSTATE############### ereport#
####################C#####################################  ereport########
##################  ###########################ereport#####errmsg#########
```

########### #############2########### ######src/####/utils/elog.h#### ###include############################# ################################## SQLSTATE#####SQL####### ereport##########################C############## ############################## ereport######################### ############ ##################ereport#####errmsg######################

####ereport################

```
ereport(ERROR,
        errcode(ERRCODE_DIVISION_BY_ZERO),
        errmsg("division by zero"));
```

###########ERROR############## errcode#####src/include/utils/errcodes.h#############SQLSTATE############ errmsg####################

```
##############################################
```

```
ereport(ERROR,
        (errcode(ERRCODE_DIVISION_BY_ZERO),
         errmsg("division by zero")));
```

########PostgreSQL#####12######################

```
################
```

```
ereport(ERROR,
        errcode(ERRCODE_AMBIGUOUS_FUNCTION),
        errmsg("function %s is not unique",
               func_signature_string(funcname, nargs,
                                     NIL, actual_arg_types)),
        errhint("Unable to choose a best candidate function. "
                "You might need to add explicit typecasts."));
```

##################errcode#errmsg##########

#######ERROR#######ereport############################ #######ERROR##### ##ereport########

ereport####################

- errcode(sqlerrcode)#######SQLSTATE############## ##################### #################ERROR######ERRCODE_INTERNAL_ERROR########## #WARNING#####ERRCODE_WARNING########NOTICE# ##ERRCODE_SUCCESSFUL_COMPLETION###### ###################errcode()#### #########################
- errmsg(const char *msg, ...)#################################### #### ####sprintf############### sprintf##################%m############# #errno####strerror#######################[1]%m#errmsg################### ###### ####################################gettext##############
- errmsg_internal(const char *msg, ...)################################ #######errmsg###### ####################################
- errmsg_plural(const char *fmt_singular, const char *fmt_plural, unsigned long n, ...)#errmsg#########################*fmt_singular*#####

[1] ###ereport############# ########errno############# errmsg##strerror(errno)#################### ## ##################

####*fmt_plural*########n## #
#######55.2.2#########

- errdetail(const char *msg, ...)#################### ################## ######################### ###########errmsg################

- errdetail_internal(const char *msg, ...)############################ ###########errdetail###### ## ##################

- errdetail_plural(const char *fmt_singular, const char *fmt_plural, unsigned long n, ...)#errdetail############################ ####### #55.2.2#########

- ##############################errdetail_log(const char *msg, ...)#errdetail###### errdetail############errdetail_log####### ################################ ## ##########################

- errdetail_log_plural(const char *fmt_singular, const char *fmt_plural, unsigned long n, ...)#errdetail_log################### ######## #######55.2.2#########

- errhint(const char *msg, ...)###################### ################## ## ###########errmsg################

- errhint_plural(const char *fmt_singular, const char *fmt_plural, unsigned long n, ...)#errhint########################### #######55.2.2# ########

- errcontext(const char *msg, ...)###ereport################## ####### #########PL#################error_context_stack################ ####### #errmsg################ ###########1##ereport################## ######## #######################

- errposition(int cursorpos)############################### ############# ###############################

- errtable(Relation rel)## ###

- errtablecol(Relation rel, int attnum)################################ #######################

- errtableconstraint(Relation rel, const char *conname)############## ## ###########pg_constraint# ##rel####### #######

- errdatatype(Oid datatypeOid)## #######

- errdomainconstraint(Oid datatypeOid, const char *conname)######### ##

- errcode_for_file_access()#########################SQLSTATE########### ########## #####errno######################## ####################%m##### ###############

- errcode_for_socket_access()####################SQLSTATE############ #########

- `errhidestmt(bool hide_stmt)`##postmaster#############STATEMENT:######## ############ ######################################
- `errhidecontext(bool hide_ctx)`##postmaster#############CONTEXT:######## ############ ## #####

> ##
>
> ereport############errtable#errtablecol# errtableconstraint#errdatatype####errdomainconstraint########## ############ ## ##################################### ################################ ###################### PostgreSQL 9.3##################SQLSTATE###23### ############################

###############elog####### ###elog######

```
elog(level, "format string", ...);
```

###########

```
ereport(level, errmsg_internal("format string", ...));
```

SQLSTATE## ######elog########### ################## ##################ereport######## ################### ##########elog############# ##########################

54.3###########################

54.3.

###########PostgreSQL###

###########

######################## ######################## ###############1########### ### ## ## ############################## ## ################

####

```
IpcMemoryCreate: shmget(key=%d, size=%u, 0%o) failed: %m
(plus a long addendum that is basically a hint)
```

####################

```
Primary:    could not create shared memory segment: %m
Detail:     Failed syscall was shmget(key=%d, size=%u, 0%o).
Hint:       The addendum, written as a complete sentence.
```

######################1################### ############### ## ## ############################

##

################################ ## ### #############\n################################ ###################### ########## ########### ##

GUI###################################

###

############################ ##1## #############

SQL######################### ### ################################### ####################### ############

######

#####[84.100418]## ###### ##################################### ###############-######################### ############################## ### ###

############################(###format_type_be())###### ################### ##########

####################################

######

##################/################

########## ################### ######################## ########################### ####

################ ########################### ################### ############### ####2################################

########: ################################## ########################

#####1## ################ ################################## ##################1#######################

#######

######################### ######################## SQL########################### ##########

##

######

############ ##########A#B############################## ###################### ############### #######I################

###########

#######

##########################

###2#######################

```
could not open file "%s": %m
```

###

```
cannot open file "%s"
```

############################## ## #
##

2## ##################
####

########################
############

##########

#####################################

#######foo.bar.baz###############

###

######1######################2######################

##################################

###############

##

```
could not open file %s: %m
```

#######
##

######

########################### ##########

```
BAD:    could not open file %s
BETTER: could not open file %s (I/O failure)
```

######################

###

################################## ##
############# ##

```
BAD:    pg_strtoint32: error in "z": cannot parse "z"
BETTER: invalid input syntax for type integer: "z"
```

###################### ############################

```
BAD:    open() failed: %m
BETTER: could not open file %s: %m
```

################################### ##
######

######################

###########

Unable. #Unable########### #cannot#####could not################

Bad. #bad result################################ ####bad###########invalid format#
#########

Illegal. #Illegal########### ###invalid#### ##############################

Unknown. #unknown############### #error: unknown response############## #########
############################## #Unrecognized#################### ################
##############

```
BAD:    unknown node type
BETTER: unrecognized node type: 42
```

Find#Exists. ##
##########find################## ##
##exist################ ######find######################

May#Can#Might. #May### #Can######
#########################might############################ ####################
#########

##. #can't############## #####cannot###########

Non-negative####. 0###################non-negative################# #0##########0#
#################

######

################### ###############

- spec
- stats
- parens
- auth
- xact

#############

##########

###################################### 55.2.2#################################

54.4.

##C

PostgreSQL#####C99############################# ####C99#####################
############postgres##############

######C99################PostgreSQL###################### ####################
##//################ #######################

##############################C################################

```
####_Static_assert()#__builtin_constant_p########C#############GCC####
#############      ##############################################C99########
##__builtin_constant_p######
```

###################

```
########static inline################## ##################################
###### #############
```

```
#define Max(x, y)       ((x) > (y) ? (x) : (y))
```

```
#####################################  ########################################
############################
```

```
##############################################################################
#############
```

```
#ifndef FRONTEND
static inline MemoryContext
MemoryContextSwitchTo(MemoryContext context)
{
    MemoryContext old = CurrentMemoryContext;

    CurrentMemoryContext = context;
    return old;
}
#endif   /* FRONTEND */
```

```
###################CurrentMemoryContext##############ifndef FRONTEND###
###### ############################################################
```

############

```
##############################################  ##############################
######################## ############################################## ######
#####################################################
```

```
################################POSIX################################
#volatile sig_atomic_t############## postgres###########################
#####SetLatch()###
```

```
################################################################################
################
```

```
static void
handle_sighup(SIGNAL_ARGS)
{
    got_SIGHUP = true;
    SetLatch(MyLatch);
}
```

############

```
#######################################################################
```

```
(*emit_log_hook) (edata);
```

```
#emit_log_hook(edata)########### ########################################
#############
```

```
paramInfo->paramFetch(paramInfo, paramId);
```

#55#

55.1.

PostgreSQL## ########################PostgreSQL################### ########################### ################

55.1.1.

#################### ##################### #################### ########### ##################################### ##############--enable-nls############## ########## ##########################libintl######msgfmt################ ########## #####################

#######################################GNU###########xgettext#msgmerge###2# ################ ########################xgettext################## #Git###### #################### ##GNU Gettext 0.10.36########

########gettext########################## ## #######

55.1.2.

###########2############## 1###PO######### #######-Portable Object-################################## 2####MO###############-Machine Object-############PO## #####MO### ########

################################.po####.mo### ################################ ################################## ############## ####psql.po#psql##PO######### #fr.mo#######MO#######

PO################

```
# comment

msgid "original string"
msgstr "translated string"

msgid "more original"
msgstr "another translated"
"string can be broken up like this"

...
```

msgid#################### ######################### msgstr######################### ########### ########C## ########### ################

```
#################### ######################################## ######################
########### #####PO##########################################

#. automatic comment
#: filename.c:1023
#, flags, flags
```

#.## ##################################### ############# #:################################## ############################ ############################ #,########################### ###2########### ###### ##################################fuzzy####### ###########fuzzy########### fuzzy ################################ ##1######c-format#########printf########### ########### ## #################c-format###############

55.1.3.

################################ ##################################### nls.mk###########################

###.po############################## ######*language*.po ############ ## ##*language*#ISO 639-1#2###########[1]###### ####fr.po######### 1################ ############*language_region*.po######## ####*region*#ISO 3166-1#2###########[2]# ##### ####pt_BR.po################### ####################################

##################################

```
make init-po
```

####*progname*.pot########## #.pot######PO################ T################ #######*language*.po########### ########################po/LINGUAS########### ##################################

```
de fr
```

####################

###########

```
make update-po
```

##############################pot##############PO############ ################ ###########################fuzzy####### ##PO#####.po.new###############

55.1.4. PO#######

PO####################### #################PO###################### Emacs#### ####PO####################

####msgstr##############################fuzzy#################

PO##################### ### ########## ## ##########fuzzy######################## ###### fuzzy########################### ################################

#########################

- ##################################### ##########
- ##printf######################## ############################## ############ ################################ ##############################

[1] https://www.loc.gov/standards/iso639-2/php/English_list.php
[2] https://www.iso.org/iso-3166-country-codes.html

```
msgstr "Die Datei %2$s hat %1$u Zeichen."
```

###########################2############ *digits*$#%#######################
######### ######printf################ ##################################
######

- ### ##################### ############### ### #######PO######
  #################
- ###################### ####cannot open file %s##########################
  ##
  ###### 54.3###########
- ## ############################
  ################################

55.2.

55.2.1.

#####PostgreSQL### ###C##############

######NLS#########

1. ##########################

   ```
   #ifdef ENABLE_NLS
   #include <locale.h>
   #endif

   ...

   #ifdef ENABLE_NLS
   setlocale(LC_ALL, "");
   bindtextdomain("progname", LOCALEDIR);
   textdomain("progname");
   #endif
   ```

   #*progname*###############

2. ###########################gettext()################## ####

   ```
   fprintf(stderr, "panic level %d\n", lvl);
   ```

   ##############

   ```
   fprintf(stderr, gettext("panic level %d\n"), lvl);
   ```

   #NLS################gettext#####################

   ################################

   ```
   #define _(x) gettext(x)
   ```

   ##################ereport()####################1##################### ###
   ####################gettext################

3. ##################nls.mk######### ###makefile######## ################### #####

   CATALOG_NAME

   textdomain()###################

   GETTEXT_FILES

   ################### #######gettext################### ############# ###################### ###################file##+###2######1#####1###### ###################

   GETTEXT_TRIGGERS

   ## ###### ##gettext()############ _############################ ############### ###############2########func:2############### ###################### ##########################func:1,2########

4. #################po/LINGUAS#############

#######################################

55.2.2.

#################################

- ############################

  printf("Files were %s.\n", flag ? "copied" : "removed");

  ########################### ###########gettext()########################## ############## ### ####### ##################################

- ##################

  printf("copied %d file%s", n, n!=1 ? "s" : "");

  ################################ #####################

  if (n==1)
  printf("copied 1 file");
  else
  printf("copied %d files", n):

  ########### ###############2################# ############################## #############

  printf("number of copied files: %d", n);

  ## ereport()############ ##############################

  errmsg_plural("copied %d file",
  "copied %d files",
  n,

```
                n)
```

##n## ######################n#######

errmsg####errdetail##################################ngettext########### ## gettext################

- ##translator#################### ######### ##########

```
/* translator: This message is not what it seems to be. */
```

#####################################

#56#

###########Version-1## ######
##################SQL############### ######################################
################ ##

#######################C##############Version-1################
#language_handler#######PostgreSQL############# ##########################
##SQL########################## C##################36.10##########

########################## ####################FunctionCallInfoBaseData
struct##########Datum#############SQL#NULL#############
##FunctionCallInfoBaseData####isnull########################## ########
################FunctionCallInfoBaseData####flinfo->fn_oid#############
##############OID########### ## #
##

pg_proc### ##
#CREATE FUNCTION#####AS###pg_proc###prosrc###### ######################
###

1##SQL######################### #########flinfo->fn_extra##############
############################ ########NULL################################
######## ##########flinfo->fn_extra##NULL##############################
########flinfo->fn_extra##
FmgrInfo######################## #####1#####flinfo->fn_mcxt###############
################# #######FmgrInfo########## ##############################
############### ####################################

#####################################FunctionCallInfoBaseData#context####
###############NULL######TriggerData########## ###########################
###############

C######################src/test/modules/plsample######### ##############
################1#################

###2#####
#############CREATE FUNCTION################
##################DO################################

#######################oid############################ ################ #
#####void############ ##########################CREATE FUNCTION#
########## ####OID####pg_proc##OID### ##############################
########### #####CREATE FUNCTION##############################
##CheckFunctionValidatorAccess()####### ##############################
##############################
##################ereport()######################## ########################
###########################

##########check_function_bodies################## ##############################
###########
###pg_dump########### (#######
##################################
########CREATE FUNCTION##############################) ###################
###################check_function_bodies#######CREATE FUNCTION########
###########SET################### #########GUC##############################
########check_function_bodies######################

##############################internal############################## ########
############ ######void############ ##################DO################ #####
########InlineCodeBlock########## ####DO##################################
############ ##############################

###CREATE EXTENSION#####################################CREATE LANGUAGE##
##################### ############36.17##########

#######src/pl################
CREATE LANGUAGE####################

#57#

###############PostgreSQL################### ################################ ######

#######contrib################ CREATE FOREIGN DATA WRAPPER###################

> ##
>
> ##SQL############################## #######PostgreSQL################# API##################PostgreSQL###API##########

57.1.

FDW################################## ######version-1############C########### ############ C####################36.10##########

FDW###### ##################################### ######################fdw_handler##### ###PostgreSQL############## ############C######SQL############################ #57.2######

###CREATE#ALTER########## ###################### ##################text##########################OID##### ######################### ##########################OID##################

- AttributeRelationId
- ForeignDataWrapperRelationId
- ForeignServerRelationId
- ForeignTableRelationId
- UserMappingRelationId

##

57.2.

FDW################################palloc###FdwRoutine######## ############## ##############

FdwRoutine####src/include/foreign/fdwapi.h############################

57.2.1. ###############FDW####

```
void
GetForeignRelSize(PlannerInfo *root,
                  RelOptInfo *baserel,
                  Oid foreigntableid);
```

########################## ## root### ######################## baserel################### ####foreigntableid##### #####pg_class OID### (foreigntableid#####################################)

##baserel->rows#########
baserel->rows###
###baserel->width############## (###############ANALYZE#################
##) ####################################baserel->tuples########### (###
#pg_class.reltuples#########ANALYZE############## ###########ANALYZE#####
########-1######)

#########57.4##########

```
void
GetForeignPaths(PlannerInfo *root,
                RelOptInfo *baserel,
                Oid foreigntableid);
```

################################ ######################### ###########
#GetForeignRelSize######

###################(ForeignPath###)#############baserel->pathlist######
#add_path############ ForeignPath#########create_foreignscan_path#####
######### #########################pathkeys############################### ###
##FDW################

#########57.4##########

```
ForeignScan *
GetForeignPlan(PlannerInfo *root,
               RelOptInfo *baserel,
               Oid foreigntableid,
               ForeignPath *best_path,
               List *tlist,
               List *scan_clauses,
               Plan *outer_plan);
```

##############ForeignScan############ ########################## ##
##GetForeignRelSize###########ForeignPath(##
#GetForeignPaths#GetForeignJoinPaths###GetForeignUpperPaths#########
#)##RecheckForeignScan########
#####ForeignScan################ ##########################
##foreigntableid#InvalidOid######

#####ForeignScan######################ForeignScan########
#make_foreignscan##############

#########57.4##########

```
void
BeginForeignScan(ForeignScanState *node,
                 int eflags);
```

################### ######################## ################################
##########################(######IterateForeignScan###############)#
ForeignScanState###############fdw_state######NULL##### ############
##ForeignScanState###(########GetForeignPlan######FDW#########
##ForeignScan######)############ eflags#####################################
####

(eflags & EXEC_FLAG_EXPLAIN_ONLY)#####################################
ExplainForeignScan#EndForeignScan##################################

```
TupleTableSlot *
IterateForeignScan(ForeignScanState *node);
```

####################################(#########ScanTupleSlot######)# ###### ######NULL###### ###################################### ##################### ########### ## ################ ##BeginForeignScan####################EState#####es_query_cxt#########

##############fdw_scan_tlist## ################### ##################################NULL################# #fdw_scan_tlist############

PostgreSQL## ##################### ## ##########################(######### #################)################

```
void
ReScanForeignScan(ForeignScanState *node);
```

############### ## ###########

```
void
EndForeignScan(ForeignScanState *node);
```

################### ###palloc## #############

57.2.2. ####################FDW####

FDW###
#######

```
void
GetForeignJoinPaths(PlannerInfo *root,
                    RelOptInfo *joinrel,
                    RelOptInfo *outerrel,
                    RelOptInfo *innerrel,
                    JoinType jointype,
                    JoinPathExtraData *extra);
```

##########2#################################### ############################ GetForeignPaths##############joinrel####ForeignPath########### #create_foreign_join_path#######add_path################################ ## ####GetForeignPaths###############1###################### ############# #################

###FDW######

###########extra->restrictlist## ######## joinrel########ForeignPath############################ ## ##joinrel##########################ForeignPath######################## ###

ForeignPath## ####### ## 1###############ForeignScan# #####scanrelid#0## ##### ##ForeignScan####fs_relids####################### ################### #################FDW################## ##################################

##FDW#fdw_scan_tlist####TargetEntry#####################################
############

##
PostgreSQL 16###fs_relids#################################### ####### #fs_base_relids##################fs_relids##################

#########57.4##########

57.2.3. ####/##################FDW####

FDW###################/################################

```
void
GetForeignUpperPaths(PlannerInfo *root,
                     UpperRelationKind stage,
                     RelOptInfo *input_rel,
                     RelOptInfo *output_rel,
                     void *extra);
```

##/### ########### ############################ ################################FDW ############# #######FDW###########################/#######ForeignPath##### #######create_foreign_upper_path##############################
#add_path########## GetForeignJoinPaths################################## ##############################

stage###########/###################### output_rel######################### ########input_rel#################### extra################### #UPPERREL_PARTIAL_GROUP_AGG#UPPERREL_GROUP_AGG############# #GroupPathExtraData########### ###UPPERREL_FINAL############## #FinalPathExtraData########### ##########################output_rel##### #ForeignPath#######input_rel#################### ######################## ##################################

#########57.4##########

57.2.4. #############FDW####

##FDW#####################FDW##

```
void
AddForeignUpdateTargets(PlannerInfo *root,
                        Index rtindex,
                        RangeTblEntry *target_rte,
                        Relation target_relation);
```

UPDATE#DELETE## FDW################## #####ID############################ #################UPDATE#DELETE######### ##################(##########)##################

#################Var###################add_row_identity_var############## ################## #########Var#########################Var#varno######## ############################## ############tableoid#######tableoid##ctid# ##ctid*N*#ctid#####vartype = RECORD########Var#wholerow##vartype#######

#Var#wholerowN######################## ##################################### ################# ########################FDW############################### ##########

##AddForeignUpdateTargets#####NULL########################### (FDW##### ###################UPDATE##################DELETE####################)

```
List *
PlanForeignModify(PlannerInfo *root,
                  ModifyTable *plan,
                  Index resultRelation,
                  int subplan_index);
```

###############ModifyTable###### #####FDW##################List##################BeginForeignModify#####

root######################### plan#fdwPrivLists############# #ModifyTable######## resultRelation########################## subplan_index#ModifyTable################0######### ####plan########### ################################

#####57.4#########

##PlanForeignModify#####NULL########################## ##BeginForeignModify#####fdw_private####NIL######

```
void
BeginForeignModify(ModifyTableState *mtstate,
                   ResultRelInfo *rinfo,
                   List *fdw_private,
                   int subplan_index,
                   int eflags);
```

##################### ######################### ############################## ######### ######################ExecForeignInsert/ ExecForeignBatchInsert#ExecForeignUpdate#ExecForeignDelete######### ##

mtstate#######ModifyTable## rinfo############ResultRelInfo###### (ResultRelInfo#ri_FdwState######## ############FDW#############) fdw_private######PlanForeignModify###### ######### subplan_index#####ModifyTable######################## eflags### ####################################

(eflags & EXEC_FLAG_EXPLAIN_ONLY)#################################### ExplainForeignModify#EndForeignModify################################

##BeginForeignModify#####NULL#################################

```
TupleTableSlot *
ExecForeignInsert(EState *estate,
                  ResultRelInfo *rinfo,
                  TupleTableSlot *slot,
                  TupleTableSlot *planSlot);
```

################## estate##################### rinfo########### #ResultRelInfo##### slot################################## planSlot# #ModifyTable##slot######## (planSlot#####INSERT#################################)

##############(####################################)###########(############
##)##############NULL### ####slot#############

##############INSERT##RETURNING########WITH CHECK OPTION################
##########AFTER ROW################## ##############FDW#RETURNING##WITH
CHECK OPTION## ####################
###

##ExecForeignInsert#####NULL##

#####################################COPY FROM##############################
COPY FROM####INSERT################### FDW##########################
############


```
TupleTableSlot **
ExecForeignBatchInsert(EState *estate,
                       ResultRelInfo *rinfo,
                       TupleTableSlot **slots,
                       TupleTableSlot **planSlots,
                       int *numSlots);
```

###################### slots#planSlots############*numSlots##############
#######ExecForeignInsert###########

###
#slots################ #############*numSlots######

##############INSERT#####WITH CHECK OPTION###############AFTER ROW######
############# ##################FDW#WITH CHECK OPTION####################
##########################

ExecForeignBatchInsert###GetForeignModifyBatchSize#####NULL##########
######ExecForeignInsert########### #####RETURNING####INSERT###########

#####################################COPY FROM##############################
COPY FROM####INSERT################### FDW##########################
############


```
int
GetForeignModifyBatchSize(ResultRelInfo *rinfo);
```

##################ExecForeignBatchInsert###################### ##########
############ExecForeignBatchInsert#####   rinfo###############
#ResultRelInfo######### FDW##
##############

ExecForeignBatchInsert###GetForeignModifyBatchSize#####NULL##########
######ExecForeignInsert###########


```
TupleTableSlot *
ExecForeignUpdate(EState *estate,
                  ResultRelInfo *rinfo,
                  TupleTableSlot *slot,
                  TupleTableSlot *planSlot);
```

##################   estate####################   rinfo###########
#ResultRelInfo##### slot################################# planSlot#
#ModifyTable######################### slot##############################
######planSlot#################################### ###planSlot##########
########### ###AddForeignUpdateTargets##################################

##############(####################################)###########(#############
##)##############NULL### ####slot#############

##############UPDATE##RETURNING########WITH CHECK OPTION###############
##########AFTER ROW################## ##############FDW#RETURNING##WITH
CHECK OPTION## ######################
##

##ExecForeignUpdate#####NULL###


```
TupleTableSlot *
ExecForeignDelete(EState *estate,
                  ResultRelInfo *rinfo,
                  TupleTableSlot *slot,
                  TupleTableSlot *planSlot);
```

##################   estate####################   rinfo###########
#ResultRelInfo##### slot################################### planSlot#
#ModifyTable##########################AddForeignUpdateTargets#######
##

#########################(##########)##############NULL### ####slot#######
##############

##############DELETE######RETURNING##################AFTER ROW##########
######## ##############FDW#RETURNING#####################################
###

##ExecForeignDelete#####NULL###


```
void
EndForeignModify(EState *estate,
                 ResultRelInfo *rinfo);
```

################### ###palloc##
#############

##EndForeignModify#####NULL#####################################

INSERT####COPY FROM################################### FDW###############
################################## ##############COPY FROM###############


```
void
BeginForeignInsert(ModifyTableState *mtstate,
                   ResultRelInfo *rinfo);
```

#################### ################################COPY   FROM#######
##################################### ################################ ##
##ExecForeignInsert###ExecForeignBatchInsert#########################

mtstate######ModifyTable###############   ################################
rinfo#ResultRelInfo######################   #########FDW###############
#ResultRelInfo#ri_FdwState#############

#####COPY   FROM###
##mtstate##
#ExecForeignInsert#planSlot######NULL######

BeginForeignInsert#####NULL################

FDW##COPY FROM##################
##################ExecForeignInsert/ExecForeignBatchInsert###########
#########

```
void
EndForeignInsert(EState *estate,
                 ResultRelInfo *rinfo);
```

################### ###palloc###
############

EndForeignInsert#####NULL###############

```
int
IsForeignRelUpdatable(Relation rel);
```

################################## ##
UPDATE##(1 << CMD_UPDATE) = 4#INSERT##(1 << CMD_INSERT) =
8#DELETE##(1 << CMD_DELETE) = 16####CmdType########

##IsForeignRelUpdatable#####NULL###############
#ExecForeignInsert#ExecForeignUpdate#ExecForeignDelete###############
################### ######FDW################################ (#############
##information_schema############
############)

################################
################### ################################ ########################
###WITH CHECK OPTION#
######################

```
bool
PlanDirectModify(PlannerInfo *root,
                 ModifyTable *plan,
                 Index resultRelation,
                 int subplan_index);
```

############################## ############################true##### ####
##false#####   ###########################   ########
##BeginDirectModify#IterateDirectModify#EndDirectModify##############
######PlanForeignModify###########

##ForeignScan##################
ForeignScan#operation#resultRelation##################### operation#
########CmdType#############(####UPDATE##CMD_UPDATE#INSERT#
#CMD_INSERT#DELETE##CMD_DELETE)#####resultRelation###resultRelation###
################

#####57.4#########

PlanDirectModify#####NULL########## ########################

```
void
BeginDirectModify(ForeignScanState *node,
                  int eflags);
```

########################   ###########################   ########
#IterateDirectModify##

ForeignScanState###############fdw_state########NULL### ##############
#ForeignScanState############PlanDirectModify#####FDW##############
#ForeignScan############### eflags#######################################

(eflags & EXEC_FLAG_EXPLAIN_ONLY)#####################################
ExplainDirectModify#EndDirectModify##################################

BeginDirectModify#####NULL##################################

```
TupleTableSlot *
IterateDirectModify(ForeignScanState *node);
```

INSERT#UPDATE#####DELETE######RETURNING###########################NULL
######RETURNING########RETURNING#####################################
##############ScanTupleSlot######## ################node-
>resultRelInfo->ri_projectReturning->pi_exprContext-
>ecxt_scantuple############## ############NULL###### ####################
########################## ################BeginDirectModify############
######EState#es_query_cxt########

##############fdw_scan_tlist##
################# RETURNING#####################################NULL######
###########fdw_scan_tlist###########

##############################FDW################### ###############FDW
#EXPLAIN ANALYZE####ForeignScanState node##################

IterateDirectModify#####NULL##################################

```
void
EndDirectModify(ForeignScanState *node);
```

########################## ###palloc##
###################

EndDirectModify#####NULL##################################

57.2.5. TRUNCATE####FDW####

```
void
ExecForeignTruncate(List *rels,
                    DropBehavior behavior,
                    bool restart_seqs);
```

############# #####TRUNCATE########################## rels#############
#Relation###########

behavior#DROP_RESTRICT#DROP_CASCADE###########TRUNCATE#####RESTRICT#
#CASCADE###################

restart_seqs#true ######TRUNCATE#####RESTART IDENTITY##############
#CONTINUE IDENTITY################

##TRUNCATE#########ONLY######ExecForeignTruncate################ #######
#######SELECT#UPDATE#DELETE##############

ExecForeignTruncate################################ ####rels#############
################################

ExecForeignTruncate#####NULL##

57.2.6. ########FDW####

FDW##57.5##

```
RowMarkType
GetForeignRowMarkType(RangeTblEntry *rte,
                      LockClauseStrength strength);
```

############################## rte######RangeTblEntry#####strength#####FOR UPDATE/SHARE######################## ######RowMarkType#################

#####UPDATE#DELETE#SELECT FOR UPDATE/SHARE###########UPDATE####DELETE# ##################################

GetForeignRowMarkType######NULL###########ROW_MARK_COPY########### ### #RefetchForeignRow#######################################

###########57.5##########

```
void
RefetchForeignRow(EState *estate,
                  ExecRowMark *erm,
                  Datum rowid,
                  TupleTableSlot *slot,
                  bool *updated);
```

###################1################## estate################### erm######## ########################ExecRowMark###### rowid################### slot#### ################################### updated###########

############erm->markType##############GetForeignRowMarkType########## #ROW_MARK_REFERENCE##################################ROW_MARK_COPY######

####*updated##true######## ###### #FDW##########true#############

################erm->waitPolicy#SKIP LOCKED##################

rowid#################ctid#### rowid##Datum############tid######## ####ID# #####################API#############

RefetchForeignRow#####NULL##################################

###########57.5##########

```
bool
RecheckForeignScan(ForeignScanState *node,
                   TupleTableSlot *slot);
```

#######################NULL#########fdw_recheck_quals################ ## ## ###### ######################################NULL################

RecheckForeignScan### ###########
#########################

##
#ForeignScan############# ## #
###########1############################ ############################# #
#GetExistingLocalJoinPath##
######### GetExistingLocalJoinPath#####################################
###################NULL##
##############################

57.2.7. EXPLAIN####FDW####

```
void
ExplainForeignScan(ForeignScanState *node,
                   ExplainState *es);
```

#############EXPLAIN######## EXPLAIN###############ExplainPropertyText#
################## es################################EXPLAIN ANALYZE######
##############ForeignScanState################

##ExplainForeignScan#####NULL###########EXPLAIN##############

```
void
ExplainForeignModify(ModifyTableState *mtstate,
                     ResultRelInfo *rinfo,
                     List *fdw_private,
                     int subplan_index,
                     struct ExplainState *es);
```

###########EXPLAIN######## EXPLAIN###############ExplainPropertyText##
################# es################################EXPLAIN ANALYZE#
###################ModifyTableState################ ###4####
#BeginForeignModify######

##ExplainForeignModify#####NULL###########EXPLAIN##############

```
void
ExplainDirectModify(ForeignScanState *node,
                    ExplainState *es);
```

##################EXPLAIN######## #####EXPLAIN#############
#ExplainPropertyText################ es##############################
##EXPLAIN ANALYZE###################ForeignScanState################

ExplainDirectModify#####NULL###########EXPLAIN##############

57.2.8. ANALYZE####FDW####

```
bool
AnalyzeForeignTable(Relation relation,
                    AcquireSampleRowsFunc *func,
                    BlockNumber *totalpages);
```

#####ANALYZE######################### ##FDW##############################
#######################################*func#totalpages*###true########## ##
#######false######

##FDW##############################AnalyzeForeignTable#####NULL########
##

##

```
int
AcquireSampleRowsFunc(Relation relation,
                      int elevel,
                      HeapTuple *rows,
                      int targrows,
                      double *totalrows,
                      double *totaldeadrows);
```

##targrows##############################*rows*############## ################
####################################*totalrows#totaldeadrows*########
##(##FDW################*totaldeadrows*#0#########)

57.2.9. IMPORT FOREIGN SCHEMA####FDW####

```
List *
ImportForeignSchema(ImportForeignSchemaStmt *stmt, Oid serverOid);
```

###################### #####IMPORT FOREIGN SCHEMA#######################
#####OID######## C#################CREATE FOREIGN TABLE################# ##
######################

ImportForeignSchemaStmt#######remote_schema########################
list_type################### ####FDW_IMPORT_SCHEMA_ALL#############
###################table_list#######
FDW_IMPORT_SCHEMA_LIMIT_TO#table_list################## ##
#FDW_IMPORT_SCHEMA_EXCEPT#table_list#################### options#####
###################### ########FDW####### ####FDW###NOT NULL############
#################### #########FDW##
##

FDW#ImportForeignSchemaStmt#local_schema############### ##############
#CREATE FOREIGN TABLE#####################

FDW#list_type###table_list########################### ###################
##################################
############# ##IsImportableForeignTable()##############################
#####

FDW#######################ImportForeignSchema#####NULL#############

57.2.10. ##########FDW####

ForeignScan########################### ##ForeignScan####################
########################### ## ###
##################
########################

```
bool
IsForeignScanParallelSafe(PlannerInfo *root, RelOptInfo *rel,
                          RangeTblEntry *rte);
```

########################## ##
########################### ###
###

#########

```
Size
EstimateDSMForeignScan(ForeignScanState *node, ParallelContext *pcxt);
```

############################ ################################## ########### ## ############################## ###########FDW##########################3##### ############

```
void
InitializeDSMForeignScan(ForeignScanState *node, ParallelContext *pcxt,
                         void *coordinate);
```

######################### coordinate#EstimateDSMForeignScan############# ############# ################################

```
void
ReInitializeDSMForeignScan(ForeignScanState *node, ParallelContext *pcxt,
                           void *coordinate);
```

ReScanForeignScan### #### ####ReScanForeignScan#################################

```
void
InitializeWorkerForeignScan(ForeignScanState *node, shm_toc *toc,
                            void *coordinate);
```

InitializeDSMForeignScan### ###### ##########################

```
void
ShutdownForeignScan(ForeignScanState *node);
```

#################################### ######################## EndForeignScan############################## ############################## ##DSM(#######)##############DSM## ########

57.2.11. #########FDW####

ForeignScan############src/backend/executor/README##################### ######### ######################################

```
bool
IsForeignPathAsyncCapable(ForeignPath *path);
```

#####ForeignPath## ########### #AppendPath## ################################

#######################IterateForeignScan############################ (#### ##)

```
void
ForeignAsyncRequest(AsyncRequest *areq);
```

ForeignScan########1########### areq#ForeignScan################ #Append#######AsyncRequest##### areq->result#################areq->request_complete#true###### #########I/O################################ ##########false##areq->callback_pending#true#####ForeignScan######### ###################### ##################NULL#############areq->request_complete####true###### ExecAsyncRequestDone#ExecAsyncRequestPending####areq################## ######

```
void
ForeignAsyncConfigureWait(AsyncRequest *areq);
```

ForeignScan######################### #####ForeignScan####areq->callback_pending####################areq#########Append#as_eventset## ################ ###src/backend/executor/execAsync.c#ExecAsyncConfigureWait################# ################### ##ForeignAsyncNotify######

```
void
ForeignAsyncNotify(AsyncRequest *areq);
```

##############ForeignScan########1########### #####ForeignAsyncRequest## ####areq###################

57.2.12. ################FDW####

```
List *
ReparameterizeForeignPathByChild(PlannerInfo *root, List *fdw_private,
                                 RelOptInfo *child_rel);
```

######child_rel## ######### ####################ForeignPath#fdw_private#################### ######## ############## ##reparameterize_path_by_child#adjust_appendrel_attrs### #adjust_appendrel_attrs_multilevel############

57.3.

FDW#########FDW## ######## ####################foreign/foreign.h########################### # ###########################

```
ForeignDataWrapper *
GetForeignDataWrapperExtended(Oid fdwid, bits16 flags);
```

###########OID#############ForeignDataWrapper########### ForeignDataWrapper#######FDW#######foreign/foreign.h########## flags ###################or############ ####FDW_MISSING_OK###################### ################NULL########

```
ForeignDataWrapper *
GetForeignDataWrapper(Oid fdwid);
```

##########OID############ForeignDataWrapper#########
##ForeignDataWrapper#######FDW###########(###foreign/foreign.h########)#

```
ForeignServer *
GetForeignServerExtended(Oid serverid, bits16 flags);
```

###########OID#########ForeignServer########### ForeignServer###########
#######foreign/foreign.h########### flags###################or############
####FSV_MISSING_OK#######################################NULL########

```
ForeignServer *
GetForeignServer(Oid serverid);
```

##########OID########ForeignServer###########ForeignServer#############
#########(###foreign/foreign.h########)#

```
UserMapping *
GetUserMapping(Oid userid, Oid serverid);
```

################################UserMapping###########(##################
##PUBLIC##############################) UserMapping#######################
#(###foreign/foreign.h########)#

```
ForeignTable *
GetForeignTable(Oid relid);
```

##########OID#########ForeignTable###########ForeignTable###############
########(###foreign/foreign.h########)#

```
List *
GetForeignColumnOptions(Oid relid, AttrNumber attnum);
```

################OID################FDW######DefElem#########################
###NIL######

##################OID#########################

```
ForeignDataWrapper *
GetForeignDataWrapperByName(const char *name, bool missing_ok);
```

######################ForeignDataWrapper############################
##missing_ok#true####NULL######################

```
ForeignServer *
GetForeignServerByName(const char *name, bool missing_ok);
```

##################ForeignServer###########################missing_ok#true####
NULL######################

57.4.

FDW########
#GetForeignRelSize#GetForeignPaths#GetForeignPlan#PlanForeignModify#GetForeignJo
PostgreSQL###

root#baserel###########################(########)############## baserel-
>baserestrictinfo####################(WHERE#)###################(#########

###################FDW##############################) baserel->reltarget->exprs#######################################ForeignScan#################### ##

###############FDW######################### #######################FDW####### #########palloc########

baserel->fdw_private##void#####FDW#################################### #######RelOptInfo###########NULL############################ ###### ##GetForeignRelSize##GetForeignPaths#GetForeignPaths# #GetForeignPlan#####################################

GetForeignPaths###ForeignPath####fdw_private########################### ######fdw_private#List## ##### ############################nodeToString####################

GetForeignPlan#######ForeignPath####fdw_private#########ForeignScan## ####################fdw_exprs#fdw_private################## ###### #copyObject################## fdw_private################################ ####### fdw_exprs####NIL## #### ##################################

GetForeignPlan##################################### ####scan_clauses### #baserel->baserestrictinfo################################## FDW##### #RestrictInfo####scan_clauses#####(extract_actual_clauses####)######### ## #####FDW######## ## ####

###############*sub_expression*####################FDW#######
##*foreign_variable* = *sub_expression*##########FDW############ ########## ##################GetForeignPaths######### #########fdw_private######## ####RestrictInfo################ ####GetForeignPlan#####scan_clauses### ##########################*sub_expression*#fdw_exprs######### ############## ##########################fdw_private################## ##################### #fdw_exprs######################WHERE *foreign_variable* = $1########### ##

READ COMMITTED## #fdw_recheck_quals#######RecheckForeignScan############## ############## ## #### fdw_recheck_quals#####RecheckForeignScan##################### ### ############################## ##################################NULL####### ####

FDW########ForeignScan#####fdw_scan_tlist##### ##############FDW## ############# ######################NIL#################################### ########## NIL#####Var#################################TargetEntry##### ########## #######FDW## ###FDW ##fdw_scan_tlist############ ##### #GetForeignJoinPaths###########################fdw_scan_tlist########### #############

FDW###*foreign_variable* = *local_variable*#############(#)############ ######baserel->baserestrictinfo########################## ############################## ### ####################################param_info##################;##### #####get_baserel_parampathinfo###### GetForeignPlan###### #*local_variable*###fdw_exprs##########################

FDW##################GetForeignPaths#######################
##GetForeignJoinPaths###############ForeignPath############ ###########
#########GetForeignPlan########## ####baserestrictinfo##################
#############JOIN##GetForeignJoinPaths#######(extra->restrictlist)#####
####

FDW### ##########
##FDW################################## #################add_path###
#UPPERREL_GROUP_AGG################ ######################################
#########
###################GetForeignPlan################### ####################
#FDW##/##################
#GetForeignUpperPaths############

PlanForeignModify#57.2.4###
#ModifyTable############################## ##################################
######### ##########################FDW##############UPPERREL_FINAL#####
###################ModifyTable########### ######57.2.6####################
######SELECT FOR UPDATE############# UPPERREL_FINAL####################
###################

UPDATE#DELETE############# PlanForeignModify#PlanDirectModify#########
##########baserel->fdw_private####################RelOptInfo##########
######INSERT#####################RelOptInfo#######
PlanForeignModify######List###ForeignScan######fdw_private######
##copyObject####################################

ON CONFLICT####INSERT################## ##################################
######## #######ON CONFLICT DO UPDATE################ ####################

57.5.

FDW###PostgreSQL##################
#################FDW################# ###################

##################################
##############################
################# ##
##############################
#####PostgreSQL#TID##

#######PostgreSQL#FDW#####################FDW############################
######### PostgreSQL#####9.5######57.2.6######API############FDW##############
##

#########READ COMMITTED#########PostgreSQL############################
######################### ##
PostgreSQL#######################################TID######################
########## ##TID######
###
##FDW### ############
######FDW####################

#########UPDATE#DELETE###########ForeignScan###############SELECT FOR
UPDATE######################## FDW######UPDATE###DELETE##############relid
#root->parse->resultRelation###############
#ExecRelationIsTargetRelation()############ ##########
##ExecForeignUpdate###ExecForeignDelete################################
###############

SELECT FOR UPDATE/SHARE##############################ForeignScan#### #SELECT FOR UPDATE/SHARE## ###### ########57.2.6###################### GetForeignRowMarkType################ ##rowmark#####
#ROW_MARK_EXCLUSIVE#ROW_MARK_NOKEYEXCLUSIVE#ROW_MARK_SHARE##
#ROW_MARK_KEYSHARE######## ##########4############################## #### ######################################get_plan_rowmark#############
#ExecFindRowMark############ #####NULL###rowmark##################
#strength######LCS_NONE#################

####UPDATE#DELETE###SELECT FOR UPDATE/SHARE##############################
###########LCS_NONE########GetForeignRowMarkType#####
#ROW_MARK_REFERENCE## ######markType#
######RefetchForeignRow############## ###########################
#GetForeignRowMarkType##############################LCS_NONE#######
#ROW_MARK_COPY#########

#######src/include/nodes/lockoptions.h#src/include/nodes/
plannodes.h##RowMarkType#PlanRowMark#########src/include/nodes/
execnodes.h##ExecRowMark################

#58#

PostgreSQL###TABLESAMPLE#####SQL#####BERNOULLI#SYSTEM################## ################### ###########TABLESAMPLE#######################

SQL#####################################C########SQL##########

```
method_name(internal) RETURNS tsm_handler
```

#####TABLESAMPLE############### internal######(##0#####)########SQL###### ################### #######palloc###TsmRoutine############### ############ ############################ ##########C####SQL###################### ###### #58.1#########

###########TsmRoutine###########################

```
List *parameterTypes
```

> ###################TABLESAMPLE##############OID######OID###### ###### ####################################FLOAT4OID################### ###### ##############################

```
bool repeatable_across_queries
```

> true############REPEATABLE## ########################### false####################REPEATABLE######## ##

```
bool repeatable_across_scans
```

> true###(########### ####################)# false####################2######################### ### ##########################

TsmRoutine####src/include/access/tsmapi.h######### #############

#####src/backend/access/tablesample############### ###########contrib##

58.1.

TSM##########################palloc###TsmRoutine######## ################# #######################NULL######

```
void
SampleScanGetSampleSize (PlannerInfo *root,
                         RelOptInfo *baserel,
                         List *paramexprs,
                         BlockNumber *pages,
                         double *tuples);
```

#################### ### #### (##################baserel->pages#baserel->tuples################# ##############) paramexprs#####TABLESAMPLE############### ############## ##estimate_expression_value()############################ #############

###(#######################
##################)# pages#tuples##########

```
void
InitSampleScan (SampleScanState *node,
                int eflags);
```

SampleScan#################### ######################## ######################
SampleScanState###########tsm_state#####NULL### InitSampleScan#
##########################palloc##node->tsm_state###### ################
#SampleScanState##################### (####node->ss.ss_currentScanDesc#
#####################)# eflags##

(eflags & EXEC_FLAG_EXPLAIN_ONLY)######################
#EXPLAIN#EndSampleScan####################################

#####(#####NULL########)############BeginSampleScan#######################
###########

```
void
BeginSampleScan (SampleScanState *node,
                 Datum *params,
                 int nparams,
                 uint32 seed);
```

################# ######################## ########################
################SampleScanState#################### (####node-
>ss.ss_currentScanDesc#####################)# nparams#####params##
##TABLESAMPLE################## ###############parameterTypes#########
######NULL############## seed##################################### ##
##REPEATABLE##############################random()######

#####node->use_bulkread#node->use_pagemode################ node-
>use_bulkread#true##(#########)#################################### ######
#######################false############## node->use_pagemode#true##(####
######)## ############################
#########false ############## ############################## ################
########################

###########repeatable_across_scans#######################################
######## #######BeginSampleScan######################## (##TABLESAMPLE######
##############)

```
BlockNumber
NextSampleBlock (SampleScanState *node, BlockNumber nblocks);
```

######################## ##################InvalidBlockNumber######

#####(#####NULL########)####### ########################### ##################
###

```
OffsetNumber
NextSampleTuple (SampleScanState *node,
                 BlockNumber blockno,
                 OffsetNumber maxoffset);
```

################################ ####################InvalidOffsetNumber###
maxoffset######################

```
                                ##

NextSampleTuple####1 .. maxoffset################################
##### ################################################################
###### ############node->donetuples################################
########
```

```
                                ##

NextSampleTuple#####NextSampleBlock##############blockno#########
#########     #########NextSampleBlock##############################
##############NextSampleBlock################     ##################
##InvalidOffsetNumber########NextSampleTuple####################
###########
```

```
void
EndSampleScan (SampleScanState *node);
```

################### ##palloc###
#######################(#####NULL########)#######

#59#

PostgreSQL## ###########
##
##
######################### ######################################

##################3############# ##
##
##

59.1.

#####[90.062112]## #######
######################(Gather path)####
#####################################

```
typedef void (*set_rel_pathlist_hook_type) (PlannerInfo *root,
                                            RelOptInfo *rel,
                                            Index rti,
                                            RangeTblEntry *rte);
extern PGDLLIMPORT set_rel_pathlist_hook_type set_rel_pathlist_hook;
```

#####[81.488934]###
#######CustomPath##########add_path######rel############## #############
#CustomPath################ ######################## ########################
##CustomPath########
##add_#######rel#########add_partial_granite#####rel####### #############
##CustomPath################### ################ ####

```
typedef struct CustomPath
{
    Path      path;
    uint32    flags;
    List     *custom_paths;
    List     *custom_restrictinfo;
    List     *custom_private;
    const CustomPathMethods *methods;
} CustomPath;
```

#####[80.526021]#path###
flags##
###CUSTOMPATH_SUPPORT_BACKWARD_SCAN##################
#CUSTOMPATH_SUPPORT_MARK_RESTORE################
#CUSTOMPATH_SUPPORT_PROJECTION#flags######
(CUSTOMPATH_SUPPORT_PROJECTION################################Var########
####################################Var###################) #####
#custom_paths#################Path########## #######Plan##########
custom_private################################ #########nodeToString###
################# ##################################### methods###########
######################-
flags####################
########################### flags###############include######CUSTOMPATH_####
BACKWARD##########################CUSTOMPATH_####_###_##############
#CUSTOMPATH_####_####### CUSTOMPATH_####_##############################
######################## ###)
##########_#########-### ####

################################## ##############_restrictinfo############
############################## ########NIL### ####_##########################
########### ###########nodeToString###################### ##################
######################## ##
###################

################(join)################ ##
##################### ##CustomPath##
########

```
typedef void (*set_join_pathlist_hook_type) (PlannerInfo *root,
                                             RelOptInfo *joinrel,
                                             RelOptInfo *outerrel,
                                             RelOptInfo *innerrel,
                                             JoinType jointype,
                                             JoinPathExtraData *extra);
extern PGDLLIMPORT set_join_pathlist_hook_type set_join_pathlist_hook;
```

###################
#####

###########extra->restrictlist##
########   joinrel########CustomPath############################   ####
#joinrel###############CustomPath##########################

59.1.1.

```
Plan *(*PlanCustomPath) (PlannerInfo *root,
                         RelOptInfo *rel,
                         CustomPath *best_path,
                         List *tlist,
                         List *clauses,
                         List *custom_plans);
```

################### #########CustomScan#####################################
####59.2##########

```
List *(*ReparameterizeCustomPathByChild) (PlannerInfo *root,
                                          List *custom_private,
                                          RelOptInfo *child_rel);
```

######################child_rel##
######## ####################CustomPath#####custom_private###############
###############   ############
#reparameterize_path_by_child#adjust_appendrel_attrs##
#adjust_appendrel_attrs_multilevel#####

59.2.

####################################

```
typedef struct CustomScan
{
    Scan      scan;
    uint32    flags;
    List     *custom_plans;
    List     *custom_exprs;
```

```
    List     *custom_private;
    List     *custom_scan_tlist;
    Bitmapset *custom_relids;
    const CustomScanMethods *methods;
} CustomScan;
```

scan## flags#CustomPath#####
######### custom_plans###Plan####################
custom_exprs#setrefs.c###subselect.c############################## ##
#custom_private###
custom_scan_tlist##################NIL########## ########################
######################## ##################################
custom_scan_tlist#################### ################Var#################
############ custom_relids##
######## ##########################1############ methods####################
################################### ##################

CustomScan#######1############scan.scanrelid##########################
######## ##########scan.scanrelid########

######copyObject################custom####################################
############# ################CustomScan##############CustomScan#########
######## CustomPath#CustomScanState############

59.2.1.

```
Node *(*CreateCustomScanState) (CustomScan *cscan);
```

##CustomScan#CustomScanState########## ##############################
######################CustomScanState###################### ###########
#methods##
ExecInitCustomScan############BeginCustomScan##########################
##########################

59.3.

CustomScan###############CustomScanState######## ################

```
typedef struct CustomScanState
{
    ScanState ss;
    uint32    flags;
    const CustomExecMethods *methods;
} CustomScanState;
```

ss###
##ss.ss_currentRelation#NULL######## flags#CustomPath###CustomScan##
############ methods###
CustomScanState#copyObject#######################
#######################

59.3.1.

```
void (*BeginCustomScan) (CustomScanState *node,
                         EState *estate,
                         int eflags);
```

#####CustomScanState########## #########ExecInitCustomScan############
####################

```
TupleTableSlot *(*ExecCustomScan) (CustomScanState *node);
```

################ ##############################ps_ResultTupleSlot###### ###
###########NULL###############

```
void (*EndCustomScan) (CustomScanState *node);
```

CustomScanState####################### ####################################
#############################

```
void (*ReScanCustomScan) (CustomScanState *node);
```

####################################

```
void (*MarkPosCustomScan) (CustomScanState *node);
```

###############RestrPosCustomScan####################   #############
##CUSTOMPATH_SUPPORT_MARK_RESTORE############################

```
void (*RestrPosCustomScan) (CustomScanState *node);
```

MarkPosCustomScan############################   #############
##CUSTOMPATH_SUPPORT_MARK_RESTORE############################

```
Size (*EstimateDSMCustomScan) (CustomScanState *node,
                               ParallelContext *pcxt);
```

########################### ### #########
####### ##

```
void (*InitializeDSMCustomScan) (CustomScanState *node,
                                 ParallelContext *pcxt,
                                 void *coordinate);
```

######################## coordinate##EstimateDSMCustomScan###############
########### ##

```
void (*ReInitializeDSMCustomScan) (CustomScanState *node,
                                   ParallelContext *pcxt,
                                   void *coordinate);
```

########################
################################ ###########ReScanCustomScan###############
################################### ##############ReScanCustomScan##########
##################

```
void (*InitializeWorkerCustomScan) (CustomScanState *node,
                                    shm_toc *toc,
                                    void *coordinate);
```

InitializeDSMCustomScan### #########
##

```
void (*ShutdownCustomScan) (CustomScanState *node);
```

################################ ####################### ##################
###EndCustomScan############ ############DSM###########################
##DSM###

```
void (*ExplainCustomScan) (CustomScanState *node,
                           List *ancestors,
                           ExplainState *es);
```

##############EXPLAIN############## ################## ##################
#ScanState###
####

#60#

##

#########Martin Utesch#<utesch@aut.tu-freiberg.de>#################
University of Mining and Technology#Institute of Automatic Control##########

60.1.

################################ ##
##################PostgreSQL#B-tree######GiST#GIN#########################
############PostgreSQL###
########

###PostgreSQL## IBM#System R######
########
PostgreSQL#######################################

############University of Mining and Technology#Institute of Automatic Control##########
###########################PostgreSQL DBMS################### ##DBMS#########
###################################### ##
#########

###

60.2.

##########GA########################## ######################## ##############
############

################1##################
################################

60.1#####
####

#60.1

comp.ai.genetic#FAQ#####GA## GA##### ####################################

60.3. PostgreSQL#############GEQO#

GEQO###TSP######### ################### ########### ################1######################## ###################4-1-3-2## ############

```
   /\
  /\ 2
 /\ 3
4  1
```

###################4###1######3#######2############## ###1#2#3#4#PostgreSQL###### ########ID######

PostgreSQL####GEQO###################

- ####GA## ##### ##########################

- GA###TSP##################################
- TSP##

GEQO######## D. Whitley#Genitor################

GEQO########PostgreSQL###

60.3.1. GEQO############

GEQO## ######################## ## ######## ################################### #######GEQO################### ########## ## ### ############3################## ################################### ########## ################## ### ############## ######### ## ############## ################################## ############################ #############

#########GEQO####################geqo_seed################# geqo_seed##### GEQO## ############### ##geqo_seed##########

60.3.2. PostgreSQL GEQO########

```
#####################################        src/backend/optimizer/geqo/
geqo_main.c#gimme_pool_size#gimme_number_generations###########2#####
#######################
```

- ##########
- ####

######################

############TSP#######GA## TSP############ ## ################ #######################

60.4.

#####################################

- The Hitch-Hiker's Guide to Evolutionary Computation[1], (FAQ for news://comp.ai.genetic)
- Evolutionary Computation and its application to art and design[2]# Craig Reynolds#####
- [elma04]
- [fong]

[1] http://www.faqs.org/faqs/ai-faq/genetic/part1/
[2] https://www.red3d.com/cwr/evolve.html

#61#

####PostgreSQL## ################ ##

#############pg_am################## pg_am################################## ########SQL####CREATE ACCESS METHOD#DROP ACCESS METHOD############## #######

###################internal#########table_am_handler################### ## ###########SQL############################## ######TableAmRoutine#### ## ############ ###################static const################## ########*API*###### ####TableAmRoutine################################ #############C####### ##SQL###################### #################TableAmRoutine############### ## ############# ####src/include/access/tableam.h[1]#############

#####[92.342342]##################################AM################src/include/executor/tuptable.h[2]########## #################AM############## ##############################

#####AM########################## ####postgres################################## ######### ########65.6######PostgreSQL#######################

###############API#############AM######/#################################### #######TID#############65.6########## TIDs#########heap################### ##

############AM#postgres#WAL###################### WAL########WAL######## ######WAL#####################

###src/backend/access/transam/xlog.c#####################

#############################src/backend/access/heap/heapam_handler.c# #####heap##########

[1] https://git.postgresql.org/gitweb/?p=postgresql.git;a=blob;f=src/include/access/tableam.h;hb=HEAD
[2] https://git.postgresql.org/gitweb/?p=postgresql.git;a=blob;f=src/include/executor/tuptable.h;hb=HEAD

#62#

####PostgreSQL## ######## ################# ##

PostgreSQL################################### ############################## ####### ##############################pg_class################# ############ ######################### ## ##########################65.6######### ## ################################

##TID######## TID##### ##############65.6############ ################################ MVCC##1### ## ################################ ############### ## #HOT################################ ##

62.1. ##########API##

###########pg_am################# pg_am################################## ## #######SQL####CREATE ACCESS METHOD#DROP ACCESS METHOD############### ######

##################internal#####1######index_am_handler################## ## ############SQL########################### ######IndexAmRoutine#palloc### ## IndexAmRoutine#############*API*## ########################### ## ############################## #############C#####SQL##################### # ######62.2#########

###IndexAmRoutine###############

```
typedef struct IndexAmRoutine
{
    NodeTag     type;

    /*
     * Total number of strategies (operators) by which we can traverse/
search
     * this AM.  Zero if AM does not have a fixed set of strategy
 assignments.
     */
    uint16      amstrategies;
    /* total number of support functions that this AM uses */
    uint16      amsupport;
    /* opclass options support function number or 0 */
    uint16      amoptsprocnum;
    /* does AM support ORDER BY indexed column's value? */
    bool        amcanorder;
    /* does AM support ORDER BY result of an operator on indexed column?
 */
    bool        amcanorderbyop;
    /* does AM support backward scanning? */
    bool        amcanbackward;
    /* does AM support UNIQUE indexes? */
    bool        amcanunique;
    /* does AM support multi-column indexes? */
    bool        amcanmulticol;
```

```
    /* does AM require scans to have a constraint on the first index
 column? */
    bool        amoptionalkey;
    /* does AM handle ScalarArrayOpExpr quals? */
    bool        amsearcharray;
    /* does AM handle IS NULL/IS NOT NULL quals? */
    bool        amsearchnulls;
    /* can index storage data type differ from column data type? */
    bool        amstorage;
    /* can an index of this type be clustered on? */
    bool        amclusterable;
    /* does AM handle predicate locks? */
    bool        ampredlocks;
    /* does AM support parallel scan? */
    bool        amcanparallel;
    /* does AM support parallel build? */
    bool        amcanbuildparallel;
    /* does AM support columns included with clause INCLUDE? */
    bool        amcaninclude;
    /* does AM use maintenance_work_mem? */
    bool        amusemaintenanceworkmem;
    /* does AM summarize tuples, with at least all tuples in the block
     * summarized in one summary */
    bool        amsummarizing;
    /* OR of parallel vacuum flags */
    uint8       amparallelvacuumoptions;
    /* type of data stored in index, or InvalidOid if variable */
    Oid         amkeytype;

    /* interface functions */
    ambuild_function ambuild;
    ambuildempty_function ambuildempty;
    aminsert_function aminsert;
    aminsertcleanup_function aminsertcleanup;
    ambulkdelete_function ambulkdelete;
    amvacuumcleanup_function amvacuumcleanup;
    amcanreturn_function amcanreturn;   /* can be NULL */
    amcostestimate_function amcostestimate;
    amoptions_function amoptions;
    amproperty_function amproperty;     /* can be NULL */
    ambuildphasename_function ambuildphasename;   /* can be NULL */
    amvalidate_function amvalidate;
    amadjustmembers_function amadjustmembers; /* can be NULL */
    ambeginscan_function ambeginscan;
    amrescan_function amrescan;
    amgettuple_function amgettuple;     /* can be NULL */
    amgetbitmap_function amgetbitmap;   /* can be NULL */
    amendscan_function amendscan;
    ammarkpos_function ammarkpos;       /* can be NULL */
    amrestrpos_function amrestrpos;     /* can be NULL */

    /* interface functions to support parallel index scans */
    amestimateparallelscan_function amestimateparallelscan;    /* can be
 NULL */
    aminitparallelscan_function aminitparallelscan;    /* can be NULL */
    amparallelrescan_function amparallelrescan;    /* can be NULL */
} IndexAmRoutine;
```

###########################pg_opfamily#pg_opclass#pg_amop###pg_amproc### ######################### ## ############# ################36.16####### ###################

##################################pg_class################################## ####################################pg_index########## ####################

############################## ### #### ################ ####pg_index################## ########################### #####Relation#######################

#####[94.352562]#IndexAmRoutine############################## amcanunique### ###62.5######### amcanmultical#################################### ##amoptionalkey## amcanmultical#####amoptionalkey## ####### ## ######### ##################################### ####amoptionalkey### ############# ##############amoptionalkey###########NULL############## ## ####################NULL#####################NULL################### ###### ######################## ## ## ##amoptionalkey### ######NULL######################### ## ##1######NULL################### ########### ####(a,b)###########WHERE a = 4###################### ###### ########a = 4########################## #####b#NULL###################### ##### ##############NULL################### ###NULL######################## #amsearchnulls############# #########IS NULL###IS NOT NULL###############

amcaninclude##included########## ######### ################### ### ##amcanmultical=false#amcaninclude=true############ ################ ##include#################### ###amoptionalkey######include##NULL########## ######

amsummarizing## ################### ## ##############################BRIN######HOT################# ############## ############## ############HOT########

62.2.

##############IndexAmRoutine#################################

```
IndexBuildResult *
ambuild (Relation heapRelation,
         Relation indexRelation,
         IndexInfo *indexInfo);
```

#####[57.243816]################ ######################### ########### ## ###ambuild## #table_index_build_scan()## ## ########################palloc################## ################### ############################## ## ################ ###ambuild#######################_######_###_####()## ######################## ###########################palloc################## amcanbuildparallel## ################ ################ ##

```
void
ambuildempty (Relation indexRelation);
```

################################(INIT_FORKNUM)######## ##################### ########### ##

```
bool
```

```
aminsert (Relation indexRelation,
          Datum *values,
          bool *isnull,
          ItemPointer heap_tid,
          Relation heapRelation,
          IndexUniqueCheck checkUnique,
          bool indexUnchanged,
          IndexInfo *indexInfo);
```

###################### values###isnull#######################heap_tid#####
#####TID### ############################amcanunique######checkUnique####
############## ######################### 62.5######### ####################
#heapRelation############# (##############################)#

indexUnchanged############################### ############################
#######################MVCC########### ###UPDATE#######################
######AM############################
#####################
#####indexUnchanged################### ##################################
##########indexUnchanged######## ######AM##indexUnchanged################
#######################

checkUnique#UNIQUE_CHECK_PARTIAL################### ####################
##############################(####################)####### ##################
#####

#################################### ##################aminsert###########
#######

#####[57.986871]#SQL##########AM#######################################
##indexInfo->ii_Context####################indexInfo->ii_AmCache#####
NULL############### ###########AM####################SQL##################
####indexInfo->ii_################indexInfo->ii_AmCache###############
####null#####)# ##
#aminsertcleanup#############

```
void
aminsertcleanup (Relation indexRelation,
                 IndexInfo *indexInfo);
```

######indexInfo->ii_AmCache############################ ##################
##

```
IndexBulkDeleteResult *
ambulkdelete (IndexVacuumInfo *info,
              IndexBulkDeleteResult *stats,
              IndexBulkDeleteCallback callback,
              void *callback_state);
```

###################### ##
################ ####callback####callback(*TID*, callback_state) returns
bool########TID## NULL###palloc####
############################## amvacuumcleanup##################NULL#####
########

maintenance_work_mem####################ambulkdelete####################
stats########################## #VACUUM#################NULL#### #########
############################## #########stats#NULL######ambulkdelete########
########

```
IndexBulkDeleteResult *
amvacuumcleanup (IndexVacuumInfo *info,
                 IndexBulkDeleteResult *stats);
```

VACUUM###0####ambulkdelete############## ################################ # ################################## stats####ambulkdelete############ ##### #################ambulkdelete############NULL###### ###NULL######palloc#### ############ ########pg_class###############VERBOSE######VACUUM########### VACUUM#######################NULL############ ########################### ###

amvacuumcleanup#ANALYZE################ #####stats###NULL############## ## #####info->analyze_only############## ################################## ####################################

```
bool
amcanreturn (Relation indexRelation, int attno);
```

#####1## ###########attno#1### ## ###### ##include############################include################ ################### ###############IndexAmRoutine####amcanreturn######NULL#############

```
void
amcostestimate (PlannerInfo *root,
                IndexPath *path,
                double loop_count,
                Cost *indexStartupCost,
                Cost *indexTotalCost,
                Selectivity *indexSelectivity,
                double *indexCorrelation,
                double *indexPages);
```

################### ############62.6#######

```
bytea *
amoptions (ArrayType *reloptions,
           bool validate);
```

########reloptions########### ########NULL#reloptions################### *reloptions##name=value*#########text####### #####bytea################### #####relcache###rd_options############# bytea################################ ### ##################StdRdOptions######### *validate*################### ## *validate*################### ############pg_catalog#########*validate*##### ############################## ############## ######################### ################NULL############

```
bool
amproperty (Oid index_oid, int attno,
            IndexAMProperty prop, const char *propname,
            bool *res, bool *isnull);
```

amproperty##################pg_index_column_has_property############### ############### ## ##IndexAmRoutine####amproperty######NULL########## ###### ##amproperty#pg_indexam_has_property########*index_oid#attno*######## ##pg_index_has_property########*index_oid####attno*######## #pg_index_column_has_property########*index_oid####attno*#1########## *prop*##################*propname*############## ################

##*prop*#AMPROP_UNKNOWN##### ############################*propname*######
############pg_strcasecmp################ ################*prop*#########
amproperty#####true##########################*res###############NULL##
####*isnull#true################ ################false#########
amproperty#####false#######################################

#######################AMPROP_DISTANCE_ORDERABLE################# ######
#####################NULL####### ###########################amcanreturn##
#################AMPROP_RETURNABLE################# ####################
################

```
char *
ambuildphasename (int64 phasenum);
```

######################### #######pgstat_progress_update_param###########
################## ##########pg_stat_progress_create_index##########

```
bool
amvalidate (Oid opclassoid);
```

##############################
################ amvalidate############################ ##########INFO###
#ereport##############

```
void
amadjustmembers (Oid opfamilyoid,
                 Oid opclassoid,
                 List *operators,
                 List *functions);
```

###CREATE
OPERATOR CLASS#ALTER OPERATOR FAMILY ADD############ ####
##*opclassoid*#InvalidOid### List###amapi.h#######OpFamilyMember#######
##################amvalidate############## #####amadjustmembers#######
############################ ##
#################### ##################### OpFamilyMember###############
##CREATE OPERATOR CLASS#####opclass###################ALTER OPERATOR
FAMILY ADD###opfamily############## #################amadjustmembers#####
################# #####GIN#GiST#SP-GiST#################opclass###############
#######################################opfamily################ ################
##############################

##WHERE######################## ####
################62.3########### ##
################ ##

```
IndexScanDesc
ambeginscan (Relation indexRelation,
             int nkeys,
             int norderbys);
```

################ nkeys###norderbys###################################### #
####################### ############################## ###palloc############
#######################RelationGetIndexScan()######################
#######ambeginscan##############################
##amrescan######

```
void
amrescan (IndexScanDesc scan,
          ScanKey keys,
          int nkeys,
          ScanKey orderbys,
          int norderbys);
```

####################### #################### ####################
##key#orderbys########NULL###### ambeginscan###########################
################## ##
#################

```
bool
amgettuple (IndexScanDesc scan,
            ScanDirection direction);
```

#######
################ ##########TID#scan####### ############################
################ ## #####
##amgettuple#scan->xs_recheck################## ########################
####### ## #
########################## ############################## ################
#amgettuple#################

#################################(###amcanreturn###############)############
##########scan->xs_want_itup##
######## amcanreturn###############NULL######## #########scan-
>xs_itupdesc############scan->xs_itup#####IndexTuple###########
##scan->xs_hitupdesc############scan->xs_hitup#####HeapTuple#######
#################IndexTuple##################
############ ########amgettuple#amrescan###amendscan###################
########

amgettuple## ######
##IndexAmRoutine####amgettuple#####NULL##############

```
int64
amgetbitmap (IndexScanDesc scan,
             TIDBitmap *tbm);
```

################################TIDBitmap######### #######################ID#
####OR###### #######################AM######################## ###ID######
######amgetbitmap######ID####################### ##
#amgettuple#xs_recheck############## ##################################
#######
########## amgetbitmap###amgettuple########################## 62.3#####
##amgetbitmap##################

amgetbitmap## ######
##IndexAmRoutine#####amgetbitmap#####NULL##############

```
void
amendscan (IndexScanDesc scan);
```

################## scan################## ######################
##ambeginscan#######################################

```
void
```

```
ammarkpos (IndexScanDesc scan);
```

############### ########1#######1##########################

ammarkpos## #########
#IndexAmRoutine####ammarkpos#####NULL###########

```
void
amrestrpos (IndexScanDesc scan);
```

#######################

amrestrpos## #########
#IndexAmRoutine####amrestrpos#####NULL###########

###
############ ###
################
#
######################################

```
Size
amestimateparallelscan (int nkeys,
                        int norderbys);
```

#####
##ParallelIndexScanDescData#AM##

######nkeys#norderbys################################ ####amrescan.Quarter
##############################

##################################0############################

```
void
aminitparallelscan (void *target);
```

target#####
#amestimateparallelscan###
#######

##

```
void
amparallelrescan (IndexScanDesc scan);
```

######aminitparallelscan####
##############################

62.3.

#####################################TID################################ ####
###

#######index_key operator constant######WHERE######### ####*index_key*####
######1###*operator*####################1#### ###############AND#####0#######
######### ############################

##################
################################

####################################
###########

#####
###WHERE######### ##
##WHERE x > 4 AND x > 14####x#B-tree###################B-tree amrescan######
################################ amrescan#################################
#######################

##############################
############

- ########btree#######################amcanorder############### ##############
  ################btree####################

- #######################amcanorderbyop############### ##########ORDER BY
  index_key operator constant######################### #################
  #amrescan##########

amgettuple###direction####### ###ForwardScanDirection########
#BackwardScanDirection############### amrescan########
#BackwardScanDirection###
######## #####amgettuple###
####amcanorder######################### #########amgettuple############
############################### ####amcanbackward#########################
#########

###
#########1################## ###ammarkpos######
################## ##########################IndexAmRoutine#ammarkpos####
#amrestrpos######################NULL##########

############
############################
############################## ################################## ############
###

##############################(##############)##########TID######################
############################## ##############TID####################I/O########
#########MVCC############################

amgettuple#########amgetbitmap##
################/######################amgettuple###########
##amgetbitmap#amgettuple## ###
###amgetbitmap#################################### #######################
############# ###amgetbitmap#direction########### #######################
######## ###amgetbitmap###
####amgetbitmap###########62.4###################

####################API#########amgettuple#####amgetbitmap################
#############

62.4.

PostgreSQL#################
##########AccessShareLock###### ######VACUUM#############
#RowExclusiveLock###### ##
########## ############REINDEX#############ACCESS EXCLUSIVE#########
(CONCURRENTLY######SHARE UPDATE EXCLUSIVE######)#

B-tree################
##src/backend/access/nbtree/README# src/backend/access/hash/README##
#####################

PostgreSQL####
#########
#########

- ################################ ### ##
  ################################### ####62.5###########

- #########VACUUM##

- ##############amgettuple#################################### #
  ##ambulkdelete################################### ####################

###########VACUUM##VACUUM##############
################## #########heap_fetch()##################################
############### ## #############
################################MVCC########################### ####MVCC##
###########(SnapshotAny##)################################## ################
######################
########ambulkdelete########
####################VACUUM######################### ##########################
##################################

################################## ### #
################ ###########TID###
#########MVCC##################
####################MVCC##################################

amgetbitmap## ######MVCC####
###############################

ampredlocks###
########################### ##
################ ##
########################### ##
#####################################

62.5.

PostgreSQL##SQL########################## ############################## ######
#############amcanunique######### ######b-tree############### INCLUDE########
####################

MVCC#################################### ############################### #####
########MVCC################################### ##############################
#####################

- ##################################### #######UPDATE########################
  ####### #################UPDATE###########

- ### #####
  ############# ################################ ################################
  ################

- ##
  ################

#######CREATE UNIQUE INDEX CONCURRENTLY##############

################################### ### ################################### ############################ ############ ######## ## ## ##

########################### ############################### ################### ############ ## ################################### ################################ ######## ## ##############################HOT ########################### ###########aminsert############checkUnique#### ########

- UNIQUE_CHECK_NO##############(################)########
- UNIQUE_CHECK_YES##
- UNIQUE_CHECK_PARTIAL#################### PostgreSQL#################### ######## ################################### ###aminsert################## ############## ###########################

  ## ########################## ########## ######################## ############################## ################### #####

- UNIQUE_CHECK_EXISTING## ## #aminsert## ############## #### ## ########################## ##

  UNIQUE_CHECK_EXISTING## ################ aminsert############################## ################### ##################################### ##################################### ################

62.6.

amcostestimate##########################WHERE####ORDER BY############## ####################### ########################WHERE######################## ############################ ## ### amcostestimate## ###################

#####amcostestimate######################

```
void
amcostestimate (PlannerInfo *root,
                IndexPath *path,
                double loop_count,
                Cost *indexStartupCost,
                Cost *indexTotalCost,
                Selectivity *indexSelectivity,
                double *indexCorrelation,
                double *indexPages);
```

###3############

root

#######################

path

################# ########################

loop_count

################################# ######################################
############ ######1#########################*loop_count*################
########################

###5################

**indexStartupCost*

########################

**indexTotalCost*

######################

**indexSelectivity*

#################

**indexCorrelation*

####################################

**indexPages*

####################

#########SQL##############C######################### #######/################
################

#############src/backend/optimizer/path/costsize.c####################
###seq_page_cost##############random_page_cost#########1##############
#cpu_index_tuple_cost################################ ##############
indexquals##########################cpu_operator_cost##################

####################################CPU#######################################
##############

##################
########## ##

indexSelectivity## #############
##

indexCorrelation######################-1.0##1.0######################### #####
#####################################

indexPages################### ####################################

loop_count###1######################1####################

#####

#####################

1. ## ################################### ###clauselist_selectivity()#########

   ```
   *indexSelectivity = clauselist_selectivity(root, path->indexquals,
                                              path->indexinfo->rel-
   >relid,
                                              JOIN_INNER, NULL);
   ```

2. ########################## ################*indexSelectivity*############ ############################ ####################path->indexinfo####### #################

3. ########################### #####*indexSelectivity*####################### ######

4. ################### ######################

   ```
   /*

    * ############################
    * random_page_cost#########seq_page_cost############
    * ##########indexquals###############
    * ############################
    */
   cost_qual_eval(&index_qual_cost, path->indexquals, root);
   *indexStartupCost = index_qual_cost.startup;
   *indexTotalCost = seq_page_cost * numIndexPages +
       (cpu_index_tuple_cost + index_qual_cost.per_tuple) *
    numIndexTuples;
   ```

   ##

5. ############### 1###########################pg_statistic############## ###### ##########################

#########src/backend/utils/adt/selfuncs.c######

#63# Write Ahead Logging for Extensions

##
PostgreSQL################2###########

###############WAL############## ########################WAL###### ######
######WAL##########################

############################WAL####
######################## ###################

63.1. ##WAL####

#####[53.846154]######WAL###########################WAL####################
###########WAL########## ############################ ######WAL############
#############WAL################WAL############ ######################

> ##
>
> ##WAL######################## ########################WAL############
> ######

##WAL###########API#access/generic_xlog.h########access/transam/generic_xlog.c#########

##WAL##########WAL##############################

1. `state = GenericXLogStart(relation)` ##################WAL##########
   ###
2. `page = GenericXLogRegisterBuffer(state, buffer, flags)` ########WAL#################### ######################### ################### ##################### 3######################## ## ###############GENERIC_XLOG_FULL_IMAGE######WAL################## ################## ## WAL###### ######################GenericXLogRegisterBuffer############
3. #########################
4. `GenericXLogFinish(state)`################WAL#########

WAL#######################GenericXLogAbort(state)############## ######## #######################

##WAL###########################

- ################ ######GenericXLogRegisterBuffer()################# ###### ########WAL###########BufferGetPage()############ ###### ### ################ #GenericXLogRegisterBuffer()####GenericXLogFinish()################ ##
- ##2###########3########################### ########################### ##### ######################################
- ##WAL##################MAX_GENERIC_XLOG_PAGES### ##################

- ##WAL#############################pd_lower#pd_upper#################### #######
- #####################GenericXLogStart()################### ## ##GenericXLogStart()#GenericXLogFinish()############################## ## #################GenericXLogFinish()######## ####### #GenericXLogAbort()#######################
- GenericXLogFinish()#############LSN############### #################### ###
- #################WAL############################## #################### ####################
- ##WAL## ######################### ####
- #######GENERIC_XLOG_FULL_IMAGE###########WAL######################### ############# #################### ## #####

63.2. ####WAL##########

#####[93.670886]#######PostgreSQL##########WAL########################## ##### #######WAL##############

######extension###WAL### ##############

#######WAL#######################################RmgrData#########
PostgreSQL####src/backend/access/transam/README#src/include/access/xlog_internal.h#########

```
/*
 * Method table for resource managers.
 *
 * This struct must be kept in sync with the PG_RMGR definition in
 * rmgr.c.
 *
 * rm_identify must return a name for the record based on xl_info (without
 * reference to the rmid). For example, XLOG_BTREE_VACUUM would be named
 * "VACUUM". rm_desc can then be called to obtain additional detail for
 the
 * record, if available (e.g. the last block).
 *
 * rm_mask takes as input a page modified by the resource manager and
 masks
 * out bits that shouldn't be flagged by wal_consistency_checking.
 *
 * RmgrTable[] is indexed by RmgrId values (see rmgrlist.h). If rm_name is
 * NULL, the corresponding RmgrTable entry is considered invalid.
 */
/*
 * ######
 * ##################
 *
 * #######rmgr.c#PG_RMGR#################
 *
 * rm_identify##xl_info#rmid############################
 * ####XLOG_BTREE_VACUUM#"VACUUM"######
 * ####rm_desc#######################
 * ##################
```

```
 *
 * rm_mask################################
 * wal_consistency_checking#########################
 *
 * RmgrTable[]#RmgrId###############(rmgrlist.h###)#
 * rm_name#NULL#######RmgrTable##############
 */
typedef struct RmgrData
{
    const char *rm_name;
    void        (*rm_redo) (XLogReaderState *record);
    void        (*rm_desc) (StringInfo buf, XLogReaderState *record);
    const char *(*rm_identify) (uint8 info);
    void        (*rm_startup) (void);
    void        (*rm_cleanup) (void);
    void        (*rm_mask) (char *pagedata, BlockNumber blkno);
    void        (*rm_decode) (struct LogicalDecodingContext *ctx,
                              struct XLogRecordBuffer *buf);
} RmgrData;
```

#####src/test/modules/test_custom_rmgrs#################WAL##########
##########

#####################

```
/*
 * Register a new custom WAL resource manager.
 *
 * Resource manager IDs must be globally unique across all extensions.
 Refer
 * to https://wiki.postgresql.org/wiki/CustomWALResourceManagers to
 reserve a
 * unique RmgrId for your extension, to avoid conflicts with other
 extension
 * developers. During development, use RM_EXPERIMENTAL_ID to avoid
 needlessly
 * reserving a new ID.
 */
/*
 * ######
 * ####WAL#############
 *
 * #########ID##################################
 * https://wiki.postgresql.org/wiki/CustomWALResourceManagers ######
 * #######RmgrId######################
 * ######RM_EXPERIMENTAL_ID#########ID##################
 */
extern void RegisterCustomRmgr(RmgrId rmid, const RmgrData *rmgr);
```

RegisterCustomRmgr########_PG_init################ ############
##*rmid*#RM_EXPERIMENTAL_ID######### ####################Custom WAL Resource Manager[1]################ID#########

#########################shared_preload_libraries######PostgreSQL#################
##

##
#####[87.632509]###################WAL############shared_preload_libraries### ############ ######PostgreSQL#####WAL################################

[1] https://wiki.postgresql.org/wiki/CustomWALResourceManagers

```
#########  #####extension  WAL#######################shared_####
_libraries############### ######PostgreSQL####WAL###################
#######################
```

#64# Built-in Index Access Methods

64.1. B-Tree######

64.1.1.

PostgreSQL#####btree#multi-way balanced tree#################### ################# ########btree############## ##############################TOAST########1/3### ########

#####[92.688172]#btree##############################btree################ ##PostgreSQL### #######btree############## ##############btree AM##################################

64.1.2. B-Tree###########

36.3#######btree########5#################### <#<=#=#>=####>### <>########### ############################ ########WHERE##<>###################### ####### ########<>#btree###################### #########pg_amop##########=###########

########################

btree##############################

- =################## ################NULL#*A*#*B*#*C*#####
  - *A = A*#########
  - *A = B*###*B = A*########
  - *A = B*##*B= C*###*A = C*#######
- <##########################NULL#*A*#*B*#*C*#####
  - *A < A*###########
  - *A < B*##*B < C*###*A < C*#######
- ####################NULL#*A*#*B*#####
  - *###A < B#A = B#B < A*############## ##############################

##3######=#<##############################

#######################################*A*#*B*#*C*#################### ########2#### #3## ############float8####### #numeric##float8################float8#numeric######################## float8############# #####float8################numeric#################### #########

###(binary-coercion)############# ################

###########btree################################ ##############################
###########################2########################### ######3##############
btree##

64.1.3. B-Tree######

36.9#######btree##############4################### 5###################

`order`

btree###1#
#pg_amproc###################amprocleftype/amprocrighttype#######
######pg_amop########################### #####2###NULL#*A*#*B*#### *A* < *B*#*A* = *B*#####*A* > *B*##########< 0#0#####> 0###int32############## NULL###########
#################### ####src/backend/access/nbtree/nbtcompare.c#####
#####

##OID######
#PG_GET_COLLATION()#########

`sortsupport`

####btree################################2######## ######################
############################## ######API#src/include/utils/sortsupport.h#########

`in_range`

####btree#####*in_range*#########################3######## ###btree############
#######################RANGE *offset* PRECEDING#RANGE *offset* FOLLOWING#
#########4.2.8######WINDOW############## ############################
##########*offset*##########

in_range#######################

```
in_range(val type1, base type1, offset type2, sub bool, less bool)
returns bool
```

val#*base*## #####
##*offset*#################################### ##########time_ops#
#interval##*offset*###in_range########## ######################
#*offset*#####in_range######### #in_range####pg_amproc#type1###
#amprocleftype#type2####amprocleftype##########

in_range##############2##Boolean############## #########*base*#*offset*####
##########*val*###########

- !*sub*## !*less*##### *val* >= (*base* + *offset*)#####
- !*sub*## *less*##### *val* <= (*base* + *offset*)#####
- *sub*## !*less*##### *val* >= (*base* - *offset*)#####
- *sub*## *less*##### *val* <= (*base* - *offset*)#####

#################*offset*########### ##############
##ERRCODE_INVALID_PRECEDING_OR_FOLLOWING_SIZE(22013)##########invalid preceding or following size in window function(######################)########### ##

##SQL############# ####
##in_range#########

###########in_range#########*base* + *offset*#*base* - *offset*##############
############# ################################ ######infinity###NaN#######
######in_range##

in_range################################### #############*offset*#*sub*###
############

- *less* = true#in_range#####*val1*#*base*###########*base*####*val2* <= *val1*###
  ############
- *less* = true#in_range######*val1*#*base*###########*base*####*val2* >= *val1*##
  #############
- *less* = true#in_range#####*val*#*base1*###########*val*####*base2* >= *base1*##
  #############
- *less* = true#in_range####*val*#*base1*###########*val*####*base2* <= *base1*####
  ###########

less = false######################

###########(type1)#############PG_GET_COLLATION()######in_range#####
#####OID######

in_range######STRICT######NULL###############

equalimage

#######btree#####equalimage (##############)#########################4####
###########btree############################## ######equalimage#####
########################

equalimage#######################

equalimage(*opcintype* oid) returns bool

########################### true######*A*###*B*##############################
#############order###0################################ equalimage######
#######false###########################

opcintype#######################pg_type.oid### ########equalimage###
############################## *opcintype*#######################OID####
#PG_GET_COLLATION()######equalimage########

############true#################equalimage######OID################
########################equalimage##############################
#true################################

############################### ############varlena#######################
###TOAST###########datum################################## ###########
#equalimage###true#######datum_image_eq() C###########order########
###############OID#equalimage#order#############

###
####### ################equalimage################################ ##
##
##

##PostgreSQL##################################equalimage########### ######
######btequalimage()################################## text###########
###########btvarstrequalimage()##################################### ###
##

options

########B-tree######options###5#####
##

options##########################

options(*relopts* local_relopts *) returns void

####local_relopts## ####
#####PG_HAS_OPCLASS_OPTIONS()###PG_GET_OPCLASS_OPTIONS()#########
################

######options##########B-Tree############## B-tree#GiST#SP-GiST#GIN###BRIN#
######################### ##########options####B-tree######################
######## ##################B-tree############PostgreSQL##B-tree##############
##########

64.1.4.

####################B-Tree#################### ######B-Tree#################
########src/backend/access/nbtree/README#########

64.1.4.1. B-Tree###

PostgreSQL#B-Tree## ##############
########################## ############################### ###################
###########99#######
################65.6######################

##B-Tree############## ################
##################################
################################## ###################### ###################
############################## ##
##

64.1.4.2.

B-Tree########MVCC## ########
###################################### ##
######### ################HOT############UPDATE################ UPDATE######
############
######## ######UPDATE## ################
################################### ##
#########UPDATE######################

B-Tree## ###########
############################## ####UPDATE############################## ######
############################### ##
###
###
##########

####################24.1.6#########autovacuum#################################
######

##

B-Tree## ######################## ##
################### #########################(########LP_DEAD############
####)################## ##
###########

##############################LP_DEAD################################
################ PostgreSQL 14######B-Tree################## ################
###
#UPDATE##########################

##############
###########################UPDATE####################B-Tree############## #####
##################
##UPDATE## ########
##VACUUM###
############

VACUUM## #######
########
################TID############# ##########################TID################
############VACUUM############################

64.1.4.3.

###
############# ################ ##########################B-Tree###################
################

#################
########TID##################
################ ################### ############################### ###########
###########################

##

B-Tree######B-Tree#######=#####NULL####################NULL#############
######B-Tree####################NULL########################

###
###################################### GIN###################B-Tree#############
##############
#################### ##
##

CREATE INDEX#REINDEX## ########
#
#########################TID######### ################################## ####
#CREATE INDEX#REINDEX############1#################

##
################# deduplicate_items#####################################
####
####################

#########(#####)################## ##
##
######### ### ##########################
##

###
##################
########################
#deduplicate_items = off######### ################################

################################### ########CREATE INDEX####REINDEX#######
#######

###

- #################text#varchar#####char########### ######################
  ##############
- numeric########### ##############################
- jsonb#B-Tree###########numeric########jsonb###########
- float4###float8########### ######-0#0############################## #####
  #############

######################PostgreSQL#####################

- ####################################

###

- INCLUDE##################

64.2. GiST######

64.2.1.

GiST#########Generalized Search Tree####### ######################################
#################### B-tree#R-tree##################GiST#############

GiST####1###

############################GiST###################[1]###Marcel Kornacker####Access Methods for Next-Generation Database Systems[2]########### PostgreSQL####GiST######
##Teodor Sigaev#Oleg Bartunov############# #############[3]#############

[1] http://gist.cs.berkeley.edu/
[2] http://www.sai.msu.su/~megera/postgres/gist/papers/concurrency/access-methods-for-next-generation.pdf.gz
[3] http://www.sai.msu.su/~megera/postgres/gist/

64.2.2.

PostgreSQL######## 64.1###GiST############ (##F####################GiST########### ########)

#64.1 ####GiST######

##	############	#######
box_ops	<< (box, box)	<-> (box, point)
	&< (box, box)	
	&& (box, box)	
	&> (box, box)	
	>> (box, box)	
	~= (box, box)	
	@> (box, box)	
	<@ (box, box)	
	&<| (box, box)	
	<<| (box, box)	
	|>> (box, box)	
	|&> (box, box)	
circle_ops	<< (circle, circle)	<-> (circle, point)
	&< (circle, circle)	
	&> (circle, circle)	
	>> (circle, circle)	
	<@ (circle, circle)	
	@> (circle, circle)	
	~= (circle, circle)	
	&& (circle, circle)	
	|>> (circle, circle)	
	<<| (circle, circle)	
	&<| (circle, circle)	
	|&> (circle, circle)	
inet_ops	<< (inet, inet)	
	<<= (inet, inet)	
	>> (inet, inet)	
	>>= (inet, inet)	
	= (inet, inet)	
	<> (inet, inet)	
	< (inet, inet)	
	<= (inet, inet)	
	> (inet, inet)	
	>= (inet, inet)	
	&& (inet, inet)	

##	############	#######
multirange_ops	= (anymultirange, anymultirange)	
	&& (anymultirange, anymultirange)	
	&& (anymultirange, anyrange)	
	@> (anymultirange, anyelement)	
	@> (anymultirange, anymultirange)	
	@> (anymultirange, anyrange)	
	<@ (anymultirange, anymultirange)	
	<@ (anymultirange, anyrange)	
	<< (anymultirange, anymultirange)	
	<< (anymultirange, anyrange)	
	>> (anymultirange, anymultirange)	
	>> (anymultirange, anyrange)	
	&< (anymultirange, anymultirange)	
	&< (anymultirange, anyrange)	
	&> (anymultirange, anymultirange)	
	&> (anymultirange, anyrange)	
	-|- (anymultirange, anymultirange)	
	-|- (anymultirange, anyrange)	
point_ops	|>> (point, point)	<-> (point, point)
	<< (point, point)	
	>> (point, point)	
	<<| (point, point)	
	~= (point, point)	
	<@ (point, box)	
	<@ (point, polygon)	
	<@ (point, circle)	
poly_ops	<< (polygon, polygon)	<-> (polygon, point)
	&< (polygon, polygon)	
	&> (polygon, polygon)	
	>> (polygon, polygon)	
	<@ (polygon, polygon)	
	@> (polygon, polygon)	
	~= (polygon, polygon)	

##	###########	#######
	&& (polygon, polygon)	
	<<| (polygon, polygon)	
	&<| (polygon, polygon)	
	|&> (polygon, polygon)	
	|>> (polygon, polygon)	
range_ops	= (anyrange, anyrange)	
	&& (anyrange, anyrange)	
	&& (anyrange, anymultirange)	
	@> (anyrange, anyelement)	
	@> (anyrange, anyrange)	
	@> (anyrange, anymultirange)	
	<@ (anyrange, anyrange)	
	<@ (anyrange, anymultirange)	
	<< (anyrange, anyrange)	
	<< (anyrange, anymultirange)	
	>> (anyrange, anyrange)	
	>> (anyrange, anymultirange)	
	&< (anyrange, anyrange)	
	&< (anyrange, anymultirange)	
	&> (anyrange, anyrange)	
	&> (anyrange, anymultirange)	
	-|- (anyrange, anyrange)	
	-|- (anyrange, anymultirange)	
tsquery_ops	<@ (tsquery, tsquery)	
	@> (tsquery, tsquery)	
tsvector_ops	@@ (tsvector, tsquery)	

#########inet_ops#######inet#cidr################ ##########CREATE INDEX# ########## ##############

```
CREATE INDEX ON my_table USING GIST (my_inet_column inet_ops);
```

64.2.3.

GiST## GiST#

####PostgreSQL######B-tree# ################### ####PostgreSQL################B-tree#################### ## ####B-tree######<#=#>#######################################

#####PostgreSQL#B-tree#######################x###y########x###y##########x###y ############################## ## ########## ####GiST## ####################################

GiST## # ## ##################B-tree#R-tree# ##################### ######GiST#####################################

GiST#######################5#################6###### ##########
##same#consistent#union##################### ##################
#penalty#picksplit########## ##############compress#decompress#########
########################
###########C############# #######PostgreSQL################### #############
#varlena########## ##########SQL##########CREATE OPERATOR CLASS####
#STORAGE################# ######8#######distance### ####################
################## ######9######fetch##compress##########################
############################ ######10######options#########################
######11######sortsupport##GiST####################

consistent

#######p#######q##
##*indexed_columnindexable_operator* q################################
################################## ######################################
################### ###true####recheck################ ##################
######### recheck = false############################recheck= true######
############### #############*indexable_operator*#######################
#######GiST#####################################

#####SQL#############

```
CREATE OR REPLACE FUNCTION my_consistent(internal, data_type, smallint,
 oid, internal)
RETURNS bool
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

####C################################

```
PG_FUNCTION_INFO_V1(my_consistent);

Datum
my_consistent(PG_FUNCTION_ARGS)
{
    GISTENTRY  *entry = (GISTENTRY *) PG_GETARG_POINTER(0);
    data_type  *query = PG_GETARG_DATA_TYPE_P(1);
    StrategyNumber strategy = (StrategyNumber) PG_GETARG_UINT16(2);
    /* Oid subtype = PG_GETARG_OID(3); */
    bool       *recheck = (bool *) PG_GETARG_POINTER(4);
    data_type  *key = DatumGetDataType(entry->key);
    bool        retval;

    /*

     * strategy#key###query##################
     *
     * ############################GIST_LEAF(entry)##########
     * ####### = ################
     *################union()############################
     */


    *recheck = true;        /* ############## */

    PG_RETURN_BOOL(retval);
}
```

####key############query################## StrategyNumber############
################## ###CREATE OPERATOR CLASS##########1########

##
#######query######################### (##########1################## ######
##query############################) consistent###SQL##################
#############query#####################################

union

#################### ##

#####SQL#############

```
CREATE OR REPLACE FUNCTION my_union(internal, internal)
RETURNS storage_type
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

####C################################

```
PG_FUNCTION_INFO_V1(my_union);

Datum
my_union(PG_FUNCTION_ARGS)
{
    GistEntryVector *entryvec = (GistEntryVector *)
 PG_GETARG_POINTER(0);
    GISTENTRY  *ent = entryvec->vector;
    data_type  *out,
               *tmp,
               *old;
    int         numranges,
                i = 0;

    numranges = entryvec->n;
    tmp = DatumGetDataType(ent[0].key);
    out = tmp;

    if (numranges == 1)
    {
        out = data_type_deep_copy(tmp);

        PG_RETURN_DATA_TYPE_P(out);
    }

    for (i = 1; i < numranges; i++)
    {
        old = out;
        tmp = DatumGetDataType(ent[i].key);
        out = my_union_implementation(out, tmp);
    }

    PG_RETURN_DATA_TYPE_P(out);
}
```

###############union(X, Y, Z) = union(union(X, Y), Z)###############
##GiST###########union##

union#######(##)#####################
######## union######palloc()######################### ####################
############

#########union###1###internal#####GistEntryVector######## 2############
################ (union##)

compress

#################################### compress##############################
##########

#####SQL##############

```
CREATE OR REPLACE FUNCTION my_compress(internal)
RETURNS internal
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

####C#################################

```
PG_FUNCTION_INFO_V1(my_compress);

Datum
my_compress(PG_FUNCTION_ARGS)
{
    GISTENTRY  *entry = (GISTENTRY *) PG_GETARG_POINTER(0);
    GISTENTRY  *retval;

    if (entry->leafkey)
    {

        /* ######## entry->key ##### */
        compressed_data_type *compressed_data =
 palloc(sizeof(compressed_data_type));


        /* entry->key ... ## *compressed_data ### */

        retval = palloc(sizeof(GISTENTRY));
        gistentryinit(*retval, PointerGetDatum(compressed_data),
                      entry->rel, entry->page, entry->offset, FALSE);
    }
    else
    {

        /* ################### */
        retval = entry;
    }

    PG_RETURN_POINTER(retval);
}
```

#####*compressed_data_type*#####################################

decompress

#######################GiST################## decompress##############
GiST############################## #decompress#####compress###########
####### ###compress#######decompress######################### ##GiST####
################################decompress#fetch##############

#####SQL##############

```
CREATE OR REPLACE FUNCTION my_decompress(internal)
RETURNS internal
```

```
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;

####C#################################


PG_FUNCTION_INFO_V1(my_decompress);

Datum
my_decompress(PG_FUNCTION_ARGS)
{
    PG_RETURN_POINTER(PG_GETARG_POINTER(0));
}

######################### ################################################
########
```

penalty

```
#####################################      ########penalty#############
penalty################### ####################

#####SQL##############


CREATE OR REPLACE FUNCTION my_penalty(internal, internal, internal)
RETURNS internal
AS 'MODULE_PATHNAME'

LANGUAGE C STRICT;  -- penalty###################

####C#################################


PG_FUNCTION_INFO_V1(my_penalty);

Datum
my_penalty(PG_FUNCTION_ARGS)
{
    GISTENTRY  *origentry = (GISTENTRY *) PG_GETARG_POINTER(0);
    GISTENTRY  *newentry = (GISTENTRY *) PG_GETARG_POINTER(1);
    float      *penalty = (float *) PG_GETARG_POINTER(2);
    data_type  *orig = DatumGetDataType(origentry->key);
    data_type  *new = DatumGetDataType(newentry->key);

    *penalty = my_penalty_implementation(orig, new);
    PG_RETURN_POINTER(penalty);
}

##########penalty######float######### ######3############################
##################################

penalty######################### ########################################
############## ##################################
```

picksplit

```
#######################################################################

#####SQL##############


CREATE OR REPLACE FUNCTION my_picksplit(internal, internal)
RETURNS internal
AS 'MODULE_PATHNAME'
```

```
LANGUAGE C STRICT;

####C#################################


PG_FUNCTION_INFO_V1(my_picksplit);

Datum
my_picksplit(PG_FUNCTION_ARGS)
{
    GistEntryVector *entryvec = (GistEntryVector *)
 PG_GETARG_POINTER(0);
    GIST_SPLITVEC *v = (GIST_SPLITVEC *) PG_GETARG_POINTER(1);
    OffsetNumber maxoff = entryvec->n - 1;
    GISTENTRY  *ent = entryvec->vector;
    int         i,
                nbytes;
    OffsetNumber *left,
                *right;
    data_type  *tmp_union;
    data_type  *unionL;
    data_type  *unionR;
    GISTENTRY **raw_entryvec;

    maxoff = entryvec->n - 1;
    nbytes = (maxoff + 1) * sizeof(OffsetNumber);

    v->spl_left = (OffsetNumber *) palloc(nbytes);
    left = v->spl_left;
    v->spl_nleft = 0;

    v->spl_right = (OffsetNumber *) palloc(nbytes);
    right = v->spl_right;
    v->spl_nright = 0;

    unionL = NULL;
    unionR = NULL;


    /* ############ */
    raw_entryvec = (GISTENTRY **) malloc(entryvec->n * sizeof(void *));
    for (i = FirstOffsetNumber; i <= maxoff; i = OffsetNumberNext(i))
        raw_entryvec[i] = &(entryvec->vector[i]);

    for (i = FirstOffsetNumber; i <= maxoff; i = OffsetNumberNext(i))
    {
        int         real_index = raw_entryvec[i] - entryvec->vector;

        tmp_union = DatumGetDataType(entryvec->vector[real_index].key);
        Assert(tmp_union != NULL);

        /*

         * #########################unionL#unionR###
         * ####v->spl_left #### v->spl_right ###############
         * ##############
         */

        if (my_choice_is_left(unionL, curl, unionR, curr))
        {
            if (unionL == NULL)
                unionL = tmp_union;
            else
                unionL = my_union_implementation(unionL, tmp_union);
```

```
                *left = real_index;
                ++left;
                ++(v->spl_nleft);
            }
            else
            {
                /*

                 * ####
                 */
            }
        }

        v->spl_ldatum = DataTypeGetDatum(unionL);
        v->spl_rdatum = DataTypeGetDatum(unionR);
        PG_RETURN_POINTER(v);
    }
```

picksplit#########v######################### v#########################
#####

penalty###picksplit########################## penalty#picksplit#####
##############GiST####################

same

################################ (##
###############)

#####SQL#############

```
CREATE OR REPLACE FUNCTION my_same(storage_type, storage_type,
 internal)
RETURNS internal
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

####C##################################

```
PG_FUNCTION_INFO_V1(my_same);

Datum
my_same(PG_FUNCTION_ARGS)
{
    prefix_range *v1 = PG_GETARG_PREFIX_RANGE_P(0);
    prefix_range *v2 = PG_GETARG_PREFIX_RANGE_P(1);
    bool       *result = (bool *) PG_GETARG_POINTER(2);

    *result = my_eq(v1, v2);
    PG_RETURN_POINTER(result);
}
```

##########same################## ######3############################## ####
##############################

distance

########p#######q## ##################
######################### ##
#
##

#####SQL####################

```
CREATE OR REPLACE FUNCTION my_distance(internal, data_type, smallint,
 oid, internal)
RETURNS float8
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

C##############################

```
PG_FUNCTION_INFO_V1(my_distance);

Datum
my_distance(PG_FUNCTION_ARGS)
{
    GISTENTRY  *entry = (GISTENTRY *) PG_GETARG_POINTER(0);
    data_type  *query = PG_GETARG_DATA_TYPE_P(1);
    StrategyNumber strategy = (StrategyNumber) PG_GETARG_UINT16(2);
    /* Oid subtype = PG_GETARG_OID(3); */
    /* bool *recheck = (bool *) PG_GETARG_POINTER(4); */
    data_type  *key = DatumGetDataType(entry->key);
    double      retval;

    /*

     * strategy#key###query##################
     */

    PG_RETURN_FLOAT8(retval);
}
```

distance######consistent##########

########################
################## ## ########
##########*recheck############## ######################################
######## ##

##############*recheck = true##################float8###float4#
########################
################################ ######################################
#############float8############ ################NULL#################
##distance######################

fetch

##################
#######################

#####SQL####################

```
CREATE OR REPLACE FUNCTION my_fetch(internal)
RETURNS internal
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

###GISTENTRY########### ###############key#######NULL################
######GISTENTRY######key########################## opclass########
################fetch#################### ###opclass##########fetch#####
####### ########no-op########

C##############################

```
PG_FUNCTION_INFO_V1(my_fetch);

Datum
my_fetch(PG_FUNCTION_ARGS)
{
    GISTENTRY  *entry = (GISTENTRY *) PG_GETARG_POINTER(0);
    input_data_type *in = DatumGetPointer(entry->key);
    fetched_data_type *fetched_data;
    GISTENTRY  *retval;

    retval = palloc(sizeof(GISTENTRY));
    fetched_data = palloc(sizeof(fetched_data_type));

    /*

     * fetched_data#################
     */


    /* fetched_data####*retval####### */
    gistentryinit(*retval, PointerGetDatum(converted_datum),
                  entry->rel, entry->page, entry->offset, FALSE);

    PG_RETURN_POINTER(retval);
}
```

compress##fetch#
#############

options

```
####################################

###SQL##############


CREATE OR REPLACE FUNCTION my_options(internal)
RETURNS void
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

```
####local_relopts#################################################
###PG_HAS_OPCLASS_OPTIONS()#PG_GET_OPCLASS_OPTIONS()##############
########
```

my_options()#################################

```
typedef enum MyEnumType
{
    MY_ENUM_ON,
    MY_ENUM_OFF,
    MY_ENUM_AUTO
} MyEnumType;

typedef struct
{

    int32   vl_len_;    /* varlena###(########!) */
    int     int_param;  /* ####### */
    double  real_param; /* ####### */
    MyEnumType enum_param; /* enum##### */
    int     str_param;  /* ######## */
} MyOptionsStruct;
```

```
/* enum####### */
static relopt_enum_elt_def myEnumValues[] =
{
    {"on", MY_ENUM_ON},
    {"off", MY_ENUM_OFF},
    {"auto", MY_ENUM_AUTO},

    {(const char *) NULL}   /* ###### */
};

static char *str_param_default = "default";

/*

 * #######: ####8##################
 */
static void
validate_my_string_relopt(const char *value)
{
    if (strlen(value) > 8)
        ereport(ERROR,
                (errcode(ERRCODE_INVALID_PARAMETER_VALUE),
                 errmsg("str_param must be at most 8 bytes")));
}

/*

 * ####: #############
 */
static Size
fill_my_string_relopt(const char *value, void *ptr)
{
    char   *tmp = str_tolower(value, strlen(value),
 DEFAULT_COLLATION_OID);
    int     len = strlen(tmp);

    if (ptr)
        strcpy((char *) ptr, tmp);

    pfree(tmp);
    return len + 1;
}

PG_FUNCTION_INFO_V1(my_options);

Datum
my_options(PG_FUNCTION_ARGS)
{
    local_relopts *relopts = (local_relopts *) PG_GETARG_POINTER(0);

    init_local_reloptions(relopts, sizeof(MyOptionsStruct));
    add_local_int_reloption(relopts, "int_param", "integer parameter",
                            100, 0, 1000000,
                            offsetof(MyOptionsStruct, int_param));
    add_local_real_reloption(relopts, "real_param", "real parameter",
                             1.0, 0.0, 1000000.0,
                             offsetof(MyOptionsStruct, real_param));
    add_local_enum_reloption(relopts, "enum_param", "enum parameter",
                             myEnumValues, MY_ENUM_ON,
                             "Valid values are: \"on\", \"off\" and
 \"auto\".",
                             offsetof(MyOptionsStruct, enum_param));
```

```
    add_local_string_reloption(relopts, "str_param", "string
 parameter",
                               str_param_default,
                               &validate_my_string_relopt,
                               &fill_my_string_relopt,
                               offsetof(MyOptionsStruct, str_param));

    PG_RETURN_VOID();
}

PG_FUNCTION_INFO_V1(my_compress);

Datum
my_compress(PG_FUNCTION_ARGS)
{
    int     int_param = 100;
    double  real_param = 1.0;
    MyEnumType enum_param = MY_ENUM_ON;
    char   *str_param = str_param_default;

    /*

     * ###########'options'##########options##########
     * #############'options'######################
     * ###############options##############
     */
    if (PG_HAS_OPCLASS_OPTIONS())
    {
        MyOptionsStruct *options = (MyOptionsStruct *)
 PG_GET_OPCLASS_OPTIONS();

        int_param = options->int_param;
        real_param = options->real_param;
        enum_param = options->enum_param;
        str_param = GET_STRING_RELOPTION(options, str_param);
    }


    /* ############ */
}
```

GiST##
#####gtsvector_options()#########

sortsupport

```
################################ CREATE INDEX#REINDEX########### #########
##############################################
```

sortsupport########### ########CREATE INDEX######penalty##
#picksplit##

#####SQL#############

```
CREATE OR REPLACE FUNCTION my_sortsupport(internal)
RETURNS void
AS 'MODULE_PATHNAME'
LANGUAGE C STRICT;
```

###SortSupport########### ###########comparator################# ####
####3#########Datum#2##SortSupport########### Datum############2#####
###################compress############### ###API#src/include/utils/
sortsupport.h#########

```
####C################################


PG_FUNCTION_INFO_V1(my_sortsupport);

static int
my_fastcmp(Datum x, Datum y, SortSupport ssup)
{

  /* ########z########x#y########## */

  int z1 = ComputeSpatialCode(x);
  int z2 = ComputeSpatialCode(y);

  return z1 == z2 ? 0 : z1 > z2 ? 1 : -1;
}

Datum
my_sortsupport(PG_FUNCTION_ARGS)
{
  SortSupport ssup = (SortSupport) PG_GETARG_POINTER(0);

  ssup->comparator = my_fastcmp;
  PG_RETURN_VOID();
}
```

####GiST################################ ###CurrentMemoryContext######## ############ ####palloc######pfree####################### ################### ################################ #######fcinfo->flinfo->fn_mcxt########### ##############fcinfo->flinfo->fn_extra########### ###################### GiST###################################### fn_extra#############pfree######## ###### ###################

64.2.4.

64.2.4.1. GiST#########

GiST#########################1#1######### ################################## ####################I/O################################## PostgreSQL#GiST## ###############2################################

################################64.2.3###########sortsupport################## ##################################

#################################### ################################I/O####### ########## ## ##

#########penalty############################CPU########## ################## ############################ ##################################### ########## ##########################

##############################effective_cache_size######GiST###################### ### #######CREATE INDEX#####buffering######################## ########### ####### ##

64.2.5.

PostgreSQL#########GiST################################## ################ #tsvector#tsquery###########################R-Tree########### #src/backend/access/gist/gistproc.c########## ###contrib########GiST####### ####

`btree_gist`

: ############B-tree####

`cube`

: ##############

`hstore`

: ####################

`intarray`

: int4##1######RD-Tree

`ltree`

: ##############

`pg_trgm`

: ####################

`seg`

: ################

64.3. SP-GiST######

64.3.1.

SP-GiST########(Space-Partitioned)GiST######## SP-GiST####################kd#####(####)############################## ##

##I/O#######################################SP-GiST##

GiST#####SP-GiST###

############Purdue###SP-GiST###########WEB###[4]######## PostgreSQL##SP-GiST#######Teodor Sigaev#Oleg Bartunov####################WEB###[5]######

64.3.2.

PostgreSQL################# 64.2#####SP-GiST##############

#64.2 ####SP-GiST######

##	############	#######
box_ops	<< (box,box)	<-> (box,point)
	&< (box,box)	
	&> (box,box)	
	>> (box,box)	

[4] https://www.cs.purdue.edu/spgist/
[5] http://www.sai.msu.su/~megera/wiki/spgist_dev

##	###########	#######
	<@ (box,box)	
	@> (box,box)	
	~= (box,box)	
	&& (box,box)	
	<<| (box,box)	
	&<| (box,box)	
	|&> (box,box)	
	|>> (box,box)	
inet_ops	<< (inet,inet)	
	<<= (inet,inet)	
	>> (inet,inet)	
	>>= (inet,inet)	
	= (inet,inet)	
	<> (inet,inet)	
	< (inet,inet)	
	<= (inet,inet)	
	> (inet,inet)	
	>= (inet,inet)	
	&& (inet,inet)	
kd_point_ops	|>> (point,point)	<-> (point,point)
	<< (point,point)	
	>> (point,point)	
	<<| (point,point)	
	~= (point,point)	
	<@ (point,box)	
poly_ops	<< (polygon,polygon)	<-> (polygon,point)
	&< (polygon,polygon)	
	&> (polygon,polygon)	
	>> (polygon,polygon)	
	<@ (polygon,polygon)	
	@> (polygon,polygon)	
	~= (polygon,polygon)	
	&& (polygon,polygon)	
	<<| (polygon,polygon)	
	&<| (polygon,polygon)	
	|>> (polygon,polygon)	
	|&> (polygon,polygon)	
quad_point_ops	|>> (point,point)	<-> (point,point)
	<< (point,point)	
	>> (point,point)	
	<<| (point,point)	

##	###########	#######
	~= (point,point)	
	<@ (point,box)	
range_ops	= (anyrange,anyrange)	
	&& (anyrange,anyrange)	
	@> (anyrange,anyelement)	
	@> (anyrange,anyrange)	
	<@ (anyrange,anyrange)	
	<< (anyrange,anyrange)	
	>> (anyrange,anyrange)	
	&< (anyrange,anyrange)	
	&> (anyrange,anyrange)	
	-|- (anyrange,anyrange)	
text_ops	= (text,text)	
	< (text,text)	
	<= (text,text)	
	> (text,text)	
	>= (text,text)	
	~<~ (text,text)	
	~<=~ (text,text)	
	~>=~ (text,text)	
	~>~ (text,text)	
	^@ (text,text)	

point##2###########quad_point_ops######## kd_point_ops#################### ##

quad_point_ops#kd_point_ops#poly_ops######<->########################## ################k##(k-NN)########

64.3.3.

SP-GiST## SP-GiST############################## ####################

SP-GiST## ## ### #######

SP-GiST#######INCLUDE################################ INCLUDE##SP-GiST#####
###########################

########################### ###########1##################################
##
###
64.3.4.2#######
##############################
###################4#######
############### ################################4#################

######################(##)######################SP-GiST######################
################### ##
##########

> ##
>
> SP-GiST#######null################ SP-GiST###############null############
> ####################################null##################################
> ####### (SP-GiST##########NULL#########################) ###########
> ##NULL###########

SP-GiST##################################5##################2###### 5#########
#2##internal##################1##################C############2############
####C########### 4################################void#######
##leaf_consistent#boolean######## ################################ ######
################################## ######6#######compress##################
#datum########################## ######7#######options###################C
#####internal#########void######

5######################

config

> ###################OID##########################
>
> ###SQL#############
>
> CREATE FUNCTION my_config(internal, internal) RETURNS void ...
>
> 1######C#spgConfigIn######################## 2######C#spgConfigOut#####
> ####################
>
> ```
> typedef struct spgConfigIn
> {
>
> Oid         attType;        /* ############### */
> } spgConfigIn;
>
> typedef struct spgConfigOut
> {
>
> Oid         prefixType;     /* ############## */
> Oid         labelType;      /* ################## */
> Oid         leafType;       /* ############ */
> bool        canReturnData;  /* ################### */
> bool        longValuesOK;   /* #######1############# */
> ```

```
} spgConfigOut;

attType############################## ##################################
######

################prefixType#VOIDOID############   ####################
##labelType#VOIDOID############       ################################
##canReturnData#true#####  attType#####################################
##longValuesOK#true####(64.3.4.1##)#

leaf Type########opckeytype#####################################
(opckeytype#0#############################################################
#######) #########config#####leaf Type#################################
######################### ###leaf Type########(0)########opckeytype####
######################

attType#leaf Type#################compress#############          ###
#compress################attType##leaf Type############

choose

############################

###SQL#############


CREATE FUNCTION my_choose(internal, internal) RETURNS void ...

1######C#spgChooseIn######################## 2######C#spgChooseOut#####
####################


typedef struct spgChooseIn
{

    Datum       datum;          /* ################ */
    Datum       leafDatum;      /* ################# */
    int         level;          /* (0#####)###### */


    /* ############## */
    bool        allTheSame;     /* ####all-the-same########### */
    bool        hasPrefix;      /* ############## */
    Datum       prefixDatum;    /* ############ */
    int         nNodes;         /* ########### */
    Datum      *nodeLabels;     /* #########(####NULL) */
} spgChooseIn;

typedef enum spgChooseResultType
{

    spgMatchNode = 1,           /* ########## */
    spgAddNode,                 /* ############## */
    spgSplitTuple               /* ##########(##########) */
} spgChooseResultType;

typedef struct spgChooseOut
{

    spgChooseResultType resultType;     /* ############# */
    union
    {

        struct                  /* spgMatchNode### */
```

```
        {

            int         nodeN;      /* #########(0#########) */
            int         levelAdd;   /* ############ */
            Datum       restDatum;  /* ######### */
        }           matchNode;

        struct                  /* spgAddNode### */
        {

            Datum       nodeLabel;  /* ########## */
            int         nodeN;      /* ######(0#########) */
        }           addNode;

        struct                  /* spgSplitTuple### */
        {

            /* #####1############################ */
            bool        prefixHasPrefix;    /* ########### */
            Datum       prefixPrefixDatum;  /* ######### */
            int         prefixNNodes;       /* #### */
            Datum      *prefixNodeLabels;   /* #####(########NULL) */
            int         childNodeN;         /* ############## */


            /* ################################ */
            bool        postfixHasPrefix;   /* ########### */
            Datum       postfixPrefixDatum; /* ######### */
        }           splitTuple;
    }           result;
} spgChooseOut;
```

datum############spgConfigIn.attType########
leafDatum#spgConfigOut.leafType################compress########
#datum#########compress#########datum#######
leafDatum##choose#picksplit#################################### #####
##############leafDatum############################ level########0####
################## ##################################allTheSame#true####
(64.3.4.3##)# #################hasPrefix#true##### #####prefixDatum#######
nNodes################nodeLabels#######################NULL######

choose################1###
###

############1##########resultType#spgMatchNode####### nodeN#######
#######(0###)######### levelAdd####################level##############
###################0######## restDatum#################################
##datum#######################leafDatum####################

######################resultType#spgAddNode####### nodeLabel########
#############nodeN####################(0###)######### ###########choose###
##############################spgMatchNode############

#####################resultType#spgSplitTuple####### #################
##
prefixHasPrefix##############################prefixPrefixDatum#######
########   ##
prefixNNodes###################prefixNodeLabels###########palloc###
###################NULL####### ##
################################ childNodeN#########################(0###)
#########   postfixHasPrefix############################
#postfixPrefixDatum############# ####################################

####################2######(#####)#####################################
######################choose########### #######spgSplitTuple#####
#################spgAddNode########## #####choose#spgMatchNode#######
###############

picksplit

##################################

###SQL#############

```
CREATE FUNCTION my_picksplit(internal, internal) RETURNS void ...
```

1######C#spgPickSplitIn######################## 2######C#spgPickSplitOut########################

```
typedef struct spgPickSplitIn
{

    int         nTuples;        /* ######## */
    Datum      *datums;         /* #####(##nTuples###) */
    int         level;          /* (0#####)###### */
} spgPickSplitIn;

typedef struct spgPickSplitOut
{

    bool        hasPrefix;      /* ################ */
    Datum       prefixDatum;    /* ########## */


    int         nNodes;         /* ############# */
    Datum      *nodeLabels;     /* #####(########NULL) */


    int        *mapTuplesToNodes;   /* ################### */
    Datum      *leafTupleDatums;    /* ###################### */
} spgPickSplitOut;
```

nTuples################# datums#spgConfigOut.leafType###############
level##

hasPrefix###########################prefixDatum############# nNodes##
##################nodeLabels###########################NULL#######
mapTuplesToNodes#######################(0###)############
leafTupleDatums#########################(############################
######datums#######)# picksplit##
##nodeLabels#mapTuplesToNodes#leafTupleDatums######palloc##########
##########

2################picksplit######2#####################################
##################################### ####picksplit#####################
########SP-GiST###
######### ###########################allTheSame######### choose##
#inner_consistent###64.3.4.3####
######

config###longValuesOK#true#####1#####################picksplit#1#####
############ ## ######1########
######################## ######64.3.4.1##########

inner_consistent

##############(#)########

###SQL##############

```
CREATE FUNCTION my_inner_consistent(internal, internal) RETURNS
 void ...
```

1######C#spgInnerConsistentIn####################### 2######C#spgInnerConsistentOut#######################

```
typedef struct spgInnerConsistentIn
{

    ScanKey     scankeys;       /* ############ */
    ScanKey     orderbys;       /* ################ */
    int         nkeys;          /* scankeys##### */
    int         norderbys;      /* orderbys##### */


    Datum       reconstructedValue;     /* ######### */
    void       *traversalValue; /* ############# */
    MemoryContext traversalMemoryContext;   /* ############# */
    int         level;          /* (0#####)###### */
    bool        returnData;     /* ################# */


    /* ############## */
    bool        allTheSame;     /* ####all-the-same########### */
    bool        hasPrefix;      /* ########### */
    Datum       prefixDatum;    /* ############ */
    int         nNodes;         /* ############ */
    Datum      *nodeLabels;     /* #########(####NULL) */
} spgInnerConsistentIn;

typedef struct spgInnerConsistentOut
{

    int         nNodes;         /* ########### */
    int        *nodeNumbers;    /* ################ */
    int        *levelAdds;      /* ################ */
    Datum      *reconstructedValues;    /* ########### */
    void      **traversalValues;        /* ############# */
    double    **distances;              /* ###### */
} spgInnerConsistentOut;
```

##scankeys####nkeys################## ######AND#################### #################### (nkeys = 0 ################################ #########) ###consistent#################sk_strategy###sk_argument###### ## ########NULL########### #sk_flags#################SP-GiST####################### ##orderbys## ##norderbys##(#####)################## reconstructedValue############ #######################inner_consistent#############(Datum) 0#### ## traversalValue############inner_consistent####################### ##########NULL### traversalMemoryContext############################## level####################0########### returnData####################### ##true#########config###canReturnData################### ##########all-the-same#############allTheSame## ###########################64.3.4.3##### #######################hasPrefix### ##########prefixDatum####### nNodes##################nodeLabels######## #######################NULL###

nNodes##################################nodeNumbers#################
####### ##levelAdds#####
(##)
#######################reconstructedValues################
##reconstructedValues#NULL####### #######spgConfigOut.leafType###
(##################################leafType###typlen#typbyval#
############) #############orderbys######distances#############(#####
##############)# ######NULL####### ##############################
##traversalValues######################1########### #######
#traversalValues#NULL####### inner_consistent################
#nodeNumbers#levelAdds#distances#reconstructedValues#traversalValues#
######palloc#################### ####traversalValues#############
#traversalMemoryContext########## ########1##palloc##############

leaf_consistent

##################true######

###SQL#############

```
CREATE FUNCTION my_leaf_consistent(internal, internal) RETURNS bool ...
```

1######C#spgLeafConsistentIn####################### 2######C#spgLeafConsistentOut#######################

```
typedef struct spgLeafConsistentIn
{

    ScanKey     scankeys;       /* ############ */
    ScanKey     orderbys;       /* ################ */
    int         nkeys;          /* scankeys##### */
    int         norderbys;      /* orderbys##### */


    Datum       reconstructedValue;     /* ######### */
    void       *traversalValue; /* ############# */
    int         level;          /* (0#####)###### */
    bool        returnData;     /* ################# */


    Datum       leafDatum;      /* ########## */
} spgLeafConsistentIn;

typedef struct spgLeafConsistentOut
{

    Datum       leafValue;        /* ################# */
    bool        recheck;          /* ##################true */
    bool        recheckDistances; /* #################true */
    double     *distances;        /* ###### */
} spgLeafConsistentOut;
```

##scankeys####nkeys################## ######AND##################
####################### (nkeys#0#######################################
##) ###consistent#################sk_strategy###sk_argument#############
#######NULL###########sk_flags#
################SP-GiST####################### ##orderbys###
#norderbys#################### reconstructedValue################
##################inner_consistent#############(Datum) 0####
traversalValue############inner_consistent######################
##########NULL### level######################0########## returnData#####

```
    ####################true#########config###canReturnData##############
    ### leafDatum################spgConfigOut.leafType#######

    ########################true###########false######              true##
    ##returnData#true#####leafValue############################
    (spgConfigIn.attType##)############## ################################
    ########################recheck#true##############         ###########
    ##orderbys######distances############## #######NULL####### ###########
    1##########recheckDistances#true####### ################################
    #################

####################

Datum compress(Datum in)

    #######################################       spgConfigIn.attType#######
    ##spgConfigOut.leafType######## ##########TOAST############

    ##:   compress####################   ########compress#################
    #scankeys###########

options

    ####################################

    ###SQL#############


    CREATE OR REPLACE FUNCTION my_options(internal)
    RETURNS void
    AS 'MODULE_PATHNAME'
    LANGUAGE C STRICT;

    ####local_relopts################################################ ######
    ###PG_HAS_OPCLASS_OPTIONS()#PG_GET_OPCLASS_OPTIONS()##############
    ########

    SP-GiST##########################################

SP-GiST##############################################################
#CurrentMemoryContext######### ######palloc##########pfree##################
####### (config############################## #######config################
##########################)

#########################################PG_GET_COLLATION()################
##########
```

64.3.4.

```
######SP-GiST#######################################################
```

64.3.4.1. SP-GiST###

```
####################1###########(######8kB)############## ####################
##############################################################################
##################################### ##################################
#longValuesOK############ ########SP-GiST##################################
############

############################################################## ######1#########
##################################
```

(#######
###) ################1#####
############################### ################################2############
############ #######picksplit#################SP-GiST#####64.3.4.3###########
###########

longValuesOK#######SP-GiST##
#############1##################### ##############################choose####
######10##########################SP-GiST############

64.3.4.2. #########SP-GiST

#####################4#######
#####4############ ## #########
#(####################)####picksplit###nodeLabels#####NULL##########
#choose###spgSplitTuple######prefixNodeLabels#####NULL########## #####
####choose#####inner_consistent###########nodeLabels#NULL###### #######
##

#########################choose#spgAddNode################################
############

64.3.4.3. #All-the-Same######

picksplit#############2##################SP-GiST##########picksplit######
############ ##picksplit########
#####(###)##################################### #######allTheSame########
##choose#####inner_consistent##

allTheSame#######choose#spgMatchNode#################################
####### ###########nodeN#######(###########)####1##############
choose#spgAddNode## ################
#####spgSplitTuple#################

allTheSame#########################inner_consistent####################
###################1################## ##################################
#########inner_consistent####################################

64.3.5.

PostgreSQL############## 64.2#######SP-GiST########################## #######
#src/backend/access/spgist/#src/backend/utils/adt/##########

64.4. GIN######

64.4.1.

GIN###########(Generalized Inverted Index)###### GIN##########################(1)##
################(2)## #########
#######################

GIN#########################

GIN#######(###########(posting list))################ ####################ID######
#########################ID######################### ################## ####
GIN##################################

GIN########GIN## ##################### ########### ## ###################

GIN####1## ### GiST############

PostgreSQL####GIN#######Teodor Sigaev#Oleg Bartunov########### GIN#########web### #[6]##############

64.4.2.

PostgreSQL######## 64.3###GIN########### (##F###################GIN########### #######)

#64.3 ####GIN######

##	###########
array_ops	&& (anyarray,anyarray)
	@> (anyarray,anyarray)
	<@ (anyarray,anyarray)
	= (anyarray,anyarray)
jsonb_ops	@> (jsonb,jsonb)
	@? (jsonb,jsonpath)
	@@ (jsonb,jsonpath)
	? (jsonb,text)
	?| (jsonb,text[])
	?& (jsonb,text[])
jsonb_path_ops	@> (jsonb,jsonb)
	@? (jsonb,jsonpath)
	@@ (jsonb,jsonpath)
tsvector_ops	@@ (tsvector,tsquery)

jsonb##2###########jsonb_ops######## jsonb_path_ops#################### ########################## ###8.14.4#########

64.4.3.

GIN#################### ## GIN##### ##############################

GIN## ############################ ########################## #####GIN#######################################

GIN#########################2######

Datum *extractValue(Datum itemValue, int32 *nkeys, bool **nullFlags)

> ################palloc################### #########*nkeys############## ###### ##NULL###########*nkeys##bool####palloc###########*nullFlags######### #NULL############ #######NULL#####*nullFlags#NULL################## # ################NULL###########

[6] http://www.sai.msu.su/~megera/wiki/Gin

```
Datum *extractQuery(Datum query, int32 *nkeys, StrategyNumber n, bool
**pmatch, Pointer **extra_data, bool **nullFlags, int32 *searchMode)

    ################palloc################## ####query#################### #####
    ############ n###################36.16.2##### extractQuery######query####
    ##################################n############ ########*nkeys########
    ##### ########NULL###############*nkeys##bool####palloc######*nullFlags##
    ################NULL############ ########NULL###*nullFlags#NULL########
    ############ query##############NULL##########

    searchMode#######       #####extractQuery############################
    *searchMode#GIN_SEARCH_MODE_DEFAULT################################
    ###########################
    *searchMode#GIN_SEARCH_MODE_INCLUDE_EMPTY##########################
    ############################        ##################################
    *searchMode#GIN_SEARCH_MODE_ALL####################################
    ####NULL################## ############################################
    ########## ################################ ###########################GIN
    ###################### ###################access/gin.h##########

    pmatch########################## #######extractQuery#*nkeys##bool#####
    ##########*pmatch############# ########################################
    ############ *pmatch#NULL##########GIN################ ########NULL####
    ########################################

    extra_data##extractQuery#consistent#comparePartial################
    ##############             #######extractQuery#*nkeys##################
    #*extra_data#################################### ########NULL########
    ################################## ##*extra_data############consistent#
    ###########comparePartial##########

#################################################     ###2###########     2#
#consistent###3##triConsistent####            triConsistent###########
##triConsistent############ ####2######################################### 2
#########################################################################

bool consistent(bool check[], StrategyNumber n, Datum query, int32
nkeys, Pointer extra_data[], bool *recheck, Datum queryKeys[], bool
nullFlags[])

    ##################n###################recheck############################
    ## GIN################################################## #############
    ######################################### check#####nkeys######query###
    ###########extractQuery############# ############################check##
    ######## ####(check[i] == true)####extractQuery######i###################
    ### ##query####consistent########################## ####queryKeys[]##
    #nullFlags[]####extractQuery######### extra_data#extractQuery######
    ##############NULL###

    extractQuery#queryKeys[]##NULL################NULL##########
    #check[]###### ####check[]####IS NOT DISTINCT FROM######
    ## consistent###########NULL#####################nullFlags[]########
    ######

    ######*recheck##############################true###############false######
    ####### ####false############################## true######*recheck#false######
    ############################ true######*recheck#true######################
    ################ ##########################################################
    ########
```

GinTernaryValue triConsistent(GinTernaryValue check[], StrategyNumber n, Datum query, int32 nkeys, Pointer extra_data[], Datum queryKeys[], bool nullFlags[])

triConsistent#consistent#######check###################3#########
##GIN_TRUE#GIN_FALSE#GIN_MAYBE### GIN_FALSE#GIN_TRUE#############
##GIN_MAYBE#################### GIN_MAYBE##############################
#########################GIN_TRUE####### ####GIN_MAYBE#################
#############GIN_FALSE############ ###GIN_MAYBE#######################
################################GIN_MAYBE############

check#####GIN_MAYBE######GIN_MAYBE#######consistent###recheck######
#########

####GIN#################################### ################################

int compare(Datum a, Datum b)

############################0######0####0############ ##########2########
#################### NULL####################

###########compare##############GIN#######################btree###########
######### btree#######################GIN####################################
################ ####(array_ops####)##GIN#########################

########GIN#######################

int comparePartial(Datum partial_key, Datum key, StrategyNumber n, Pointer extra_data)

########################### ############### ##############################
###################### ############################ ####################
########################### ##
###########n######## ##extra_data#extractQuery#######################
#NULL### NULL####################

void options(local_relopts *relopts)

####################################

options####local_relopts##
########PG_HAS_OPCLASS_OPTIONS()#PG_GET_OPCLASS_OPTIONS()######
###############

#####################GIN###

#############################comparePartial################ ###
#extractQuery################pmatch################## #######64.4.4.2#######
##

#####Datum######################### extractValue##########################
#########STORAGE########### extractQuery#consistent###triConsistent###
##query#################################### ################################
################ ########3########SQL##############################query##
##################################

64.4.4.

GIN###############B-tree############ ##
##################B-tree################(posting tree)############################
##################################(posting list)##### # 64.1#GIN##################
####

PostgreSQL 9.1##NULL########################## #############NULL##NULL## #extractValue## ########################## ######

#######GIN#############################B-tree############### ################### #######

#64.1 GIN###

64.4.4.1. GIN######

1###GIN#################### ################################## GIN## ############################## ## ##gin_clean_pending_list####################(pending list)#gin_pending_list_limit# ##GIN############# ############ #####################GIN#################### ############################## ##########################

##

#####################GIN#########fastupdate################################ ####### ###CREATE INDEX#########

64.4.4.2.

GIN########################## ## #compare## extractQuery############## ########################pmatch########### #######comparePartial########## #### comparePartial## ############################

64.4.5. GIN###

#####

> ############################GIN######################## ################### ##GIN#################################

GIN##`fastupdate`################################## #64.4.4.1########### ##### ##################################

maintenance_work_mem

GIN############`maintenance_work_mem`############ ######################### ########

gin_pending_list_limit

`fastupdate`#######GIN##########################
`#gin_pending_list_limit`######################### ######################## ## ############## `##gin_pending_list_limit`## ###### ##

`gin_pending_list_limit`##################GIN#####################GIN## ################## ##################GIN################################

gin_fuzzy_search_limit

GIN##################################PostgreSQL########### ################ ######### ## #### ## ######################### ########

#########################GIN####################
`##gin_fuzzy_search_limit`############ #########0############## #0####### ##

###

##########5000##20000#############

64.4.6.

GIN############################ ###`extractValue`#NULL######################### ################################`extractQuery`########NULL################ ######################### ####NULL##################NULL#############

64.4.7.

PostgreSQL######### 64.3####GIN############ ###`contrib`#######GIN###############

`btree_gin`

#############B-tree#####

`hstore`

######################

`intarray`

`int[]`#########

`pg_trgm`

####################

64.5. BRIN######

64.5.1.

BRIN################(Block Range Index)##### BRIN##

BRIN########*(block range)*###########(page range)########### *#######(block range)*###

BRIN###BRIN###BRIN#################################BRIN###

BRIN###

#######################pages_per_range##pages_per_range###pages_per_range###(###########################)###

64.5.1.1.

###

###

##autovacuum###autosummarize###

###search_path#####pg_####,pg_temp#########

#################brin_summarize_new_values(regclass)
###
#brin_summarize_range(regclass, bigint)

###autovacuum##

```
LOG:  request for BRIN range summarization for index "brin_wi_idx" page
 128 was not recorded
```

###

########brin_desummarize_range(regclass, bigint)##

64.5.2.

PostgreSQL######### 64.4#####BRIN#############

minmax## *inclusion*## *bloom*## *minmax-multi*###

#64.4 ####BRIN######

##	###########
bit_minmax_ops	= (bit,bit)
	< (bit,bit)
	> (bit,bit)
	<= (bit,bit)
	>= (bit,bit)
box_inclusion_ops	@> (box,point)
	<< (box,box)
	&< (box,box)
	&> (box,box)
	>> (box,box)
	<@ (box,box)
	@> (box,box)
	~= (box,box)
	&& (box,box)
	<<| (box,box)
	&<| (box,box)
	|&> (box,box)
	|>> (box,box)
bpchar_bloom_ops	= (character,character)
bpchar_minmax_ops	= (character,character)
	< (character,character)
	<= (character,character)
	> (character,character)
	>= (character,character)
bytea_bloom_ops	= (bytea,bytea)
bytea_minmax_ops	= (bytea,bytea)
	< (bytea,bytea)
	<= (bytea,bytea)
	> (bytea,bytea)
	>= (bytea,bytea)
char_bloom_ops	= ("char","char")
char_minmax_ops	= ("char","char")
	< ("char","char")
	<= ("char","char")
	> ("char","char")
	>= ("char","char")
date_bloom_ops	= (date,date)
date_minmax_ops	= (date,date)
	< (date,date)
	<= (date,date)

##	############
	> (date,date)
	>= (date,date)
date_minmax_multi_ops	= (date,date)
	< (date,date)
	<= (date,date)
	> (date,date)
	>= (date,date)
float4_bloom_ops	= (float4,float4)
float4_minmax_ops	= (float4,float4)
	< (float4,float4)
	> (float4,float4)
	<= (float4,float4)
	>= (float4,float4)
float4_minmax_multi_ops	= (float4,float4)
	< (float4,float4)
	> (float4,float4)
	<= (float4,float4)
	>= (float4,float4)
float8_bloom_ops	= (float8,float8)
float8_minmax_ops	= (float8,float8)
	< (float8,float8)
	<= (float8,float8)
	> (float8,float8)
	>= (float8,float8)
float8_minmax_multi_ops	= (float8,float8)
	< (float8,float8)
	<= (float8,float8)
	> (float8,float8)
	>= (float8,float8)
inet_inclusion_ops	<< (inet,inet)
	<<= (inet,inet)
	>> (inet,inet)
	>>= (inet,inet)
	= (inet,inet)
	&& (inet,inet)
inet_bloom_ops	= (inet,inet)
inet_minmax_ops	= (inet,inet)
	< (inet,inet)
	<= (inet,inet)
	> (inet,inet)
	>= (inet,inet)

##	############
inet_minmax_multi_ops	= (inet,inet)
	< (inet,inet)
	<= (inet,inet)
	> (inet,inet)
	>= (inet,inet)
int2_bloom_ops	= (int2,int2)
int2_minmax_ops	= (int2,int2)
	< (int2,int2)
	> (int2,int2)
	<= (int2,int2)
	>= (int2,int2)
int2_minmax_multi_ops	= (int2,int2)
	< (int2,int2)
	> (int2,int2)
	<= (int2,int2)
	>= (int2,int2)
int4_bloom_ops	= (int4,int4)
int4_minmax_ops	= (int4,int4)
	< (int4,int4)
	> (int4,int4)
	<= (int4,int4)
	>= (int4,int4)
int4_minmax_multi_ops	= (int4,int4)
	< (int4,int4)
	> (int4,int4)
	<= (int4,int4)
	>= (int4,int4)
int8_bloom_ops	= (bigint,bigint)
int8_minmax_ops	= (bigint,bigint)
	< (bigint,bigint)
	> (bigint,bigint)
	<= (bigint,bigint)
	>= (bigint,bigint)
int8_minmax_multi_ops	= (bigint,bigint)
	< (bigint,bigint)
	> (bigint,bigint)
	<= (bigint,bigint)
	>= (bigint,bigint)
interval_bloom_ops	= (interval,interval)
interval_minmax_ops	= (interval,interval)
	< (interval,interval)

##	############
	<= (interval,interval)
	> (interval,interval)
	>= (interval,interval)
interval_minmax_multi_ops	= (interval,interval)
	< (interval,interval)
	<= (interval,interval)
	> (interval,interval)
	>= (interval,interval)
macaddr_bloom_ops	= (macaddr,macaddr)
macaddr_minmax_ops	= (macaddr,macaddr)
	< (macaddr,macaddr)
	<= (macaddr,macaddr)
	> (macaddr,macaddr)
	>= (macaddr,macaddr)
macaddr_minmax_multi_ops	= (macaddr,macaddr)
	< (macaddr,macaddr)
	<= (macaddr,macaddr)
	> (macaddr,macaddr)
	>= (macaddr,macaddr)
macaddr8_bloom_ops	= (macaddr8,macaddr8)
macaddr8_minmax_ops	= (macaddr8,macaddr8)
	< (macaddr8,macaddr8)
	<= (macaddr8,macaddr8)
	> (macaddr8,macaddr8)
	>= (macaddr8,macaddr8)
macaddr8_minmax_multi_ops	= (macaddr8,macaddr8)
	< (macaddr8,macaddr8)
	<= (macaddr8,macaddr8)
	> (macaddr8,macaddr8)
	>= (macaddr8,macaddr8)
name_bloom_ops	= (name,name)
name_minmax_ops	= (name,name)
	< (name,name)
	<= (name,name)
	> (name,name)
	>= (name,name)
numeric_bloom_ops	= (numeric,numeric)
numeric_minmax_ops	= (numeric,numeric)
	< (numeric,numeric)
	<= (numeric,numeric)
	> (numeric,numeric)

##	############
	>= (numeric,numeric)
numeric_minmax_multi_ops	= (numeric,numeric)
	< (numeric,numeric)
	<= (numeric,numeric)
	> (numeric,numeric)
	>= (numeric,numeric)
oid_bloom_ops	= (oid,oid)
oid_minmax_ops	= (oid,oid)
	< (oid,oid)
	> (oid,oid)
	<= (oid,oid)
	>= (oid,oid)
oid_minmax_multi_ops	= (oid,oid)
	< (oid,oid)
	> (oid,oid)
	<= (oid,oid)
	>= (oid,oid)
pg_lsn_bloom_ops	= (pg_lsn,pg_lsn)
pg_lsn_minmax_ops	= (pg_lsn,pg_lsn)
	< (pg_lsn,pg_lsn)
	> (pg_lsn,pg_lsn)
	<= (pg_lsn,pg_lsn)
	>= (pg_lsn,pg_lsn)
pg_lsn_minmax_multi_ops	= (pg_lsn,pg_lsn)
	< (pg_lsn,pg_lsn)
	> (pg_lsn,pg_lsn)
	<= (pg_lsn,pg_lsn)
	>= (pg_lsn,pg_lsn)
range_inclusion_ops	= (anyrange,anyrange)
	< (anyrange,anyrange)
	<= (anyrange,anyrange)
	>= (anyrange,anyrange)
	> (anyrange,anyrange)
	&& (anyrange,anyrange)
	@> (anyrange,anyelement)
	@> (anyrange,anyrange)
	<@ (anyrange,anyrange)
	<< (anyrange,anyrange)
	>> (anyrange,anyrange)
	&< (anyrange,anyrange)
	&> (anyrange,anyrange)

##	############
	-|- (anyrange,anyrange)
text_bloom_ops	= (text,text)
text_minmax_ops	= (text,text)
	< (text,text)
	<= (text,text)
	> (text,text)
	>= (text,text)
tid_bloom_ops	= (tid,tid)
tid_minmax_ops	= (tid,tid)
	< (tid,tid)
	> (tid,tid)
	<= (tid,tid)
	>= (tid,tid)
tid_minmax_multi_ops	= (tid,tid)
	< (tid,tid)
	> (tid,tid)
	<= (tid,tid)
	>= (tid,tid)
timestamp_bloom_ops	= (timestamp,timestamp)
timestamp_minmax_ops	= (timestamp,timestamp)
	< (timestamp,timestamp)
	<= (timestamp,timestamp)
	> (timestamp,timestamp)
	>= (timestamp,timestamp)
timestamp_minmax_multi_ops	= (timestamp,timestamp)
	< (timestamp,timestamp)
	<= (timestamp,timestamp)
	> (timestamp,timestamp)
	>= (timestamp,timestamp)
timestamptz_bloom_ops	= (timestamptz,timestamptz)
timestamptz_minmax_ops	= (timestamptz,timestamptz)
	< (timestamptz,timestamptz)
	<= (timestamptz,timestamptz)
	> (timestamptz,timestamptz)
	>= (timestamptz,timestamptz)
timestamptz_minmax_multi_ops	= (timestamptz,timestamptz)
	< (timestamptz,timestamptz)
	<= (timestamptz,timestamptz)
	> (timestamptz,timestamptz)
	>= (timestamptz,timestamptz)
time_bloom_ops	= (time,time)

##	###########
time_minmax_ops	= (time,time)
	< (time,time)
	<= (time,time)
	> (time,time)
	>= (time,time)
time_minmax_multi_ops	= (time,time)
	< (time,time)
	<= (time,time)
	> (time,time)
	>= (time,time)
timetz_bloom_ops	= (timetz,timetz)
timetz_minmax_ops	= (timetz,timetz)
	< (timetz,timetz)
	<= (timetz,timetz)
	> (timetz,timetz)
	>= (timetz,timetz)
timetz_minmax_multi_ops	= (timetz,timetz)
	< (timetz,timetz)
	<= (timetz,timetz)
	> (timetz,timetz)
	>= (timetz,timetz)
uuid_bloom_ops	= (uuid,uuid)
uuid_minmax_ops	= (uuid,uuid)
	< (uuid,uuid)
	> (uuid,uuid)
	<= (uuid,uuid)
	>= (uuid,uuid)
uuid_minmax_multi_ops	= (uuid,uuid)
	< (uuid,uuid)
	> (uuid,uuid)
	<= (uuid,uuid)
	>= (uuid,uuid)
varbit_minmax_ops	= (varbit,varbit)
	< (varbit,varbit)
	> (varbit,varbit)
	<= (varbit,varbit)
	>= (varbit,varbit)

64.5.2.1.

################################
bloom######minmax-multi######################

bloom#####################

`n_distinct_per_range`

BRIN##NULL########### ## ##ALTER TABLE#`n_distinct`############## ############################## #NULL############### ##################NULL####################(######## 290#)##############-1############ #######`-0.1`#######NULL######`16`###

`false_positive_rate`

##################BRIN################################## ##0.0001##0.25###### ##############0.01#####1%########

minmax-multi#####################

`values_per_range`

##############BRIN minmax########################### ######################### ####8##256##################32###

64.5.3.

BRIN###
BRIN##############################

BRIN###
################### ####BRIN##

BRIN##########4###################

`BrinOpcInfo *opcInfo(Oid type_oid)`

################################# ####palloc###`BrinOpcInfo`################ BrinOpcInfo###########

```
typedef struct BrinOpcInfo
{
    /* Number of columns stored in an index column of this opclass */
    uint16      oi_nstored;

    /* Opaque pointer for the opclass' private use */
    void       *oi_opaque;

    /* Type cache entries of the stored columns */
    TypeCacheEntry *oi_typcache[FLEXIBLE_ARRAY_MEMBER];
} BrinOpcInfo;
```

`BrinOpcInfo.oi_opaque`## #######

`bool consistent(BrinDesc *bdesc, BrinValues *column, ScanKey *keys, int nkeys)`

#############ScanKey############################## ######################### ### ########################### #######`nkeys`###############

`bool consistent(BrinDesc *bdesc, BrinValues *column, ScanKey key)`

ScanKey############################## ###################### ###consistent##### ##############

`bool addValue(BrinDesc *bdesc, BrinValues *column, Datum newval, bool isnull)`

    ## ## ##########true#######

`bool unionTuples(BrinDesc *bdesc, BrinValues *a, BrinValues *b)`

    2################ #####2## ## 2##############

#######BRIN#######################

`void options(local_relopts *relopts)`

    ##################################

    options####local_relopts## ## ########PG_HAS_OPCLASS_OPTIONS()#PG_GET_OPCLASS_OPTIONS()###### ###############

    ######################BRIN##

########4####################minmax#inclusion#minmax-multi#bloom### ############# ##################### ## ############### ############################## ## ####

###4## ###################### ####### ####################################

64.5##minmax################# ## ########(######)#########

#64.5 Minmax################

#########	######
###### 1	####`brin_minmax_opcinfo()`
###### 2	####`brin_minmax_add_value()`
###### 3	####`brin_minmax_consistent()`
###### 4	####`brin_minmax_union()`
Operator Strategy 1	operator less-than
Operator Strategy 2	operator less-than-or-equal-to
Operator Strategy 3	operator equal-to
Operator Strategy 4	operator greater-than-or-equal-to
Operator Strategy 5	operator greater-than

64.6##inclusion################# ## ########################## ################### ############### ###############

#64.6 Inclusion################

#########	######	###
###### 1	####`brin_inclusion_opcinfo()`	

#########	######	###
###### 2	####brin_inclusion_add_value()	
###### 3	####brin_inclusion_consistent()	
###### 4	####brin_inclusion_union()	
###### 11	2############	
###### 12	2#########################	
###### 13	############################	
###### 14	###################	
Operator Strategy 1	operator left-of	Operator Strategy 4
Operator Strategy 2	operator does-not-extend-to-the-right-of	Operator Strategy 5
Operator Strategy 3	operator overlaps	
Operator Strategy 4	operator does-not-extend-to-the-left-of	Operator Strategy 1
Operator Strategy 5	operator right-of	Operator Strategy 2
Operator Strategy 6, 18	operator same-as-or-equal-to	Operator Strategy 7
Operator Strategy 7, 16, 24, 25	operator contains-or-equal-to	
Operator Strategy 8, 26, 27	operator is-contained-by-or-equal-to	Operator Strategy 3
Operator Strategy 9	operator does-not-extend-above	Operator Strategy 11
Operator Strategy 10	operator is-below	Operator Strategy 12
Operator Strategy 11	operator is-above	Operator Strategy 9
Operator Strategy 12	operator does-not-extend-below	Operator Strategy 10
Operator Strategy 20	operator less-than	Operator Strategy 5
Operator Strategy 21	operator less-than-or-equal-to	Operator Strategy 5
Operator Strategy 22	operator greater-than	Operator Strategy 1
Operator Strategy 23	operator greater-than-or-equal-to	Operator Strategy 1

########1##10##BRIN###############SQL########11######## ######11############## ############ ####################2#################### ##STORAGE############## ##inclusion ######################## #########STORAGE####################

########12#14################################ ########12########################## #################### ########14######################## ########13################ ################################## BRIN## ##########################

64.7###bloom########### ############# ##########(##########)#########

#64.7 Bloom####################

#########	######
########## 1	####brin_bloom_opcinfo()
########## 2	####brin_bloom_add_value()
########## 3	####brin_bloom_consistent()
########## 4	####brin_bloom_union()
########## 5	####brin_bloom_options()

#########	######
########## 11	##############
Operator Strategy 1	operator equal-to

###########1##10##BRIN################SQL#######11######### ######11########## ################ #################1#######################

minmax-multi##################################minmax######################### minmax##################################minmax-multi############################## #### # 64.8##################minmax-multi######################### ##########(###### ####)#########

#64.8 minmax-multi####################

#########	######
########## 1	####`brin_minmax_multi_opcinfo()`
########## 2	####`brin_minmax_multi_add_value()`
########## 3	### #`brin_minmax_multi_consistent()`
########## 4	####`brin_minmax_multi_union()`
########## 5	####`brin_minmax_multi_options()`
########## 11	2########(#####)#######
Operator Strategy 1	operator less-than
Operator Strategy 2	operator less-than-or-equal-to
Operator Strategy 3	operator equal-to
Operator Strategy 4	operator greater-than-or-equal-to
Operator Strategy 5	operator greater-than

minmax#inclusion############################ ############################ minmax# ### ################################### #### # 64.6#######inclusion## ## ##############`STORAGE`## ########### minmax#####`float4_minmax_ops`#inclusion#####`box_inclusion_ops`## ########

64.6.

64.6.1.

PostgreSQL## ################### ############################### ## #############

##

###########=#########################WHERE############################

##############4########################### #####UUID#URL########################### #######B-tree################### ############################(lossy)######## ###### ######################################

################################SELECT#UPDATE#################### B-tree####### ################################## ## ##

#####I/O#################/RAM#########/####################

####################################### ####################################### ## ## ### ## ############################### ##############################B################

################

B####################################### ##################################### LP_DEAD########################### ### ############################ ############################## ################ VACUUM########

######VACUUM#################################### ############################ ############################ ####REINDEX#################################### ####################

##

################

64.6.2.

############4############ ####################(###0)############################ ################################ ## ##

################################(relcache)########################### ###

##

##########################

###src/backend/access/hash/README########## ##

#65#

####PostgreSQL################################

65.1.

##################################

###
###############PGDATA######### ###PGDATA####/var/lib/pgsql/data### ######
###

65.1######PGDATA################################
#postgresql.conf#pg_hba.conf###pg_ident.conf####################
#PGDATA#######

#65.1 PGDATA###

##	##
PG_VERSION	PostgreSQL#################
base	##############################
current_logfiles	##############################
global	pg_database########################### ####
pg_commit_ts	################################
pg_dynshmem	###################################
pg_logical	###########################
pg_multixact	## #######
pg_notify	LISTEN/NOTIFY#################
pg_replslot	###########################
pg_serial	## ###
pg_snapshots	##############################
pg_stat	##############################
pg_stat_tmp	##############################
pg_subtrans	##############################
pg_tblspc	##############################
pg_twophase	##################################
pg_wal	WAL###########################
pg_xact	################################
postgresql.auto.conf	ALTER SYSTEM######################## ######
postmaster.opts	##################################
postmaster.pid	###postmaster####ID#PID############### ###postmaster#################Unix##### ################################IP##### ##*#TCP###########################ID## ###########################

##################PGDATA/base#########################pg_database#######
OID###### ##

###########heap###
PostgreSQL##################################

#####[97.522523]############################## ################################
################# #########pg_class.relfilenode######### ###########
#########t*BBB_FFF*######### ###*BBB*######################ID#*FFF*##########
##
####65.3######## ####################_fsm################## ##################
###_vm############### #######65.4#########
##_init##############65.5####

##
##################OID#################################### TRUNCATE#REINDEX#CLUSTER############ALTER TABLE############OID### ################### ##############OID################## ##pg_class##### ################pg_class.relfilenode######## ######################### ################pg_relation_filenode()##############

################################
##############.1##########.2######### ##
############################## #######PostgreSQL######--with-segsize#######
################) ##
########

##########################*TOAST*################## TOAST######################
################################ TOAST##########pg_class.reltoastrelid#####
###TOAST########### 65.2#########

##################65.6################

################### ###################PGDATA/pg_tblspc####################
################CREATE TABLESPACE################################## ########
###########OID###### ####################PG_9.0_201008051###PostgreSQL######
######################### ##########################CREATE TABLESPACE#####
####################################
############OID######################### ####################################
########## pg_default###### pg_tblspc################PGDATA/base###### #
###pg_global######pg_tblspc################PGDATA/global######

pg_relation_filepath()############(PGDATA######)######### ################
################## ## ##########
######################_fsm#_vm#_init#################

##################################PGDATA/base/pgsql_tmp######pg_default###
##############################pgsql_tmp################ ########
#pgsql_tmp*PPP.NNN*######## ####*PPP*###########PID####*NNN*####################
########

65.2. TOAST

####TOAST#########The Oversized-Attribute Storage Technique#############

PostgreSQL##############8############################## #######################
##########################
################### #####TOAST################################## [###TOAST###

##] TOAST##########################
#####

#########TOAST######## ## TOAST
####################*varlena*############### ############4#####################
############## TOAST#################### *TOAST*##########################
######################## ######TOAST#############C#########TOAST#########
################ #######*TOAST*###############4##############################
####################PG_DETOAST_DATUM################################ ###
#36.13.1##########

TOAST#varlena######2###(#######################################)######### ##
######TOAST###############1######2^{30} - 1########### #####0###############
TOAST## ################
################4##########1############ ##################################
######## #######127#################################1GB################### 1
##########################4################4############## ##################
############################## ########1################0####################
#####TOAST##############2##########
############## ################0###
#####4## ################varlena####
################################## #####TOAST####################

#########################CREATE TABLE###ALTER TABLE#COMPRESSION#########
################ ################################default_toast_compression#######
#######

########TOAST#################### ##########*TOAST*#########################
TOAST######TOAST#################################### #######################
##############################TOAST######access/common/toast_internals.c#
############## ######65.2.1######### ####TOAST################################
####### ## ####
##65.2.2##########

65.2.1. ########TOAST##

#######1###TOAST#####################TOAST#########OID#####
#pg_class.reltoastrelid########### ######TOAST##################TOAST###
########

####################TOAST_MAX_CHUNK_SIZE########### ###########4######1###
#######2000############### ##################TOAST################### ####
TOAST#####chunk_id#####TOAST#########OID##chunk_seq#########
###chunk_data################ chunk_id#chunk_seq#########################
##############TOAST#######################TOAST#####OID######
OID(chunk_id)############## ##
varlena###################TOAST######
#########################18########

TOAST####################TOAST_TUPLE_THRESHOLD######2####################
TOAST#########TOAST_TUPLE_TARGET#########2##############################
######################### ######################## #########################
TOAST############

TOAST###############TOAST###############4##############

- PLAIN############### ###TOAST#########################
- EXTENDED################ #######TOAST############### ###################
  ##############

- EXTERNAL############## EXTERNAL#######text#bytea###################### ################################### ######################
- MAIN#################### ########################### ######################### ################

TOAST################################### ###ALTER TABLE ... SET STORAGE# ###########################

TOAST_TUPLE_TARGET#ALTER TABLE ... SET (toast_tuple_target = N)#### ############

TOAST######################## ######################### ### ########### ##################################### ################HTML###### URL#########TOAST##10%## #URL######HTML############ ####HTML####7############TOAST##############

65.2.2. ########TOAST##

TOAST## ################### ######## ##############################2##########

##TOAST##############################varlena######### ########################## #################1GB################################ ######################### ################ ## ####################

##TOAST###################################### ###PostgreSQL######varlena####### ####NULL#######NULL############################### ############N############## ###################### ## ################################### TOAST######################varlena######## ########################TOAST############################ ################### ###########src/include/utils/expandeddatum.h#####API############## ##### ####c############################## ##################PG_DETOAST_DATUM######## ############varlena########## ##################1##1################

########TOAST##################################### ########################## ## ################## ## ##

##############TOAST#########TOAST## ########## ######TOAST##########################varlena####### ################ ##############TOAST#################

65.3.

###FSM### #### ##_fsm################### ###### ##################12345#####FSM######################12345_fsm######## ##########

########FSM################# #######FSM#################################### ###1############ ######################

#FSM############1######################## ########################FSM######## ## ############################ ###########################

FSM##################################src/backend/storage/freespace/README######### pg_freespacemap#####################################

65.4.

###VM##### ############################## ####################_vm######################### ######################### ##################12345####VM#################### #12345_vm############### #######VM################

###############2######### ## ######################## ## ###### 2## ######################### ##############

###
######### ############################ ##############################

pg_visibility##############################

65.5.

####################

65.6.

####PostgreSQL#################################### [1] TOAST#################### ##########

#######1####8################### ## ##### ######################################

#########################8## ######## ############## ################################## ############################ ################ ##

65.2##################### ######5#########

#65.2

####	##
PageHeaderData	###24##############################
ItemIdData	######################## ############### ##### 1#######4######
####	############## ######################## ###################
####	##############
#####	## ##################

###########24##########(PageHeaderData)########### ###### 65.3######## ######## ################WAL######### 2#########data checksums#########################

[1] ## heap######################## ######################## ##

##########2############## ####2###########3#####
#pd_lower#pd_upper#pd_special## ##
########################## ########2######pd_pagesize_version#############
############ PostgreSQL 8.3##########4#PostgreSQL 8.1#8.2########3#PostgreSQL 8.0
########2#PostgreSQL 7.3#7.4########1### ##################0## ################
##
################################## ################################## ##########
############XMAX#########

#65.3 PageHeaderData######

#####	#	##	##
pd_lsn	PageXLogRecPtr	8###	LSN: ############## ##WAL############ #####
pd_checksum	uint16	2###	########
pd_flags	uint16	2###	######
pd_lower	LocationIndex	2 ###	################
pd_upper	LocationIndex	2###	################
pd_special	LocationIndex	2###	##################
pd_pagesize_version	uint16	2###	################### ######
pd_prune_xid	TransactionId	4###	################### ##XMAX###########

########src/include/storage/bufpage.h#########

##################ItemIdData#### ######4########## ####################
##################################
############################pd_lower############
###
##PostgreSQL##############ItemPointer#CTID################################
#######

##########
##########HeapTupleHeaderData###########

####B-tree################
################################
#####pd_special##############

65.1#####################################

#65.1

65.6.1.

######################## ###################23#########################NULL##### #############ID#################### ######## 65.4####### ##################### ######MAXALIGN########t_hoff################## NULL#######*HEAP_HASNULL*## ##t_infomask################### ################################1############# #############t_infomask2################### ############1#####NULL##0#### NULL###### ########################NULL######## ######ID#*HEAP_HASOID_OLD*### #t_infomask################### ##########t_hoff########### t_hoff#MAXALIGN ###################NULL#############ID####### ###############ID############### ###

#65.4 HeapTupleHeaderData######

#####	#	##	##
t_xmin	TransactionId	4###	##XID####
t_xmax	TransactionId	4###	##XID####
t_cid	CommandId	4###	##############CID# ####t_xvac####
t_xvac	TransactionId	4###	########VACUUM# ##XID
t_ctid	ItemPointerData	6###	################# #TID
t_infomask2	uint16	2###	############
t_infomask	uint16	2###	########
t_hoff	uint8	1###	##############

########src/include/access/htup_details.h#########

##########################pg_attribute######################### ################# ######attlen###attalign### ###############NULL################################ ### ##########*heap_getattr#fastgetattr###heap_getsysattr*#############

################################### ###NULL################NULL########### ## NULL############ ######################### ################################### # ########attlen == -1############## ##################################struct varlena################## ##########################TOAST######################## ## #65.2###########

65.7. ########(HOT)

#################PostgreSQL###############(MVCC)############# ####MVCC######## ################ ################################### ########################### ##

##################PostgreSQL#########HOT############## ################

- ## ###PostgreSQL################## #BRIN###
- ##################################

##################2############

- ##

- ## ######################## ## ## ## ##)#

#####`fillfactor`########HOT################################ ##########HOT################ ## ##########pg_stat_all_tables##HOT####HOT#############

#66#

######PostgreSQL############################ ##################*xact*########

66.1.

#########BEGIN###START TRANSACTION############COMMIT###ROLLBACK#######
###############SQL#######################

#####[88.190476]##############VirtualTransactionId#####virtualXID###vxid##
############ #########ID####backendID###localXID#########################
##############ID4/12532#backendID#4##localXID#12532###
#########VirtualTransactionIdvirtualXID###vxid######)########## ######
#############procNumber###localXID################################### ###
##############ID4/12532#procNumber#4##localXID#12532###

###TransactionId####xid#####278394##PostgreSQL############################
###################### ### ########xid
#########xid###################### ##
###################### ##

#########ID#xid#32#####40################# 32################### #########
#######################64####xid8########xid####### # 9.82####xid8###### XID#
PostgreSQL#MVCC################################

#####xid###########################pg_xact#######################
track_commit_timestamp###########pg_commit_ts#############

#####[92.610837]#vxid#xid####################################GID##########
GID###200## GID##xid######
#pg_prepared_xacts#######

66.2.

######################ID##pg_locks#virtualxid#transactionid####### ######
#########virtualxid######transactionid#NULL### ########################
####

##########virtualxid################transactionid###### ################
#########pgrowlocks############# ######################ID#mxid#24.1.5.1######
##############

66.3.

###########################
################################### ## ###
#######*subxact*########

#####[91.845494]###########SAVEPOINT####################PL/pgSQL#EXCEPTION#
############### PL/Python#PL/Tcl######################### ####################
############## ##
###########

####################ID###ID####################ID##subxid####### ###########
#####subxid############################ ####################################
#############ID######### #xid####subxid###################

#subxid#####xid#pg_subtrans############# #######xid#################### ####################################

##xid################## ##

xid###pg_xact############## ############### ##

###64################################## ####pg_subtrans####xid############### ###I/O#################

66.4. 2##########

PostgreSQL#####################################2######2PC################# #######PREPARE TRANSACTION#COMMIT PREPARED#ROLLBACK PREPARED### 2########### ################################ PostgreSQL##X/Open XA########################### #######################

####PREPARE TRANSACTION################COMMIT PREPARED###ROLLBACK PREPARED########## ## ######################WAL########## ##################### #pg_twophase############## ################pg_prepared_xacts###########

#67#

PostgreSQL## ######
####################################C#######################################
########## ##C###############
############################## ##################################

###############src/include/catalog/##############C################ ######
################################pg_class####pg_class.h##################
OID##################

SQL####################################initdb##bootstrap######################
####### ######pg_class.h## #######
##src/include/catalog/## ###
##pg_proc.dat##pg_proc#######################

###BKI#Backend
Interface: ######################### ##########postgres.bki##genbki.pl###Perl
##########PostgreSQL## postgres.bki#
PostgreSQL##share##############
#####

genbki.pl####################pg_class####pg_class_d.h################ ##
################################enum##########################C#########

#####PostgreSQL######BKI###
############################## ######################### ##########BKI########
##########

67.1.

##################################C####### ###CATALOG##########C##########
#####typedef struct FormData_*catalogname*###### ####################
######genbki.h######BKI########################
##NULL######################## CATALOG############ genbki.h######BKI######
##

###
#################C#################### ###################NULL###########
#####pg_type.typrelid#NULL########
#######typetup->typrelid##########################typetup->typelem#####
##################typrelid########## ######################################
###

############################NULL##########C#################### ####ifdef
CATALOG_VARLEN … #endif###################CATALOG_VARLEN###############
#####C##
###################NULL###########pg_attribute################## #######
############NULL##############################NOT NULL######### #########
##BKI_FORCE_NOT_NULL#BKI_FORCE_NULL#####################

###############pg_xxx.h###########include########## ##################C#####
############## ######src/backend/catalog/########################## ###
################pg_xxx_d.h####include##### #######OID##define##############
########## ##ifdef
EXPOSE_TO_CLIENT_CODE … #endif################genbki.pl#######
#pg_xxx_d.h#################

##################################BKI create################################
###################### ##
pg_class#pg_type###
######## ################pg_attribute##########################genbki.pl##
########### #######################################

67.2.

##.dat########
##

67.2.1.

###.dat######Perl################ ################1#######1####################
############# pg_database.dat####################################

```
[

# A comment could appear here.
{ oid => '1', oid_symbol => 'Template1DbOid',
  descr => 'database\'s default template',
  datname => 'template1', encoding => 'ENCODING',
  datlocprovider => 'LOCALE_PROVIDER', datistemplate => 't',
  datallowconn => 't', dathasloginevt => 'f', datconnlimit => '-1',
 datfrozenxid => '0',
  datminmxid => '1', dattablespace => 'pg_default', datcollate =>
 'LC_COLLATE',
  datctype => 'LC_CTYPE', datlocale => 'DATLOCALE', datacl => '_null_' },

]
```

#########

- ######################## ################################ #####################
- ################*key=>value*######## #####*key*####################
  #oid#oid_symbol#array_type_oid#descr### #oid#oid_symbol#######67.2.2##
  #########array_type_oid#67.2.4######### descr################
  ##pg_description#pg_shdescription############# ######################
  #.h## ######
  #pg_database.h################datdba##############
- ######################## ########################## ##########################
  ######### ###Perl################################## ##########################
  #####4.1.2.2## ####\t############
  ##########################4########### Perl##########################\\######
- NULL##_null_###### ##############################
- ####################################
- ###########OID#################OID#################### ######
  #dattablespace############## ######67.2.3#######
- ## ##########################
  #reformat_dat_file.pl##################
  - ####################oid#oid_symbol#array_type_oid############
    ##descr########### ############################

- ##########80################################ ############################ ###
- #####.h################################reformat_dat_file.pl######### ############# ###############
- reformat_dat_file.pl################### ################reformat_dat_file.pl############## #########src/include/catalog/######make reformat-dat-files###########

- ############################reformat_dat_file.pl###### ############# #Catalog::ParseData()############

67.2.2. OID###

###############oid=> *nnnn*######################OID########### #######OID# ##########oid_symbol => *name*################OID##C#############

################OID##########################OID############ C######OID#### ##########OID###### ##############oid################ ################### OID####### ##OID############### ###################

C#####OID######################### pg_proc################################## ######## src/backend/utils/Gen_fmgrtab.pl####### ######################### ####pg_type OID################### #####oid_symbol######################### ################# ####pg_class########OID################### ########### #####oid_symbol#########################

################OID#######src/include/catalog/unused_oids########### ## ####OID########## ########45-900#OIDs 45##900################ ###OID 1–9999 ################## unused_oids################.dat###########OID### ####### #########duplicate_oids############ #genbki.pl################ OID################OID#######

#######################OID############8000—9999############################ OID############ ###################OID############ 8000—9999############## ########git################OID############################## ################ ##################################OID######### #####renumber_oids.pl###### ###### ##########################OID################################ #renumber_oids.pl############

##########OID################################OID########################### ######## ##OID#######

genbki.pl##########OID#############OID#############10000—11999######### ##################################OID############################OID##### 10000###########OID#######################

FirstUnpinnedObjectId (12000)###OID########################### #IsPinnedObject()################## initdb############################ ##OID#####FirstUnpinnedObjectId############ ### ##information_schema.sql#######################initdb####################### ######genbki.pl#######################

##################OID#16384######### ###genbki.pl#initdb############OID#### 10000—16383################ ############OID################################## ##################

67.2.3. OID####

###OID############## #
##############################OID##################################### ###
##genbki.pl########################## ##############

- BKI_LOOKUP(*lookuprule*)##################################### ##
  #*lookuprule*################pg_proc###
  BKI_LOOKUP##Oid#regproc#oidvector#Oid[]############# #################
  ################

- ###################BKI_LOOKUP(encoding)##########################OID####
  ##########genbki.pl######

- ## ###########BKI_LOOKUP####
  #BKI_LOOKUP_OPT######## ##########0########## ###regproc########
  ###0#####-########### ###########BKI_LOOKUP###########################
  genbki.pl##################

- ############################## #########pg_type#####typname############
  #############int4####integer###########

- ###pg_proc.dat###########proname####### #regproc############## ######
  ##regprocedure#####*proname(argtypename,argtypename,...)*######## ######
  #pg_proc.dat#####proargtypes############# #############

- ####*oprname(lefttype,righttype)*###### ######pg_operator.dat####
  #oprleft#####oprright########## #################0#######

- opclasses#opfamilies######################*access_method_name/object_name*#
  #####

- ########################## ##########################pg_catalog###########
  ###

genbki.pl#######################BKI############OID###### #################
####################

#########################OID####BKI_LOOKUP#BKI_LOOKUP_OPT###############
######genbki.pl##################################
################## ##
#DECLARE_FOREIGN_KEY#DECLARE_FOREIGN_KEY_OPT#DECLARE_ARRAY_FOREIGN_KEY#DECLARE_A
#########BKI_LOOKUP############(##############)###########

67.2.4.

##varlena#########pg_type####
#typarray################ genbki.pl###############pg_type###############

##############pg_type#####array_type_oid => *nnnn*#########################
#OID###### ######OID###########typarray##############

####################################### ##############
##pg_type.h#BKI_ARRAY_DEFAULT(*value*)##########################
#typalign############### ########typelem###typarray####################

67.2.5.

##

#####################. BKI_DEFAULT(*value*)################### ###########
##################################

##########################. BKI_DEFAULT##################################
#make reformat-dat-files######

#####################. ##########make reformat-dat-files###############
##########

###################. ## ##make
expand-dat-files### ##BKI_DEFAULT##
##########make reformat-dat-files####################

#############: reformat_dat_file.pl################## #################
############### #####pg_proc##############################

1. ############pg_proc.h#######

```
+    /* see PROKIND_ categories below */
+    char        prokind BKI_DEFAULT(f);
```

2. #################reformat_dat_file.pl################

```
-           # At this point we have the full row in memory as a hash
-           # and can do any operations we want. As written, it only
-           # removes default values, but this script can be adapted to
-           # do one-off bulk-editing.
+           # One-off change to migrate to prokind
+           # Default has already been filled in by now, so change to
 other
+           # values as appropriate
+           if ($values{proisagg} eq 't')
+           {
+               $values{prokind} = 'a';
+           }
+           elsif ($values{proiswindow} eq 't')
+           {
+               $values{prokind} = 'w';
+           }
```

3. ###########

```
$ cd src/include/catalog
$ perl  rewrite_dat_with_prokind.pl  pg_proc.dat
```

   #####pg_proc.dat##prokind#proisagg#proiswindow#####3################
   ##########

4. pg_proc.h###########

```
-    /* is it an aggregate? */
-    bool        proisagg BKI_DEFAULT(f);
-
-    /* is it a window function? */
-    bool        proiswindow BKI_DEFAULT(f);
```

5. ####make reformat-dat-files#####pg_proc.dat###################

####################https://www.postgresql.org/message-id/CAJVSVGVX8gXnPm
+Xa=DxR7kFYprcQ1tNcCT5D0O3ShfnM6jehA@mail.gmail.com####
#convert_oid2name.pl#remove_pg_type_oid_symbols.pl#######

67.3. BKI######

####PostgreSQL##############BKI################# ####postgres.bki######### ##################

BKI################## ##################################### ################ ################################ ################## ########################## ################## ################################## ###################################### ###/#############

#############

67.4. BKI####

create *tablename* *tableoid* [bootstrap] [shared_relation] [rowtype_oid *oid*] (*name1* = *type1* [FORCE NOT NULL | FORCE NULL] [, *name2* = *type2* [FORCE NOT NULL | FORCE NULL], ...])

##########OID *tableoid*###*tablename*############

#####bootstrap.c########### bool#bytea#char#1## ###name#int2#int4#regproc#regclass#regtype#text#oid#tid#xid#cid#int2vector#oidv ###_text#####_oid#####_char#####_aclitem##### ######################## ##pg_type############################## ################################## ##

bootstrap###########################pg_class#pg_attribute########### #### #############insert################SQL################## ####### #pg_class##################

shared_relation######################## #######OID#pg_type OID# #rowtype_oid######## ########OID########### #bootstrap ####### ##rowtype_oid################################

open *tablename*

##########*tablename*################## ##################

close *tablename*

############## ######################

insert ([*oid_value*] *value1* *value2* ...)

value1#*value2*##########################

NULL##########_null_########## ##################################### (### ##################### ######################################)

declare [unique] index *indexname* *indexoid* on *tablename* using *amname* (*opclass1* *name1* [, ...])

amname#############*tablename*#############OID *indexoid*#### ##*indexname*################## ##################*name1*#*name2*########### ######*opclass1*#*opclass2*############ ################################## ##########################

declare toast *toasttableoid* *toastindexoid* on *tablename*

tablename###########TOAST########## ##TOAST#####OID## #*toasttableoid*################OID###*toastindexoid*######### declare index####################

build indices

###################

67.5. BKI#################

open###
##pg_class#pg_attribute#pg_proc####pg_type#######################
##bootstrap########create###########################

##############################declare index###declare toast############

###postgres.bki################

1. 1#########create bootstrap
2. ####################insert
3. close
4. ################
5. ##########bootstrap####create
6. open
7. ########insert
8. close
9. ###################
10. ########TOAST#######
11. build indices

##

67.6. BKI##

##############oid##int4##text##3####oid#cola#colb####OID 420 ### #test_table###########2##############

```
create test_table 420 (oid = oid, cola = int4, colb = text)
open test_table
insert ( 421 1 'value 1' )
insert ( 422 2 _null_ )
close test_table
```

#68#

####14.1#14.2## ##################### ######################################

########################### ########################### ######################## ##############

68.1.

#####[55.755396]######PostgreSQL############################ #############8.3##### ### ################################## ###ANALYZE############################ #####ANALYZE######################### ##########PostgreSQL################### ########## ###ANALYZE##############################ANALYZE################### ###########

###################

```
EXPLAIN SELECT * FROM tenk1;

                        QUERY PLAN
-------------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..458.00 rows=10000 width=244)
```

##########tenk1##############14.2####### ########################## ######### #pg_class#########

```
SELECT relpages, reltuples FROM pg_class WHERE relname = 'tenk1';

 relpages | reltuples
----------+-----------
      358 |     10000
```

################VACUUM###ANALYZE############ ################################ ##################### ###relpages######reltuples########################## ###### ######relpages################reltuples######

##WHERE#################

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 1000;

                                    QUERY PLAN
-------------------------------------------------------------------
-------------
 Bitmap Heap Scan on tenk1  (cost=24.06..394.64 rows=1007 width=244)
   Recheck Cond: (unique1 < 1000)
   ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..23.80 rows=1007
 width=0)
         Index Cond: (unique1 < 1000)
```

#####WHERE#########pg_operator##<################ ###oprrest######### ###### ####scalarltsel### scalarltsel####pg_statistic##unique1############## ## ################pg_stats##############

```
SELECT histogram_bounds FROM pg_stats
WHERE tablename='tenk1' AND attname='unique1';

                    histogram_bounds
```

```
-----------------------------------------------------
 {0,993,1997,3050,4040,5036,5957,7057,8029,9016,9995}
```

####< 1000######################### ######(selectivity)### ####################(bucket) ####### ## 1000####### ##2#######(993–1997)###### ##

```
selectivity = (1 + (1000 - bucket[2].min)/(bucket[2].max -
 bucket[2].min))/num_buckets
            = (1 + (1000 - 993)/(1997 - 993))/10
            = 0.100697
```

####1#########2#################################### ###############tenk1##### ################

```
rows = rel_cardinality * selectivity
     = 10000 * 0.100697

     = 1007  (####)
```

###WHERE####################

```
EXPLAIN SELECT * FROM tenk1 WHERE stringu1 = 'CRAAAA';

                        QUERY PLAN
-------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..483.00 rows=30 width=244)
   Filter: (stringu1 = 'CRAAAA'::name)
```

##########WHERE########=############eqsel####### ######################### ### #############MCV############ MCV######## #################

```
SELECT null_frac, n_distinct, most_common_vals, most_common_freqs FROM
 pg_stats
WHERE tablename='tenk1' AND attname='stringu1';

null_frac         | 0
n_distinct        | 676
most_common_vals  | {EJAAAA,BBAAAA,CRAAAA,FCAAAA,FEAAAA,GSAAAA,
JOAAAA,MCAAAA,NAAAAA,WGAAAA}
most_common_freqs | {0.00333333,0.003,0.003,0.003,0.003,0.003,
0.003,0.003,0.003,0.003}
```

CRAAAA#MCV#########################MCF##################

```
selectivity = mcf[3]
            = 0.003
```

#######################tenk1########

```
rows = 10000 * 0.003
     = 30
```

################# ##########MCV#######

```
EXPLAIN SELECT * FROM tenk1 WHERE stringu1 = 'xxx';

                        QUERY PLAN
```

```
-------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..483.00 rows=15 width=244)
   Filter: (stringu1 = 'xxx'::name)
```

##MCV################################ ###################MCV################
#######

```
selectivity = (1 - sum(mcv_freqs))/(num_distinct - num_mcv)
            = (1 - (0.00333333 + 0.003 + 0.003 + 0.003 + 0.003 + 0.003 +
                    0.003 + 0.003 + 0.003 + 0.003))/(676 - 10)
            = 0.0014559
```

####MCV###############1########################### ####MCV##################
############################## NULL################################ ########
#####NULL################ ####################

```
rows = 10000 * 0.0014559

     = 15  (####)
```

###unique1 < 1000######scalarltsel######################### ######MCV
################################## unique1##########MCV#####################
############################## ###################MCV############ #########
##MCV################## ########################### #######scalarltsel#####
#####< 1000###MCV################MCV########### ###MCV####################
###################MCV###################### #####2#######################
############

```
EXPLAIN SELECT * FROM tenk1 WHERE stringu1 < 'IAAAAA';

                        QUERY PLAN
-------------------------------------------------------
 Seq Scan on tenk1  (cost=0.00..483.00 rows=3077 width=244)
   Filter: (stringu1 < 'IAAAAA'::name)
```

###stringu1#MCV############################

```
SELECT histogram_bounds FROM pg_stats
WHERE tablename='tenk1' AND attname='stringu1';

                               histogram_bounds
-----------------------------------------------------------------
-------------
 {AAAAAA,CQAAAA,FRAAAA,IBAAAA,KRAAAA,NFAAAA,PSAAAA,SGAAAA,VAAAAA,
XLAAAA,ZZAAAA}
```

MCV#########stringu1 < 'IAAAAA'######6##########4################## ######
###MCV####################

```
selectivity = sum(relevant mvfs)
            = 0.00333333 + 0.003 + 0.003 + 0.003 + 0.003 + 0.003
            = 0.01833333
```

MCF######MCV##############0.03033333######################0.96966667########
######NULL################################ IAAAAA#####3##############
############### ##############################IAAAAA######################
0.298387###### ###MCV##MCV##################

```
selectivity = mcv_selectivity + histogram_selectivity * histogram_fraction
```

```
            = 0.01833333 + 0.298387 * 0.96966667
            = 0.307669

rows        = 10000 * 0.307669

            = 3077  (####)
```

#########################MCV#################### ############################## ################ ## #########

##WHERE####################

```
EXPLAIN SELECT * FROM tenk1 WHERE unique1 < 1000 AND stringu1 = 'xxx';

                                    QUERY PLAN
-------------------------------------------------------------------
-------------
 Bitmap Heap Scan on tenk1  (cost=23.80..396.91 rows=1 width=244)
   Recheck Cond: (unique1 < 1000)
   Filter: (stringu1 = 'xxx'::name)
   ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..23.80 rows=1007
 width=0)
         Index Cond: (unique1 < 1000)
```

#####2################# #######################

```
selectivity = selectivity(unique1 < 1000) * selectivity(stringu1 = 'xxx')
            = 0.100697 * 0.0014559
            = 0.0001466

rows        = 10000 * 0.0001466

            = 1  (####)
```

####################

#######################

```
EXPLAIN SELECT * FROM tenk1 t1, tenk2 t2
WHERE t1.unique1 < 50 AND t1.unique2 = t2.unique2;

                                      QUERY PLAN
-------------------------------------------------------------------
-------------------
 Nested Loop  (cost=4.64..456.23 rows=50 width=488)
   ->  Bitmap Heap Scan on tenk1 t1  (cost=4.64..142.17 rows=50 width=244)
         Recheck Cond: (unique1 < 50)
         ->  Bitmap Index Scan on tenk1_unique1  (cost=0.00..4.63 rows=50
 width=0)
               Index Cond: (unique1 < 50)
   ->  Index Scan using tenk2_unique2 on tenk2 t2  (cost=0.00..6.27 rows=1
 width=244)
         Index Cond: (unique2 = t1.unique2)
```

tenk1 unique1 < 50########################## ####################### ######## 50#unique1################################

```
selectivity = (0 + (50 - bucket[1].min)/(bucket[1].max - bucket[1].min))/
num_buckets
```

```
            = (0 + (50 - 0)/(993 - 0))/10
            = 0.005035

rows        = 10000 * 0.005035

            = 50  (####)
```

#####t2.unique2 = t1.unique2### ############=######### #pg_operator#oprjoin#######eqjoinsel##### eqjoinsel#tenk2###tenk1##### #########

```
SELECT tablename, null_frac,n_distinct, most_common_vals FROM pg_stats
WHERE tablename IN ('tenk1', 'tenk2') AND attname='unique2';

tablename  | null_frac | n_distinct | most_common_vals
-----------+-----------+------------+------------------
 tenk1     |         0 |         -1 |
 tenk2     |         0 |         -1 |
```

#####[62.269939]####################unique2####MCV######## ################# NULL############################ ###########unique2####MCV################ #########(n_distinct = -1)# #########num_rows########"tenk"##############NULL##### #######################

```
selectivity = (1 - null_frac1) * (1 - null_frac2) / max(num_rows1,
 num_rows2)
            = (1 - 0) * (1 - 0) / max(10000, 10000)
            = 0.0001
```

#####[76.946108]################1##NULL####################### ############## ##2################################## ####################NULL###1######### ## ################2################ ###################

```
rows = (outer_cardinality * inner_cardinality) * selectivity
     = (50 * 10000) * 0.0001
     = 50
```

2#######MCV##########eqjoinsel#MCV#######################MCV######### ## ############################

inner_cardinality#10000####tenk2######################### EXPLAIN######## #########50 * 1##########tenk2## ## ############## ################################## #####################2#### ########## ##############################

#######################WHERE###############src/backend/optimizer/util/ plancat.c###### ####################src/backend/optimizer/path/ clausesel.c###### ###################src/backend/utils/adt/selfuncs.c## #####

68.2.

68.2.1.

################################ 2#######################

```
CREATE TABLE t (a INT, b INT);
```

```
INSERT INTO t SELECT i % 100, i % 100 FROM generate_series(1, 10000) s(i);
ANALYZE t;
```

14.2###########pg_class################t###########

```
SELECT relpages, reltuples FROM pg_class WHERE relname = 't';

 relpages | reltuples
----------+-----------
       45 |     10000
```

#######################100#########################

######a#####WHERE################

```
EXPLAIN (ANALYZE, TIMING OFF) SELECT * FROM t WHERE a = 1;
                                 QUERY PLAN
-------------------------------------------------------------------
------------
 Seq Scan on t  (cost=0.00..170.00 rows=100 width=8) (actual rows=100
 loops=1)
   Filter: (a = 1)
   Rows Removed by Filter: 9900
```

#####################1%####### ### ##### ######################### WHERE#######b###################### ###AND########### ##########################

```
EXPLAIN (ANALYZE, TIMING OFF) SELECT * FROM t WHERE a = 1 AND b = 1;
                                 QUERY PLAN
-------------------------------------------------------------------
----------
 Seq Scan on t  (cost=0.00..195.00 rows=1 width=8) (actual rows=100
 loops=1)
   Filter: ((a = 1) AND (b = 1))
   Rows Removed by Filter: 9900
```

###################1%######### ## #0.01%####### ##############2#####(100)################

######ANALYZE###

```
CREATE STATISTICS stts (dependencies) ON a, b FROM t;
ANALYZE t;
EXPLAIN (ANALYZE, TIMING OFF) SELECT * FROM t WHERE a = 1 AND b = 1;
                                  QUERY PLAN
-------------------------------------------------------------------
------------
 Seq Scan on t  (cost=0.00..195.00 rows=100 width=8) (actual rows=100
 loops=1)
   Filter: ((a = 1) AND (b = 1))
   Rows Removed by Filter: 9900
```

68.2.2. ###N#####

GROUP BY## GROUP BY###########N# ######HashAggregate#########################

```
EXPLAIN (ANALYZE, TIMING OFF) SELECT COUNT(*) FROM t GROUP BY a;
```

```
                                        QUERY PLAN
-----------------------------------------------------------------
----------------------
 HashAggregate  (cost=195.00..196.00 rows=100 width=12) (actual rows=100
 loops=1)
   Group Key: a
   ->  Seq Scan on t  (cost=0.00..145.00 rows=10000 width=4) (actual
 rows=10000 loops=1)
```

#######################GROUP BY#####################################

```
EXPLAIN (ANALYZE, TIMING OFF) SELECT COUNT(*) FROM t GROUP BY a, b;
                                        QUERY PLAN
-----------------------------------------------------------------
-------------------------
 HashAggregate  (cost=220.00..230.00 rows=1000 width=16) (actual rows=100
 loops=1)
   Group Key: a, b
   ->  Seq Scan on t  (cost=0.00..145.00 rows=10000 width=8) (actual
 rows=10000 loops=1)
```

#########N##

```
DROP STATISTICS stts;
CREATE STATISTICS stts (dependencies, ndistinct) ON a, b FROM t;
ANALYZE t;
EXPLAIN (ANALYZE, TIMING OFF) SELECT COUNT(*) FROM t GROUP BY a, b;
                                        QUERY PLAN
-----------------------------------------------------------------
-------------------------
 HashAggregate  (cost=220.00..221.00 rows=100 width=16) (actual rows=100
 loops=1)
   Group Key: a, b
   ->  Seq Scan on t  (cost=0.00..145.00 rows=10000 width=8) (actual
 rows=10000 loops=1)
```

68.2.3. MCV###

68.2.1##(#########################
###########)#

#####MCV(###)###########68.1####################### ###########################
##ANALYZE##############################

##68.2.1#################################MCV########(###########################
################)#

```
DROP STATISTICS stts;
CREATE STATISTICS stts2 (mcv) ON a, b FROM t;
ANALYZE t;
EXPLAIN (ANALYZE, TIMING OFF) SELECT * FROM t WHERE a = 1 AND b = 1;
                                     QUERY PLAN
-----------------------------------------------------------------
------------
 Seq Scan on t  (cost=0.00..195.00 rows=100 width=8) (actual rows=100
 loops=1)
   Filter: ((a = 1) AND (b = 1))
   Rows Removed by Filter: 9900
```

################2##########
####MCV##############

MCV###########pg_mcv_list_items#######

```
SELECT m.* FROM pg_statistic_ext join pg_statistic_ext_data on (oid =
 stxoid),
                pg_mcv_list_items(stxdmcv) m WHERE stxname = 'stts2';
 index |  values  | nulls | frequency | base_frequency
-------+----------+-------+-----------+----------------
     0 | {0, 0}   | {f,f} |      0.01 |         0.0001
     1 | {1, 1}   | {f,f} |      0.01 |         0.0001
   ...
    49 | {49, 49} | {f,f} |      0.01 |         0.0001
    50 | {50, 50} | {f,f} |      0.01 |         0.0001
   ...
    97 | {97, 97} | {f,f} |      0.01 |         0.0001
    98 | {98, 98} | {f,f} |      0.01 |         0.0001
    99 | {99, 99} | {f,f} |      0.01 |         0.0001
(100 rows)
```

####2####100##########################(####1%###)########### #######(base frequency)################################## ###########NULL######nulls#######
###

###############MCV################################## ###src/backend/statistics/mcv.c#mcv_clauselist_selectivity#########

##########MCV#######2######1##

```
EXPLAIN (ANALYZE, TIMING OFF) SELECT * FROM t WHERE a = 1 AND b = 10;
                                 QUERY PLAN
-----------------------------------------------------------------
--------
 Seq Scan on t  (cost=0.00..195.00 rows=1 width=8) (actual rows=0 loops=1)
   Filter: ((a = 1) AND (b = 10))
   Rows Removed by Filter: 10000
```

2####MCV## ############################
#########

```
EXPLAIN (ANALYZE, TIMING OFF) SELECT * FROM t WHERE a <= 49 AND b > 49;
                                QUERY PLAN
-----------------------------------------------------------------
--------
 Seq Scan on t  (cost=0.00..195.00 rows=1 width=8) (actual rows=0 loops=1)
   Filter: ((a <= 49) AND (b > 49))
   Rows Removed by Filter: 10000
```

68.3.

####pg_statistic###
######### ##
##=################################
######## ####pg_statistic################################## ##################
##
####pg_statistic##

####################################
##################SELECT##
#LEAKPROOF########### ##
##############

######################
##
################################ #################(leak-proof)################### #
############################
######################################

###########pg_statistic#1#################################### ########NULL
##

#########
####PostgreSQL################

#69#

pg_basebackup##################pg_verifybackup################################ ### ################### ##############

############UTF-8########JSON######## ####JSON#######################
##PostgreSQL#`json`###`jsonb`##
########### ##JSON###

69.1.

############JSON##################

`PostgreSQL-Backup-Manifest-Version`

> ################### PostgreSQL17###2##########1###

`System-Identifier`

> #################PostgreSQL#################### ########`PostgreSQL-`### ###-######-######2###########

`Files`

> ## ####################
> WAL################################ ################69.2#######

`WAL-Ranges`

> ##WAL#####
> ########### ################69.3#######

`Manifest-Checksum`

> ################################## #################SHA256######### ######### ## SHA256#######CRC32C############## ##

69.2.

`##################Path###Encoded-Path#######  ###Path#######  ########` ############################## ######################2#####`pg_tblspc`#######
OID########### ###UTF-8###
`#Encoded-Path`######### ################16################## #######16####2### 1###########

###2#############

`Size`

> ##########################

`Last-Modified`

> ################################## ############################
> #pg_verifybackup######## ###################

##

`Checksum-Algorithm`

: ################################## ####################################
  #################### ################CRC32C#SHA224#SHA256#SHA384###
  ##SHA512###

`Checksum`

: ##################################2####16################

69.3. ##############WAL########

WAL####################3##########

`Timeline`

: WAL################################

`Start-LSN`

: ##LSN### ##LSN#PostgreSQL##########
  ##############2##16##################1##8###################

`End-LSN`

: ##LSN### ###`Start-LSN`###########
  ##

######WAL########### ##
################################## ###############WAL####################

VIII.

##

A. PostgreSQL###### 2286
B. ##/####### 2294
B.1. ##/####### 2294
B.2. ################### 2295
B.3. ##/####### 2295
B.4. ##/######## 2296
B.5. POSIX###### 2297
B.6. ##### 2299
B.7. #####(Julian Date) 2299
C. SQL##### 2301
D. SQL#### 2324
D.1. ########### 2325
D.2. ############ 2336
D.3. XML####SQL/XML#### 2344
D.3.1. ######XPath 1.0###### 2344
D.3.2. ######### 2345
E. ####### 2347
E.1. ####17 2347
E.1.1. ## 2347
E.1.2. #####17#### 2347
E.1.3. ### 2348
E.1.4. ## 2358
E.2. ####### 2358
F. ################ 2359
F.1. amcheck — ######################### 2360
F.1.1. ## 2360
F.1.2. #####`heapallindexed`## 2363
F.1.3. `amcheck`####### 2363
F.1.4. ##### 2364
F.2. auth_delay — ########### 2365
F.2.1. ####### 2365
F.2.2. ## 2365
F.3. auto_explain — ############### 2366
F.3.1. ####### 2366
F.3.2. # 2367
F.3.3. ## 2368
F.4. basebackup_to_shell — pg_basebackup#####"shell"## 2369
F.4.1. ####### 2369
F.4.2. ## 2369
F.5. basic_archive — WAL############ 2370
F.5.1. ####### 2370
F.5.2. ## 2370
F.5.3. ## 2370
F.6. bloom — ######################### 2371
F.6.1. ##### 2371
F.6.2. # 2371
F.6.3. ############# 2373
F.6.4. #### 2373
F.6.5. ## 2373
F.7. btree_gin — GIN#######B-tree## 2375
F.7.1. ### 2375
F.7.2. ## 2375
F.8. btree_gist — GiST#######B-tree## 2376
F.8.1. ### 2376
F.8.2. ## 2377

F.9. citext — ################ 2378
F.9.1. ## 2378
F.9.2. #### 2378
F.9.3. ######## 2378
F.9.4. #### 2379
F.9.5. ## 2379
F.10. cube — ########## 2381
F.10.1. ## 2381
F.10.2. ## 2381
F.10.3. #### 2381
F.10.4. ##### 2383
F.10.5. ## 2384
F.10.6. ##### 2384
F.11. dblink — ##PostgreSQL########### 2385
F.12. dict_int — ################ 2415
F.12.1. ## 2415
F.12.2. #### 2415
F.13. dict_xsyn — ############## 2416
F.13.1. ## 2416
F.13.2. #### 2416
F.14. earthdistance — ######### 2418
F.14.1. cube######### 2418
F.14.2. point######### 2419
F.15. file_fdw — ############################## 2420
F.16. fuzzystrmatch — ############### 2423
F.16.1. Soundex 2423
F.16.2. Daitch-Mokotoff Soundex 2423
F.16.3. #########(Levenshtein) 2425
F.16.4. Metaphone 2426
F.16.5. Double Metaphone 2426
F.17. hstore — hstore##/##### 2427
F.17.1. `hstore`##### 2427
F.17.2. `hstore`####### 2427
F.17.3. ###### 2431
F.17.4. # 2432
F.17.5. #### 2432
F.17.6. ### 2433
F.17.7. ## 2433
F.17.8. ## 2434
F.18. intagg — ########### 2435
F.18.1. ## 2435
F.18.2. ### 2435
F.19. intarray — ########## 2437
F.19.1. `intarray`######## 2437
F.19.2. ########## 2438
F.19.3. # 2439
F.19.4. ###### 2439
F.19.5. ## 2439
F.20. isn — #######ISBN#EAN#UPC########## 2440
F.20.1. #### 2440
F.20.2. #### 2440
F.20.3. ###### 2441
F.20.4. # 2441
F.20.5. #### 2442
F.20.6. ## 2443
F.21. lo — ############## 2444
F.21.1. ## 2444
F.21.2. #### 2444

F.21.3. #### 2444
F.21.4. ## 2444
F.22. ltree — ############# 2446
F.22.1. ## 2446
F.22.2. ###### 2447
F.22.3. ###### 2449
F.22.4. # 2450
F.22.5. ## 2452
F.22.6. ## 2452
F.23. pageinspect — ################ 2453
F.23.1. ###### 2453
F.23.2. ##### 2454
F.23.3. B-tree## 2455
F.23.4. BRIN## 2457
F.23.5. GIN## 2458
F.23.6. GiST## 2459
F.23.7. Hash## 2460
F.24. passwordcheck — ############ 2463
F.25. pg_buffercache — PostgreSQL################## 2464
F.25.1. `pg_buffercache`### 2464
F.25.2. `pg_buffercache_summary()`## 2465
F.25.3. The `pg_buffercache_usage_counts()` Function 2465
F.25.4. The `pg_buffercache_evict` Function 2466
F.25.5. ###### 2466
F.25.6. ## 2467
F.26. pgcrypto — #### 2468
F.26.1. ######## 2468
F.26.2. ########### 2468
F.26.3. PGP##### 2470
F.26.4. ########### 2475
F.26.5. ######## 2476
F.26.6. ## 2476
F.26.7. ## 2477
F.27. pg_freespacemap — ########### 2478
F.27.1. ## 2478
F.27.2. ###### 2478
F.27.3. ## 2479
F.28. pg_prewarm — ########################## 2480
F.28.1. ## 2480
F.28.2. ####### 2480
F.28.3. ## 2481
F.29. pgrowlocks — ############## 2482
F.29.1. ## 2482
F.29.2. ###### 2482
F.29.3. ## 2483
F.30. pg_stat_statements — SQL###################### 2484
F.30.1. `pg_stat_statements` ### 2484
F.30.2. `pg_stat_statements_info`### 2487
F.30.3. ## 2488
F.30.4. ####### 2488
F.30.5. ###### 2489
F.30.6. ## 2491
F.31. pgstattuple — ############### 2492
F.31.1. ## 2492
F.31.2. ## 2495
F.32. pg_surgery — ###################### 2496
F.32.1. ## 2496
F.32.2. ## 2497

F.33. pg_trgm — ########################## 2498
F.33.1. ################## 2498
F.33.2. ###### 2498
F.33.3. GUC##### 2500
F.33.4. ######### 2500
F.33.5. ######## 2501
F.33.6. ## 2502
F.33.7. ## 2502
F.34. pg_visibility — ############### 2503
F.34.1. ## 2503
F.34.2. ## 2504
F.35. pg_walinspect — #####WAL## 2505
F.35.1. ###### 2505
F.35.2. ## 2507
F.36. postgres_fdw — ###PostgreSQL################## 2508
F.36.1. postgres_fdw############## 2508
F.36.2. ## 2513
F.36.3. #### 2513
F.36.4. ######### 2513
F.36.5. ############ 2514
F.36.6. ############ 2514
F.36.7. ######## 2514
F.36.8. Wait Events 2514
F.36.9. ####### 2515
F.36.10. # 2515
F.36.11. ## 2516
F.37. seg — ################### 2517
F.37.1. ## 2517
F.37.2. ## 2517
F.37.3. ## 2518
F.37.4. #### 2518
F.37.5. ## 2519
F.37.6. ##### 2519
F.38. sepgsql — SELinux###################MAC########### 2520
F.38.1. ## 2520
F.38.2. ###### 2520
F.38.3. ######### 2521
F.38.4. GUC##### 2522
F.38.5. ## 2522
F.38.6. sepgsql## 2525
F.38.7. #### 2526
F.38.8. ###### 2526
F.38.9. ## 2526
F.39. spi — ###################/# 2527
F.39.1. refint — ########### 2527
F.39.2. autoinc — ############ 2527
F.39.3. insert_username — ################## 2527
F.39.4. moddatetime — ############ 2527
F.40. sslinfo — #######SSL####### 2528
F.40.1. ####### 2528
F.40.2. ## 2529
F.41. tablefunc — ########(`crosstab`#) 2530
F.41.1. ####### 2530
F.41.2. ## 2538
F.42. tcn — ######################## 2539
F.43. test_decoding — SQL####WAL#################/###### 2540
F.44. tsm_system_rows — `TABLESAMPLE`####`SYSTEM_ROWS`######### 2541
F.44.1. # 2541

F.45. tsm_system_time — TABLESAMPLE####SYSTEM_TIME######### 2542
F.45.1. # 2542
F.46. unaccent — ################# 2543
F.46.1. ## 2543
F.46.2. #### 2543
F.46.3. ## 2544
F.47. uuid-ossp — UUID### 2545
F.47.1. uuid-ossp## 2545
F.47.2. uuid-ossp### 2546
F.47.3. ## 2546
F.48. xml2 — XPath#####XSLT## 2547
F.48.1. ################ 2547
F.48.2. ##### 2547
F.48.3. xpath_table 2548
F.48.4. XSLT## 2549
F.48.5. ## 2550
G. ############# 2551
G.1. ############## 2551
G.2. ########### 2558
H. ######## 2559
H.1. ############# 2559
H.2. ##### 2559
H.3. ##### 2559
H.4. ## 2559
I. ########### 2560
I.1. Git############ 2560
J. ########## 2561
J.1. DocBook 2561
J.2. ###### 2561
J.2.1. Fedora#RHEL################# 2562
J.2.2. FreeBSD######## 2562
J.2.3. Debian##### 2562
J.2.4. macOS 2562
J.2.5. configure##### 2563
J.3. Make########### 2563
J.3.1. HTML 2563
J.3.2. ######## 2563
J.3.3. PDF 2563
J.3.4. #### 2564
J.4. Meson########### 2564
J.5. ##### 2564
J.5.1. Emacs 2565
J.6. ####### 2565
J.6.1. ######### 2565
K. PostgreSQL### 2567
L. ### 2568
M. ### 2575
N. ### 2587
N.1. ######### 2587
N.2. ###### 2587
O. ############### 2588
O.1. recovery.conf#####postgresql.conf### 2588
O.2. ######################## 2588
O.3. pg_xlogdump####pg_waldump### 2588
O.4. pg_resetxlog####pg_resetwal### 2588
O.5. pg_receivexlog####pg_receivewal### 2588
P. ### 2589

##A PostgreSQL######

PostgreSQL#######################SQL#####SQLSTATE##############5##############
####### ##
########## ##########PostgreSQL###
############## PostgreSQL#################################SQL####################
##SQL##

################2################3#################### ##########################
#####################################

A.1##PostgreSQL 17beta3######################### ###SQL##########################
######## ############## #############3###000#################### ################
##

##################PL/pgSQL############# ########################PL/pgSQL#######
##################################00#01#02####

###
##unique_violation#################### ####################################
PostgreSQL 9.3###
#################SQLSTATE###23################################

#A.1 PostgreSQL######

######	###
### 00 — ####	
00000	successful_completion
### 01 — ##	
01000	warning
0100C	dynamic_result_sets_returned
01008	implicit_zero_bit_padding
01003	null_value_eliminated_in_set_function
01007	privilege_not_granted
01006	privilege_not_revoked
01004	string_data_right_truncation
01P01	deprecated_feature
### 02 — ######(##SQL##############)	
02000	no_data
02001	no_additional_dynamic_result_sets_returned
### 03 — SQL#####	
03000	sql_statement_not_yet_complete
### 08 — #####	
08000	connection_exception
08003	connection_does_not_exist
08006	connection_failure
08001	sqlclient_unable_to_establish_sqlconnection
08004	sqlserver_rejected_establishment_of_sqlconnection
08007	transaction_resolution_unknown

######	###
08P01	protocol_violation
### 09 — #############	
09000	triggered_action_exception
### 0A — #########	
0A000	feature_not_supported
### 0B — #############	
0B000	invalid_transaction_initiation
### 0F — #######	
0F000	locator_exception
0F001	invalid_locator_specification
### 0L — ######	
0L000	invalid_grantor
0LP01	invalid_grant_operation
### 0P — ########	
0P000	invalid_role_specification
### 0Z — #####	
0Z000	diagnostics_exception
0Z002	stacked_diagnostics_accessed_without_active_handler
### 20 — case######	
20000	case_not_found
### 21 — ####	
21000	cardinality_violation
### 22 — #####	
22000	data_exception
2202E	array_subscript_error
22021	character_not_in_repertoire
22008	datetime_field_overflow
22012	division_by_zero
22005	error_in_assignment
2200B	escape_character_conflict
22022	indicator_overflow
22015	interval_field_overflow
2201E	invalid_argument_for_logarithm
22014	invalid_argument_for_ntile_function
22016	invalid_argument_for_nth_value_function
2201F	invalid_argument_for_power_function
2201G	invalid_argument_for_width_bucket_function
22018	invalid_character_value_for_cast
22007	invalid_datetime_format
22019	invalid_escape_character
2200D	invalid_escape_octet

######	###
22025	invalid_escape_sequence
22P06	nonstandard_use_of_escape_character
22010	invalid_indicator_parameter_value
22023	invalid_parameter_value
22013	invalid_preceding_or_following_size
2201B	invalid_regular_expression
2201W	invalid_row_count_in_limit_clause
2201X	invalid_row_count_in_result_offset_clause
2202H	invalid_tablesample_argument
2202G	invalid_tablesample_repeat
22009	invalid_time_zone_displacement_value
2200C	invalid_use_of_escape_character
2200G	most_specific_type_mismatch
22004	null_value_not_allowed
22002	null_value_no_indicator_parameter
22003	numeric_value_out_of_range
2200H	sequence_generator_limit_exceeded
22026	string_data_length_mismatch
22001	string_data_right_truncation
22011	substring_error
22027	trim_error
22024	unterminated_c_string
2200F	zero_length_character_string
22P01	floating_point_exception
22P02	invalid_text_representation
22P03	invalid_binary_representation
22P04	bad_copy_file_format
22P05	untranslatable_character
2200L	not_an_xml_document
2200M	invalid_xml_document
2200N	invalid_xml_content
2200S	invalid_xml_comment
2200T	invalid_xml_processing_instruction
22030	duplicate_json_object_key_value
22031	invalid_argument_for_sql_json_datetime_function
22032	invalid_json_text
22033	invalid_sql_json_subscript
22034	more_than_one_sql_json_item
22035	no_sql_json_item
22036	non_numeric_sql_json_item
22037	non_unique_keys_in_a_json_object

######	###
22038	singleton_sql_json_item_required
22039	sql_json_array_not_found
2203A	sql_json_member_not_found
2203B	sql_json_number_not_found
2203C	sql_json_object_not_found
2203D	too_many_json_array_elements
2203E	too_many_json_object_members
2203F	sql_json_scalar_required
### 23 — #######	
23000	integrity_constraint_violation
23001	restrict_violation
23502	not_null_violation
23503	foreign_key_violation
23505	unique_violation
23514	check_violation
23P01	exclusion_violation
### 24 — ########	
24000	invalid_cursor_state
### 25 — #############	
25000	invalid_transaction_state
25001	active_sql_transaction
25002	branch_transaction_already_active
25008	held_cursor_requires_same_isolation_level
25003	inappropriate_access_mode_for_branch_transaction
25004	inappropriate_isolation_level_for_branch_transaction
25005	no_active_sql_transaction_for_branch_transaction
25006	read_only_sql_transaction
25007	schema_and_data_statement_mixing_not_supported
25P01	no_active_sql_transaction
25P02	in_failed_sql_transaction
25P03	idle_in_transaction_session_timeout
### 26 — ###SQL####	
26000	invalid_sql_statement_name
### 27 — #############	
27000	triggered_data_change_violation
### 28 — #######	
28000	invalid_authorization_specification
28P01	invalid_password
### 2B — ###############	
2B000	dependent_privilege_descriptors_still_exist
2BP01	dependent_objects_still_exist

######	###
### 2D — #############	
2D000	invalid_transaction_termination
### 2F — SQL######	
2F000	sql_routine_exception
2F005	function_executed_no_return_statement
2F002	modifying_sql_data_not_permitted
2F003	prohibited_sql_statement_attempted
2F004	reading_sql_data_not_permitted
### 34 — ########	
34000	invalid_cursor_name
### 38 — #######	
38000	external_routine_exception
38001	containing_sql_not_permitted
38002	modifying_sql_data_not_permitted
38003	prohibited_sql_statement_attempted
38004	reading_sql_data_not_permitted
### 39 — ###########	
39000	external_routine_invocation_exception
39001	invalid_sqlstate_returned
39004	null_value_not_allowed
39P01	trigger_protocol_violated
39P02	srf_protocol_violated
39P03	event_trigger_protocol_violated
### 3B — ########	
3B000	savepoint_exception
3B001	invalid_savepoint_specification
### 3D — ########	
3D000	invalid_catalog_name
### 3F — ########	
3F000	invalid_schema_name
### 40 — #############	
40000	transaction_rollback
40002	transaction_integrity_constraint_violation
40001	serialization_failure
40003	statement_completion_unknown
40P01	deadlock_detected
### 42 — ################	
42000	syntax_error_or_access_rule_violation
42601	syntax_error
42501	insufficient_privilege
42846	cannot_coerce

######	###
42803	grouping_error
42P20	windowing_error
42P19	invalid_recursion
42830	invalid_foreign_key
42602	invalid_name
42622	name_too_long
42939	reserved_name
42804	datatype_mismatch
42P18	indeterminate_datatype
42P21	collation_mismatch
42P22	indeterminate_collation
42809	wrong_object_type
428C9	generated_always
42703	undefined_column
42883	undefined_function
42P01	undefined_table
42P02	undefined_parameter
42704	undefined_object
42701	duplicate_column
42P03	duplicate_cursor
42P04	duplicate_database
42723	duplicate_function
42P05	duplicate_prepared_statement
42P06	duplicate_schema
42P07	duplicate_table
42712	duplicate_alias
42710	duplicate_object
42702	ambiguous_column
42725	ambiguous_function
42P08	ambiguous_parameter
42P09	ambiguous_alias
42P10	invalid_column_reference
42611	invalid_column_definition
42P11	invalid_cursor_definition
42P12	invalid_database_definition
42P13	invalid_function_definition
42P14	invalid_prepared_statement_definition
42P15	invalid_schema_definition
42P16	invalid_table_definition
42P17	invalid_object_definition
### 44 — WITH CHECK OPTION##	

######	###
44000	with_check_option_violation
### 53 — ######	
53000	insufficient_resources
53100	disk_full
53200	out_of_memory
53300	too_many_connections
53400	configuration_limit_exceeded
### 54 — ########	
54000	program_limit_exceeded
54001	statement_too_complex
54011	too_many_columns
54023	too_many_arguments
### 55 — ###############	
55000	object_not_in_prerequisite_state
55006	object_in_use
55P02	cant_change_runtime_param
55P03	lock_not_available
55P04	unsafe_new_enum_value_usage
### 57 — #####	
57000	operator_intervention
57014	query_canceled
57P01	admin_shutdown
57P02	crash_shutdown
57P03	cannot_connect_now
57P04	database_dropped
57P05	idle_session_timeout
### 58 — #######(PostgreSQL########)	
58000	system_error
58030	io_error
58P01	undefined_file
58P02	duplicate_file
### 72 — #########	
72000	snapshot_too_old
### F0 — ########	
F0000	config_file_error
F0001	lock_file_exists
### HV — ########### (SQL/MED)	
HV000	fdw_error
HV005	fdw_column_name_not_found
HV002	fdw_dynamic_parameter_value_needed
HV010	fdw_function_sequence_error

######	###
HV021	fdw_inconsistent_descriptor_information
HV024	fdw_invalid_attribute_value
HV007	fdw_invalid_column_name
HV008	fdw_invalid_column_number
HV004	fdw_invalid_data_type
HV006	fdw_invalid_data_type_descriptors
HV091	fdw_invalid_descriptor_field_identifier
HV00B	fdw_invalid_handle
HV00C	fdw_invalid_option_index
HV00D	fdw_invalid_option_name
HV090	fdw_invalid_string_length_or_buffer_length
HV00A	fdw_invalid_string_format
HV009	fdw_invalid_use_of_null_pointer
HV014	fdw_too_many_handles
HV001	fdw_out_of_memory
HV00P	fdw_no_schemas
HV00J	fdw_option_name_not_found
HV00K	fdw_reply_handle
HV00Q	fdw_schema_not_found
HV00R	fdw_table_not_found
HV00L	fdw_unable_to_create_execution
HV00M	fdw_unable_to_create_reply
HV00N	fdw_unable_to_establish_connection
### P0 — PL/pgSQL###	
P0000	plpgsql_error
P0001	raise_exception
P0002	no_data_found
P0003	too_many_rows
P0004	assert_failure
### XX — #####	
XX000	internal_error
XX001	data_corrupted
XX002	index_corrupted

##B ##/########

PostgreSQL#######/## ######################
################################
############### ##

##

B.1. ##/########

##/#######################

1. ##

   a. ###########:################### ###################

   b. ############-########/#####2########.############################ ####
   ##################################America/New_York##

   c. #######################ISO 8601###########1999#1#13####19990113########
   ###14:15:16###141516#########

   d. ###########+############-####################################

2. ##

   a. ############################ ########B.4###################

   b. ##############################today#########Thursday#######
   #January##########at#on#######################

   c. #######################

3. ##############################

   a. #####8####6##19990118##
   #990118########### ######YYYYMMDD###YYMMDD###

   b. #######3################################

   c. 4####6######################HHMM###HHMMSS#########

   d. 3## ######################yy-mm-
   dd#########

   e. ###################DateStyle###mm-dd-yy#dd-mm-yy#yy-mm-dd###########
   ## ##############################

4. ##BC#######################1##### ######################1BC####1##########
   ###

5. BC################2#######4############# ########70######2000###############
   ##1900########

###
######99##0099##################99##############4##############

B.2.

##/## #####2#31##################

#########UTC############################ ####TimeZone######America/New_York#########################

```
=> SELECT '2018-03-11 02:30'::timestamptz;
      timestamptz
------------------------
 2018-03-11 03:30:00-04
(1 row)
```

###################(spring-forward transition)#########2:30AM######## 2AM EST## 3AM EDT############## PostgreSQL########(UTC-5)####################3:30AM EDT (UTC-4)##########

##############(fall-back transition)###########

```
=> SELECT '2018-11-04 01:30'::timestamptz;
      timestamptz
------------------------
 2018-11-04 01:30:00-05
(1 row)
```

#####1:30AM################## ##1:30AM EDT####2 AM EDT##1 AM EST#######1#### 1:30AM EST###### ####PostgreSQL########(UTC-5)#################### ############ ##############

```
=> SELECT '2018-11-04 01:30 EDT'::timestamptz;
      timestamptz
------------------------
 2018-11-04 01:30:00-04
(1 row)
```

############################## ##UTC ############## ##UTC## ############ ##

###########UTC############################UTC#####################UTC####### ########## #################UTC##########UTC##########################

B.3. ##/########

B.1####################

#B.1

#	###
January	Jan
February	Feb
March	Mar
April	Apr
May	

#	###
June	Jun
July	Jul
August	Aug
September	Sep#Sept
October	Oct
November	Nov
December	Dec

B.2#######################

#B.2

##	###
Sunday	Sun
Monday	Mon
Tuesday	Tue#Tues
Wednesday	Wed#Weds
Thursday	Thu#Thur#Thurs
Friday	Fri
Saturday	Sat

B.3#######################

#B.3 ##/###########

###	##
AM	12:00#####
AT	##############
JULIAN, JD, J	#############
ON	##############
PM	12:00#12:00#####
T	#########

B.4. ##/#########

#######################PostgreSQL#################################### timezone_abbreviations####################### ##################################### ################ #######################.../share/timezonesets/######### ## ##

timezone_abbreviations## .../share/timezonesets/#######################
#################### #timezone_abbreviations############################
#################### ######################################

###################################### ########################

zone_abbreviation offset
zone_abbreviation offset D

zone_abbreviation time_zone_name
@INCLUDE *file_name*
@OVERRIDE

zone_abbreviation############## *offset*#UTC###################### ####### ################### #####-18000#########5######################### D######### ##########################

#####IANA#######################*time_zone_name*############ ############## ## ################################ ### ################### ########### ################################ ############################

> ###
>
> ################################UTC################################## ####*offset*#############

@INCLUDE######.../share/timezonesets/######################### ######### #################

@OVERRIDE## ##### ################################

###############Default#################################### ###Australia## #India#################### #########Default################ ############## ########

###################Africa.txt#America.txt########## #####IANA############ ############################## ## ###### ######################timezone_abbreviations#######################

> ##
>
> ## ########################## #####

> ##
>
> ####################PostgreSQL###################### #####Australia#### ####SAT###################### ############SAT####################

> ##
>
> .../share/timezonesets/########################## ################## ##########

B.5. POSIX######

PostgreSQL#TZ#######POSIX############################## POSIX############## ##

POSIX##################

STD offset [*DST* [*dstoffset*] [, *rule*]]

(##) ##############

- *STD*####################
- *offset*#UTC##############
- *DST*################### ##UTC### ###########
- *dstoffset*#UTC############## ################################ *offset*#### #######################
- ##########*rule*####################

##############EST########<UTC-05>########################## ############## ### ######################B.4######### ##########

###########UTC######################## ######*hh*[:*mm*[:*ss*]]#################### #########(+ #### -)# ########################(#####PostgreSQL#######ISO-8601##### #################) *hh*###########*mm*#*ss*##################

########*rule*################

dstdate [/ *dsttime*] , *stddate* [/ *stdtime*]

(##########################) #########*dstdate*#*dsttime*################## #*stddate*#*stdtime*###### (################################) ################ ######

n

#############0##364######365#######

J*n*

######*n*#1##365######2#29################ (######2#29######################## ## ####2########################## ## #######)

M*m.n.d*

############################## *m*#1##12########## *n*#*n*########*d*########### *n*# ###1##5##5##################(4##################)# *d*####0##6##0########## ## ##M3.2.0##3##############

> ##
>
> M############################ ## (IANA############)################################

##############02:00:00###

##################rule#######################2020#################
#M3.2.0,M11.1.0(3###2#############11###1##########################2#######)
######## ######2007####################################

#####[95.198330]#####CET-1CEST,M3.5.0,M10.5.0/3#(2020####)##################
###########CET########UTC#####(#)################CEST############UTC######
#########3#######AM2######10########3AM CEST#######

4#######EST5EDT#CST6CDT#MST7MDT#PST8PDT#POSIX########### ####(######
#)IANA## #############POSIX#######
##

#######################POSIX############################ ####SET TIMEZONE
TO FOOBAR0################UTC#############

B.6.

##SQL## PostgreSQL#
##SQL### #######################

##########45###########Julius Caesar############### ##################1582#######
######### ###########365#+1/4#=365.25######### ######128###1#########

###########13##Gregory XIII###########Council of Trent######################### ####
#########365+97/400#=365.2425######### ################1#########3,300#######

365+97/400################400###97#######################

4################
####100#################
####400##################

######1700#1800#1900#2100#2200#############1600#2000#2400######## ##############
#4##################

1582#2#######1582##10###10################10#4#####10#15####### ##############
#################### ##20##
################ 1752## ######1752#9#2###
#1752#9#14######### #########Unix#####cal#########################

```
$ cal 9 1752
   September 1752
 S  M Tu  W Th  F  S
       1  2 14 15 16
17 18 19 20 21 22 23
24 25 26 27 28 29 30
```

##############################
######################PostgreSQL##
##

###############14######################
###2637################# ##

B.7. #####(Julian Date)

################## ############################ #############Joseph Justus
Scaliger#1540–1609################################Julius Caesar Scaliger#1484–1558###
###########

############JD 0############################## JD 0##########4713#1#1############
###4714#11#24######## ## #######0#####0
########UTC#####UTC#####1##### JD 0####4714#11#24##UTC######4714#11#25##UTC
#####24###############

PostgreSQL###
############### PostgreSQL##################0#####0################

##UTC+12####### #####

```
=> SELECT extract(julian from '2021-06-23 7:00:00-04'::timestamptz at time
 zone 'UTC+12');
          extract
-------------------------------
 2459388.95833333333333333333
(1 row)
=> SELECT extract(julian from '2021-06-23 8:00:00-04'::timestamptz at time
 zone 'UTC+12');
              extract
--------------------------------------
 2459389.00000000000000000000000000
(1 row)
=> SELECT extract(julian from date '2021-06-23');
 extract
---------
 2459389
(1 row)
```

##C SQL#####

C.1###SQL#PostgreSQL 17beta3########################### ##########4.1.1###### ##########SQL#######################SQL-92########## ###########################

SQL################## ## ### ###############SQL##

PostgreSQL#################### ### ######## SQL########### PostgreSQL#######################################CHECK###########SELECT 55 AS CHECK# #############

C.1#PostgreSQL## ## ########################### ## ################################ ################################## ############################### ## ###########PostgreSQL###############

#####7.3.2##########################AS#######################################AS############### #########AS###############

###

C.1######PostgreSQL##

#C.1 SQL#####

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
A		###	###	
ABORT	###			
ABS		##	##	
ABSENT	###	##	##	
ABSOLUTE	###	###	###	##
ACCESS	###			
ACCORDING		###	###	
ACOS		##	##	
ACTION	###	###	###	##
ADA		###	###	###
ADD	###	###	###	##
ADMIN	###	###	###	
AFTER	###	###	###	
AGGREGATE	###			
ALL	##	##	##	##
ALLOCATE		##	##	##
ALSO	###			
ALTER	###	##	##	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
ALWAYS	###	###	###	
ANALYSE	##			
ANALYZE	##			
AND	##	##	##	##
ANY	##	##	##	##
ANY_VALUE		##		
ARE		##	##	##
ARRAY	##, requires AS	##	##	
ARRAY_AGG		##	##	
ARRAY_MAX_CARDINALITY		##	##	
AS	##, requires AS	##	##	##
ASC	##	###	###	##
ASENSITIVE	###	##	##	
ASIN		##	##	
ASSERTION	###	###	###	##
ASSIGNMENT	###	###	###	
ASYMMETRIC	##	##	##	
AT	###	##	##	##
ATAN		##	##	
ATOMIC	###	##	##	
ATTACH	###			
ATTRIBUTE	###	###	###	
ATTRIBUTES		###	###	
AUTHORIZATION	##(############)	##	##	##
AVG		##	##	##
BACKWARD	###			
BASE64		###	###	
BEFORE	###	###	###	
BEGIN	###	##	##	##
BEGIN_FRAME		##	##	
BEGIN_PARTITION		##	##	
BERNOULLI		###	###	
BETWEEN	###(#############)	##	##	##
BIGINT	###(#############)	##	##	
BINARY	##(############)	##	##	
BIT	###(#############)			##
BIT_LENGTH				##
BLOB		##	##	
BLOCKED		###	###	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
BOM		###	###	
BOOLEAN	###(#############)	##	##	
BOTH	##	##	##	##
BREADTH	###	###	###	
BTRIM		##		
BY	###	##	##	##
C		###	###	###
CACHE	###			
CALL	###	##	##	
CALLED	###	##	##	
CARDINALITY		##	##	
CASCADE	###	###	###	##
CASCADED	###	##	##	##
CASE	##	##	##	##
CAST	##	##	##	##
CATALOG	###	###	###	##
CATALOG_NAME		###	###	###
CEIL		##	##	
CEILING		##	##	
CHAIN	###	###	###	
CHAINING		###	###	
CHAR	###(#############), requires AS	##	##	##
CHARACTER	###(#############), requires AS	##	##	##
CHARACTERISTICS	###	###	###	
CHARACTERS		###	###	
CHARACTER_LENGTH		##	##	##
CHARACTER_SET_CATALOG		###	###	###
CHARACTER_SET_NAME		###	###	###
CHARACTER_SET_SCHEMA		###	###	###
CHAR_LENGTH		##	##	##
CHECK	##	##	##	##
CHECKPOINT	###			
CLASS	###			
CLASSIFIER		##	##	
CLASS_ORIGIN		###	###	###
CLOB		##	##	
CLOSE	###	##	##	##
CLUSTER	###			

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
COALESCE	###(#############)	##	##	##
COBOL		###	###	###
COLLATE	##	##	##	##
COLLATION	##(############)	###	###	##
COLLATION_CATALOG		###	###	###
COLLATION_NAME		###	###	###
COLLATION_SCHEMA		###	###	###
COLLECT		##	##	
COLUMN	##	##	##	##
COLUMNS	###	###	###	
COLUMN_NAME		###	###	###
COMMAND_FUNCTION		###	###	###
COMMAND_FUNCTION_CODE		###	###	
COMMENT	###			
COMMENTS	###			
COMMIT	###	##	##	##
COMMITTED	###	###	###	###
COMPRESSION	###			
CONCURRENTLY	##(############)			
CONDITION		##	##	
CONDITIONAL	###	###	###	
CONDITION_NUMBER		###	###	###
CONFIGURATION	###			
CONFLICT	###			
CONNECT		##	##	##
CONNECTION	###	###	###	##
CONNECTION_NAME		###	###	###
CONSTRAINT	##	##	##	##
CONSTRAINTS	###	###	###	##
CONSTRAINT_CATALOG		###	###	###
CONSTRAINT_NAME		###	###	###
CONSTRAINT_SCHEMA		###	###	###
CONSTRUCTOR		###	###	
CONTAINS		##	##	
CONTENT	###	###	###	
CONTINUE	###	###	###	##
CONTROL		###	###	
CONVERSION	###			
CONVERT		##	##	##
COPARTITION		###		

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
COPY	###	##	##	
CORR		##	##	
CORRESPONDING		##	##	##
COS		##	##	
COSH		##	##	
COST	###			
COUNT		##	##	##
COVAR_POP		##	##	
COVAR_SAMP		##	##	
CREATE	##, requires AS	##	##	##
CROSS	##(############)	##	##	##
CSV	###			
CUBE	###	##	##	
CUME_DIST		##	##	
CURRENT	###	##	##	##
CURRENT_CATALOG	##	##	##	
CURRENT_DATE	##	##	##	##
CURRENT_DEFAULT_ TRANSFORM_GROUP		##	##	
CURRENT_PATH		##	##	
CURRENT_ROLE	##	##	##	
CURRENT_ROW		##	##	
CURRENT_SCHEMA	##(############)	##	##	
CURRENT_TIME	##	##	##	##
CURRENT_TIMESTAMP	##	##	##	##
CURRENT_TRANSFORM_ GROUP_FOR_TYPE		##	##	
CURRENT_USER	##	##	##	##
CURSOR	###	##	##	##
CURSOR_NAME		###	###	###
CYCLE	###	##	##	
DATA	###	###	###	###
DATABASE	###			
DATALINK		##	##	
DATE		##	##	##
DATETIME_INTERVAL_CODE		###	###	###
DATETIME_INTERVAL_ PRECISION		###	###	###
DAY	###, requires AS	##	##	##
DB		###	###	
DEALLOCATE	###	##	##	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
DEC	###(#############)	##	##	##
DECFLOAT		##	##	
DECIMAL	###(#############)	##	##	##
DECLARE	###	##	##	##
DEFAULT	##	##	##	##
DEFAULTS	###	###	###	
DEFERRABLE	##	###	###	##
DEFERRED	###	###	###	##
DEFINE		##	##	
DEFINED		###	###	
DEFINER	###	###	###	
DEGREE		###	###	
DELETE	###	##	##	##
DELIMITER	###			
DELIMITERS	###			
DENSE_RANK		##	##	
DEPENDS	###			
DEPTH	###	###	###	
DEREF		##	##	
DERIVED		###	###	
DESC	##	###	###	##
DESCRIBE		##	##	##
DESCRIPTOR		###	###	##
DETACH	###			
DETERMINISTIC		##	##	
DIAGNOSTICS		###	###	##
DICTIONARY	###			
DISABLE	###			
DISCARD	###			
DISCONNECT		##	##	##
DISPATCH		###	###	
DISTINCT	##	##	##	##
DLNEWCOPY		##	##	
DLPREVIOUSCOPY		##	##	
DLURLCOMPLETE		##	##	
DLURLCOMPLETEONLY		##	##	
DLURLCOMPLETEWRITE		##	##	
DLURLPATH		##	##	
DLURLPATHONLY		##	##	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
DLURLPATHWRITE		##	##	
DLURLSCHEME		##	##	
DLURLSERVER		##	##	
DLVALUE		##	##	
DO	##			
DOCUMENT	###	###	###	
DOMAIN	###	###	###	##
DOUBLE	###	##	##	##
DROP	###	##	##	##
DYNAMIC		##	##	
DYNAMIC_FUNCTION		###	###	###
DYNAMIC_FUNCTION_CODE		###	###	
EACH	###	##	##	
ELEMENT		##	##	
ELSE	##	##	##	##
EMPTY	###	##	##	
ENABLE	###			
ENCODING	###	###	###	
ENCRYPTED	###			
END	##	##	##	##
END-EXEC		##	##	##
END_FRAME		##	##	
END_PARTITION		##	##	
ENFORCED		###	###	
ENUM	###			
EQUALS		##	##	
ERROR	###	###	###	
ESCAPE	###	##	##	##
EVENT	###			
EVERY		##	##	
EXCEPT	##, requires AS	##	##	##
EXCEPTION				##
EXCLUDE	###	###	###	
EXCLUDING	###	###	###	
EXCLUSIVE	###			
EXEC		##	##	##
EXECUTE	###	##	##	##
EXISTS	###(#############)	##	##	##
EXP		##	##	
EXPLAIN	###			

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
EXPRESSION	###	###	###	
EXTENSION	###			
EXTERNAL	###	##	##	##
EXTRACT	###(#############)	##	##	##
FALSE	##	##	##	##
FAMILY	###			
FETCH	##, requires AS	##	##	##
FILE		###	###	
FILTER	###, requires AS	##	##	
FINAL		###	###	
FINALIZE	###			
FINISH		###	###	
FIRST	###	###	###	##
FIRST_VALUE		##	##	
FLAG		###	###	
FLOAT	###(#############)	##	##	##
FLOOR		##	##	
FOLLOWING	###	###	###	
FOR	##, requires AS	##	##	##
FORCE	###			
FOREIGN	##	##	##	##
FORMAT	###	###	###	
FORTRAN		###	###	###
FORWARD	###			
FOUND		###	###	##
FRAME_ROW		##	##	
FREE		##	##	
FREEZE	##(############)			
FROM	##, requires AS	##	##	##
FS		###	###	
FULFILL		###	###	
FULL	##(############)	##	##	##
FUNCTION	###	##	##	
FUNCTIONS	###			
FUSION		##	##	
G		###	###	
GENERAL		###	###	
GENERATED	###	###	###	
GET		##	##	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
GLOBAL	###	##	##	##
GO		###	###	##
GOTO		###	###	##
GRANT	##, requires AS	##	##	##
GRANTED	###	###	###	
GREATEST	###(#############)	##		
GROUP	##, requires AS	##	##	##
GROUPING	###(#############)	##	##	
GROUPS	###	##	##	
HANDLER	###			
HAVING	##, requires AS	##	##	##
HEADER	###			
HEX		###	###	
HIERARCHY		###	###	
HOLD	###	##	##	
HOUR	###, requires AS	##	##	##
ID		###	###	
IDENTITY	###	##	##	##
IF	###			
IGNORE		###	###	
ILIKE	##(############)			
IMMEDIATE	###	###	###	##
IMMEDIATELY		###	###	
IMMUTABLE	###			
IMPLEMENTATION		###	###	
IMPLICIT	###			
IMPORT	###	##	##	
IN	##	##	##	##
INCLUDE	###			
INCLUDING	###	###	###	
INCREMENT	###	###	###	
INDENT	###	###	###	
INDEX	###			
INDEXES	###			
INDICATOR		##	##	##
INHERIT	###			
INHERITS	###			
INITIAL		##	##	
INITIALLY	##	###	###	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
INLINE	###			
INNER	##(############)	##	##	##
INOUT	###(#############)	##	##	
INPUT	###	###	###	##
INSENSITIVE	###	##	##	##
INSERT	###	##	##	##
INSTANCE		###	###	
INSTANTIABLE		###	###	
INSTEAD	###	###	###	
INT	###(#############)	##	##	##
INTEGER	###(#############)	##	##	##
INTEGRITY		###	###	
INTERSECT	##, requires AS	##	##	##
INTERSECTION		##	##	
INTERVAL	###(#############)	##	##	##
INTO	##, requires AS	##	##	##
INVOKER	###	###	###	
IS	##(############)	##	##	##
ISNULL	##(############), requires AS			
ISOLATION	###	###	###	##
JOIN	##(############)	##	##	##
JSON	###(#############)	##		
JSON_ARRAY	###(#############)	##	##	
JSON_ARRAYAGG	###(#############)	##	##	
JSON_EXISTS	###(#############)	##	##	
JSON_OBJECT	###(#############)	##	##	
JSON_OBJECTAGG	###(#############)	##	##	
JSON_QUERY	###(#############)	##	##	
JSON_SCALAR	###(#############)	##		
JSON_SERIALIZE	###(#############)	##		

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
JSON_TABLE	###(#############)	##	##	
JSON_TABLE_PRIMITIVE		##	##	
JSON_VALUE	###(#############)	##	##	
K		###	###	
KEEP	###	###	###	
KEY	###	###	###	##
KEYS	###	###	###	
KEY_MEMBER		###	###	
KEY_TYPE		###	###	
LABEL	###			
LAG		##	##	
LANGUAGE	###	##	##	##
LARGE	###	##	##	
LAST	###	###	###	##
LAST_VALUE		##	##	
LATERAL	##	##	##	
LEAD		##	##	
LEADING	##	##	##	##
LEAKPROOF	###			
LEAST	###(#############)	##		
LEFT	##(############)	##	##	##
LENGTH		###	###	###
LEVEL	###	###	###	##
LIBRARY		###	###	
LIKE	##(############)	##	##	##
LIKE_REGEX		##	##	
LIMIT	##, requires AS	###	###	
LINK		###	###	
LISTAGG		##	##	
LISTEN	###			
LN		##	##	
LOAD	###			
LOCAL	###	##	##	##
LOCALTIME	##	##	##	
LOCALTIMESTAMP	##	##	##	
LOCATION	###	###	###	
LOCATOR		###	###	
LOCK	###			
LOCKED	###			

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
LOG		##	##	
LOG10		##	##	
LOGGED	###			
LOWER		##	##	##
LPAD		##		
LTRIM		##		
M		###	###	
MAP		###	###	
MAPPING	###	###	###	
MATCH	###	##	##	##
MATCHED	###	###	###	
MATCHES		##	##	
MATCH_NUMBER		##	##	
MATCH_RECOGNIZE		##	##	
MATERIALIZED	###			
MAX		##	##	##
MAXVALUE	###	###	###	
MEASURES		###	###	
MEMBER		##	##	
MERGE	###	##	##	
MERGE_ACTION	###(#############)			
MESSAGE_LENGTH		###	###	###
MESSAGE_OCTET_LENGTH		###	###	###
MESSAGE_TEXT		###	###	###
METHOD	###	##	##	
MIN		##	##	##
MINUTE	###, requires AS	##	##	##
MINVALUE	###	###	###	
MOD		##	##	
MODE	###			
MODIFIES		##	##	
MODULE		##	##	##
MONTH	###, requires AS	##	##	##
MORE		###	###	###
MOVE	###			
MULTISET		##	##	
MUMPS		###	###	###
NAME	###	###	###	###
NAMES	###	###	###	##
NAMESPACE		###	###	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
NATIONAL	###(#############)	##	##	##
NATURAL	##(############)	##	##	##
NCHAR	###(#############)	##	##	##
NCLOB		##	##	
NESTED	###	###	###	
NESTING		###	###	
NEW	###	##	##	
NEXT	###	###	###	##
NFC	###	###	###	
NFD	###	###	###	
NFKC	###	###	###	
NFKD	###	###	###	
NIL		###	###	
NO	###	##	##	##
NONE	###(#############)	##	##	
NORMALIZE	###(#############)	##	##	
NORMALIZED	###	###	###	
NOT	##	##	##	##
NOTHING	###			
NOTIFY	###			
NOTNULL	##(############), requires AS			
NOWAIT	###			
NTH_VALUE		##	##	
NTILE		##	##	
NULL	##	##	##	##
NULLABLE		###	###	###
NULLIF	###(#############)	##	##	##
NULLS	###	###	###	
NULL_ORDERING		###	###	
NUMBER		###	###	###
NUMERIC	###(#############)	##	##	##
OBJECT	###	###	###	
OCCURRENCE		###	###	
OCCURRENCES_REGEX		##	##	
OCTETS		###	###	
OCTET_LENGTH		##	##	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
OF	###	##	##	##
OFF	###	###	###	
OFFSET	##, requires AS	##	##	
OIDS	###			
OLD	###	##	##	
OMIT	###	##	##	
ON	##, requires AS	##	##	##
ONE		##	##	
ONLY	##	##	##	##
OPEN		##	##	##
OPERATOR	###			
OPTION	###	###	###	##
OPTIONS	###	###	###	
OR	##	##	##	##
ORDER	##, requires AS	##	##	##
ORDERING		###	###	
ORDINALITY	###	###	###	
OTHERS	###	###	###	
OUT	###(#############)	##	##	
OUTER	##(############)	##	##	##
OUTPUT		###	###	##
OVER	###, requires AS	##	##	
OVERFLOW		###	###	
OVERLAPS	##(############), requires AS	##	##	##
OVERLAY	###(#############)	##	##	
OVERRIDING	###	###	###	
OWNED	###			
OWNER	###			
P		###	###	
PAD		###	###	##
PARALLEL	###			
PARAMETER	###	##	##	
PARAMETER_MODE		###	###	
PARAMETER_NAME		###	###	
PARAMETER_ORDINAL_POSITION		###	###	
PARAMETER_SPECIFIC_CATALOG		###	###	
PARAMETER_SPECIFIC_NAME		###	###	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
PARAMETER_SPECIFIC_SCHEMA		###	###	
PARSER	###			
PARTIAL	###	###	###	##
PARTITION	###	##	##	
PARTITIONS	###			
PASCAL		###	###	###
PASS		###	###	
PASSING	###	###	###	
PASSTHROUGH		###	###	
PASSWORD	###			
PAST		###	###	
PATH	###	###	###	
PATTERN		##	##	
PER		##	##	
PERCENT		##	##	
PERCENTILE_CONT		##	##	
PERCENTILE_DISC		##	##	
PERCENT_RANK		##	##	
PERIOD		##	##	
PERMISSION		###	###	
PERMUTE		###	###	
PIPE		###	###	
PLACING	##	###	###	
PLAN	###	###	###	
PLANS	###			
PLI		###	###	###
POLICY	###			
PORTION		##	##	
POSITION	###(#############)	##	##	##
POSITION_REGEX		##	##	
POWER		##	##	
PRECEDES		##	##	
PRECEDING	###	###	###	
PRECISION	###(#############), requires AS	##	##	##
PREPARE	###	##	##	##
PREPARED	###			
PRESERVE	###	###	###	##
PREV		###	###	
PRIMARY	##	##	##	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
PRIOR	###	###	###	##
PRIVATE		###	###	
PRIVILEGES	###	###	###	##
PROCEDURAL	###			
PROCEDURE	###	##	##	##
PROCEDURES	###			
PROGRAM	###			
PRUNE		###	###	
PTF		##	##	
PUBLIC		###	###	##
PUBLICATION	###			
QUOTE	###			
QUOTES	###	###	###	
RANGE	###	##	##	
RANK		##	##	
READ	###	###	###	##
READS		##	##	
REAL	###(#############)	##	##	##
REASSIGN	###			
RECHECK	###			
RECOVERY		###	###	
RECURSIVE	###	##	##	
REF	###	##	##	
REFERENCES	##	##	##	##
REFERENCING	###	##	##	
REFRESH	###			
REGR_AVGX		##	##	
REGR_AVGY		##	##	
REGR_COUNT		##	##	
REGR_INTERCEPT		##	##	
REGR_R2		##	##	
REGR_SLOPE		##	##	
REGR_SXX		##	##	
REGR_SXY		##	##	
REGR_SYY		##	##	
REINDEX	###			
RELATIVE	###	###	###	##
RELEASE	###	##	##	
RENAME	###			
REPEATABLE	###	###	###	###

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
REPLACE	###			
REPLICA	###			
REQUIRING		###	###	
RESET	###			
RESPECT		###	###	
RESTART	###	###	###	
RESTORE		###	###	
RESTRICT	###	###	###	##
RESULT		##	##	
RETURN	###	##	##	
RETURNED_CARDINALITY		###	###	
RETURNED_LENGTH		###	###	###
RETURNED_OCTET_LENGTH		###	###	###
RETURNED_SQLSTATE		###	###	###
RETURNING	##, requires AS	###	###	
RETURNS	###	##	##	
REVOKE	###	##	##	##
RIGHT	##(############)	##	##	##
ROLE	###	###	###	
ROLLBACK	###	##	##	##
ROLLUP	###	##	##	
ROUTINE	###	###	###	
ROUTINES	###			
ROUTINE_CATALOG		###	###	
ROUTINE_NAME		###	###	
ROUTINE_SCHEMA		###	###	
ROW	###(#############)	##	##	
ROWS	###	##	##	##
ROW_COUNT		###	###	###
ROW_NUMBER		##	##	
RPAD		##		
RTRIM		##		
RULE	###			
RUNNING		##	##	
SAVEPOINT	###	##	##	
SCALAR	###	###	###	
SCALE		###	###	###
SCHEMA	###	###	###	##
SCHEMAS	###			
SCHEMA_NAME		###	###	###

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
SCOPE		##	##	
SCOPE_CATALOG		###	###	
SCOPE_NAME		###	###	
SCOPE_SCHEMA		###	###	
SCROLL	###	##	##	##
SEARCH	###	##	##	
SECOND	###, requires AS	##	##	##
SECTION		###	###	##
SECURITY	###	###	###	
SEEK		##	##	
SELECT	##	##	##	##
SELECTIVE		###	###	
SELF		###	###	
SEMANTICS		###	###	
SENSITIVE		##	##	
SEQUENCE	###	###	###	
SEQUENCES	###			
SERIALIZABLE	###	###	###	###
SERVER	###	###	###	
SERVER_NAME		###	###	###
SESSION	###	###	###	##
SESSION_USER	##	##	##	##
SET	###	##	##	##
SETOF	###(#############)			
SETS	###	###	###	
SHARE	###			
SHOW	###	##	##	
SIMILAR	##(############)	##	##	
SIMPLE	###	###	###	
SIN		##	##	
SINH		##	##	
SIZE		###	###	##
SKIP	###	##	##	
SMALLINT	###(#############)	##	##	##
SNAPSHOT	###			
SOME	##	##	##	##
SORT_DIRECTION		###	###	
SOURCE	###	###	###	
SPACE		###	###	##

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
SPECIFIC		##	##	
SPECIFICTYPE		##	##	
SPECIFIC_NAME		###	###	
SPLIT	###			
SQL	###	##	##	##
SQLCODE				##
SQLERROR				##
SQLEXCEPTION		##	##	
SQLSTATE		##	##	##
SQLWARNING		##	##	
SQRT		##	##	
STABLE	###			
STANDALONE	###	###	###	
START	###	##	##	
STATE		###	###	
STATEMENT	###	###	###	
STATIC		##	##	
STATISTICS	###			
STDDEV_POP		##	##	
STDDEV_SAMP		##	##	
STDIN	###			
STDOUT	###			
STORAGE	###			
STORED	###			
STRICT	###			
STRING	###	###	###	
STRIP	###	###	###	
STRUCTURE		###	###	
STYLE		###	###	
SUBCLASS_ORIGIN		###	###	###
SUBMULTISET		##	##	
SUBSCRIPTION	###			
SUBSET		##	##	
SUBSTRING	###(###########)	##	##	##
SUBSTRING_REGEX		##	##	
SUCCEEDS		##	##	
SUM		##	##	##
SUPPORT	###			
SYMMETRIC	##	##	##	
SYSID	###			

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
SYSTEM	###	##	##	
SYSTEM_TIME		##	##	
SYSTEM_USER	##	##	##	##
T		###	###	
TABLE	##	##	##	##
TABLES	###			
TABLESAMPLE	##(###########)	##	##	
TABLESPACE	###			
TABLE_NAME		###	###	###
TAN		##	##	
TANH		##	##	
TARGET	###			
TEMP	###			
TEMPLATE	###			
TEMPORARY	###	###	###	##
TEXT	###			
THEN	##	##	##	##
THROUGH		###	###	
TIES	###	###	###	
TIME	###(###########)	##	##	##
TIMESTAMP	###(###########)	##	##	##
TIMEZONE_HOUR		##	##	##
TIMEZONE_MINUTE		##	##	##
TO	##, requires AS	##	##	##
TOKEN		###	###	
TOP_LEVEL_COUNT		###	###	
TRAILING	##	##	##	##
TRANSACTION	###	###	###	##
TRANSACTIONS_COMMITTED		###	###	
TRANSACTIONS_ROLLED_BACK		###	###	
TRANSACTION_ACTIVE		###	###	
TRANSFORM	###	###	###	
TRANSFORMS		###	###	
TRANSLATE		##	##	##
TRANSLATE_REGEX		##	##	
TRANSLATION		##	##	##
TREAT	###(###########)	##	##	
TRIGGER	###	##	##	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
TRIGGER_CATALOG		###	###	
TRIGGER_NAME		###	###	
TRIGGER_SCHEMA		###	###	
TRIM	###(#############)	##	##	##
TRIM_ARRAY		##	##	
TRUE	##	##	##	##
TRUNCATE	###	##	##	
TRUSTED	###			
TYPE	###	###	###	###
TYPES	###			
UESCAPE	###	##	##	
UNBOUNDED	###	###	###	
UNCOMMITTED	###	###	###	###
UNCONDITIONAL	###	###	###	
UNDER		###	###	
UNENCRYPTED	###			
UNION	##, requires AS	##	##	##
UNIQUE	##	##	##	##
UNKNOWN	###	##	##	##
UNLINK		###	###	
UNLISTEN	###			
UNLOGGED	###			
UNMATCHED		###	###	
UNNAMED		###	###	###
UNNEST		##	##	
UNTIL	###			
UNTYPED		###	###	
UPDATE	###	##	##	##
UPPER		##	##	##
URI		###	###	
USAGE		###	###	##
USER	##	##	##	##
USER_DEFINED_TYPE_CATALOG		###	###	
USER_DEFINED_TYPE_CODE		###	###	
USER_DEFINED_TYPE_NAME		###	###	
USER_DEFINED_TYPE_SCHEMA		###	###	
USING	##	##	##	##
UTF16		###	###	
UTF32		###	###	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
UTF8		###	###	
VACUUM	###			
VALID	###	###	###	
VALIDATE	###			
VALIDATOR	###			
VALUE	###	##	##	##
VALUES	###(#############)	##	##	##
VALUE_OF		##	##	
VARBINARY		##	##	
VARCHAR	###(#############)	##	##	##
VARIADIC	##			
VARYING	###, requires AS	##	##	##
VAR_POP		##	##	
VAR_SAMP		##	##	
VERBOSE	##(############)			
VERSION	###	###	###	
VERSIONING		##	##	
VIEW	###	###	###	##
VIEWS	###			
VOLATILE	###			
WHEN	##	##	##	##
WHENEVER		##	##	##
WHERE	##, requires AS	##	##	##
WHITESPACE	###	###	###	
WIDTH_BUCKET		##	##	
WINDOW	##, requires AS	##	##	
WITH	##, requires AS	##	##	##
WITHIN	###, requires AS	##	##	
WITHOUT	###, requires AS	##	##	
WORK	###	###	###	##
WRAPPER	###	###	###	
WRITE	###	###	###	##
XML	###	##	##	
XMLAGG		##	##	
XMLATTRIBUTES	###(#############)	##	##	
XMLBINARY		##	##	
XMLCAST		##	##	
XMLCOMMENT		##	##	

#####	PostgreSQL	SQL:2023	SQL:2016	SQL-92
XMLCONCAT	###(#############)	##	##	
XMLDECLARATION		###	###	
XMLDOCUMENT		##	##	
XMLELEMENT	###(#############)	##	##	
XMLEXISTS	###(#############)	##	##	
XMLFOREST	###(#############)	##	##	
XMLITERATE		##	##	
XMLNAMESPACES	###(#############)	##	##	
XMLPARSE	###(#############)	##	##	
XMLPI	###(#############)	##	##	
XMLQUERY		##	##	
XMLROOT	###(#############)			
XMLSCHEMA		###	###	
XMLSERIALIZE	###(#############)	##	##	
XMLTABLE	###(#############)	##	##	
XMLTEXT		##	##	
XMLVALIDATE		##	##	
YEAR	###, requires AS	##	##	##
YES	###	###	###	
ZONE	###	###	###	##

##D SQL####

#####PostgreSQL##########SQL###################### ######################### ## ####################################

##SQL#######ISO/IEC 9075 #Database Language SQL#### ####SQL################### #### #######2023######## ##2023###ISO/IEC 9075:2023######SQL:2023######## #### #####SQL:2016#SQL:2011#SQL:2008#SQL:2006#SQL:2003#SQL:1999#SQL-92### ####### ### PostgreSQL################ ################SQL######################## ##SQL########################## ############# ############################

SQL-92##########Entry#Intermediate#Full###3############### ##SQL################## ##########Entry############## Intermediate####Full################################ ###############

SQL:1999#####SQL##SQL-92#######3## ##### ####################SQL################################ ################

#####################################:

- ISO/IEC 9075-1 Framework (SQL/Framework)
- ISO/IEC 9075-2 Foundation (SQL/Foundation)
- ISO/IEC 9075-3 Call Level Interface (SQL/CLI)
- ISO/IEC 9075-4 Persistent Stored Modules (SQL/PSM)
- ISO/IEC 9075-9 Management of External Data (SQL/MED)
- ISO/IEC 9075-10 Object Language Bindings (SQL/OLB)
- ISO/IEC 9075-11 Information and Definition Schemas (SQL/Schemata)
- ISO/IEC 9075-13 Routines and Types using the Java Language (SQL/JRT)
- ISO/IEC 9075-14 XML-related specifications (SQL/XML)
- ISO/IEC 9075-15 Multi-dimensional arrays (SQL/MDA)
- ISO/IEC 9075-16 Property Graph Queries (SQL/PGQ)

##

PostgreSQL###### 1#2#9#11####14############# #3###ODBC##########13###PL/Java##### ## 4#10#15###16#########PostgreSQL######## ###

PostgreSQL#SQL:2023################### #############177########PostgreSQL###### 170######## ###################### #########SQL:2023############################ ##################################

###2######PostgreSQL#########################PostgreSQL#############SQL:2023### ############### ###2################ ### ##################################### ##

##

################### ### ##################

D.1.

###	###	##	####
B012		####C	
B021		##SQL	
B128		###### SQL	
E011	##	######	
E011-01	##	NTEGER###SMALLINT####	
E011-02	##	REAL#DOUBLE PRECISION#FLOAT####	
E011-03	##	DECIMAL###NUMERIC####	
E011-04	##	#####	
E011-05	##	####	
E011-06	##	##############	
E021	##	######	
E021-01	##	CHARACTER####	
E021-02	##	CHARACTER VARYING####	
E021-03	##	######	
E021-04	##	CHARACTER_LENGTH##	#####CHARACTER# ############
E021-05	##	OCTET_LENGTH##	
E021-06	##	SUBSTRING##	
E021-07	##	#####	
E021-08	##	UPPER###LOWER##	
E021-09	##	TRIM##	
E021-10	##	#############	
E021-11	##	POSITION##	
E021-12	##	#####	
E031	##	###	
E031-01	##	#######	
E031-02	##	#######	
E031-03	##	##########	
E051	##	#########	
E051-01	##	SELECT DISTINCT	
E051-02	##	GROUP BY#	
E051-04	##	##################GROUP BY	

###	###	##	####
E051-05	##	############	
E051-06	##	HAVING#	
E051-07	##	###########*	
E051-08	##	FROM######	
E051-09	##	FROM########	
E061	##	###########	
E061-01	##	####	
E061-02	##	BETWEEN##	
E061-03	##	#########IN##	
E061-04	##	LIKE##	
E061-05	##	LIKE## ESCAPE#	
E061-06	##	NULL##	
E061-07	##	########	
E061-08	##	EXISTS##	
E061-09	##	############	
E061-11	##	IN##########	
E061-12	##	################	
E061-13	##	########	
E061-14	##	####	
E071	##	########	
E071-01	##	UNION DISTINCT#######	
E071-02	##	UNION ALL#######	
E071-03	##	EXCEPT DISTINCT#######	
E071-05	##	################################	
E071-06	##	###############	
E081	##	####	
E081-01	##	SELECT##	
E081-02	##	DELETE##	
E081-03	##	########INSERT##	
E081-04	##	########UPDATE##	
E081-05	##	#####UPDATE##	
E081-06	##	########REFERENCES##	
E081-07	##	#####REFERENCES##	
E081-08	##	WITH GRANT OPTION	
E081-09	##	USAGE##	
E081-10	##	EXECUTE##	
E091	##	####	
E091-01	##	AVG	
E091-02	##	COUNT	
E091-03	##	MAX	
E091-04	##	MIN	

###	###	##	####
E091-05	##	SUM	
E091-06	##	ALL###	
E091-07	##	DISTINCT###	
E101	##	#######	
E101-01	##	INSERT#	
E101-03	##	##UPDATE#	
E101-04	##	##DELETE#	
E111	##	###SELECT#	
E121	##	##########	
E121-01	##	DECLARE CURSOR	
E121-02	##	##############ORDER BY#	
E121-03	##	ORDER BY#####	
E121-04	##	OPEN#	
E121-06	##	####UPDATE#	
E121-07	##	####DELETE#	
E121-08	##	CLOSE#	
E121-10	##	FETCH######NEXT	
E121-17	##	WITH HOLD####	
E131	##	NULL######(########NULL)	
E141	##	#######	
E141-01	##	NOT NULL##	
E141-02	##	NOT NULL##UNIQUE##	
E141-03	##	PRIMARY KEY##	
E141-04	##	#################NO ACTION########### FOREIGN KEY##	
E141-06	##	CHECK##	
E141-07	##	######	
E141-08	##	PRIMARY KEY#######NOT NULL	
E141-10	##	###################	
E151	##	############	
E151-01	##	COMMIT#	
E151-02	##	ROLLBACK#	
E152	##	##SET TRANSACTION#	
E152-01	##	SET TRANSACTION#: ISOLATION LEVEL SERIALIZABLE#	
E152-02	##	SET TRANSACTION#: READ ONLY###READ WRITE#	
E153	##	##################	
E161	##	#############SQL####	
E171	##	SQLSTATE####	
E182	##	#########	
F021	##	########	

###	###	##	####
F021-01	##	COLUMNS###	
F021-02	##	TABLES###	
F021-03	##	VIEWS###	
F021-04	##	TABLE_CONSTRAINTS###	
F021-05	##	REFERENTIAL_CONSTRAINTS###	
F021-06	##	CHECK_CONSTRAINTS###	
F031	##	########	
F031-01	##	#############CREATE TABLE#	
F031-02	##	CREATE VIEW#	
F031-03	##	GRANT#	
F031-04	##	ALTER TABLE#: ADD COLUMN#	
F031-13	##	DROP TABLE#: RESTRICT#	
F031-16	##	DROP VIEW#: RESTRICT#	
F031-19	##	REVOKE#: RESTRICT#	
F032		CASCADE####	
F033		ALTER TABLE#: DROP COLUMN#	
F034		##REVOKE#	
F035		CASCADE###REVOKE	
F036		##########REVOKE#	
F037		REVOKE#: GRANT OPTION FOR#	
F038		WITH GRANT OPTION###REVOKE	
F041	##	########	
F041-01	##	####(INNER########)	
F041-02	##	INNER#####	
F041-03	##	LEFT OUTER JOIN	
F041-04	##	RIGHT OUTER JOIN	
F041-05	##	###########	
F041-07	##	##############################	
F041-08	##	##############(###=###)	
F051	##	######	
F051-01	##	DATE####(DATE############)	
F051-02	##	#####0#########TIME####(TIME############)	
F051-03	##	#####0#6#########TIMESTAMP#### (TIMESTAMP############)	
F051-04	##	DATE#TIME#TIMESTAMP############	
F051-05	##	################CAST	
F051-06	##	CURRENT_DATE	
F051-07	##	LOCALTIME	
F051-08	##	LOCALTIMESTAMP	
F052		#############	
F053		OVERLAPS##	

###	###	##	####
F081	##	####UNION###EXCEPT	
F111		SERIALIZABLE########	
F112		#####READ UNCOMMITTED	
F113		#####READ COMMITTED	
F114		#####REPEATABLE READ	
F131	##	#######	
F131-01	##	##########################WHERE#GROUP BY#HAVING#	
F131-02	##	##################################	
F131-03	##	###############################	
F131-04	##	GROUP BY#HAVING######################	
F131-05	##	GROUP BY#HAVING################### SELECT	
F171		###########	
F181	##	###########	
F191		######	
F200		TRUNCATE TABLE#	
F201	##	CAST##	
F202		TRUNCATE TABLE: identity column restart#####	
F221	##	#########	
F222		INSERT#: DEFAULT VALUES#	
F231		######	
F251		########	
F261	##	CASE#	
F261-01	##	##CASE	
F261-02	##	##CASE	
F261-03	##	NULLIF	
F261-04	##	COALESCE	
F262		##CASE#	
F271		########	
F281		LIKE##	
F292		UNIQUE null###	
F302		INTERSECT#######	
F303		INTERSECT DISTINCT#######	
F302-02		INTERSECT ALL#######	
F304		EXCEPT ALL#######	
F305		INTERSECT ALL#######	
F311	##	#######	
F311-01	##	CREATE SCHEMA	
F311-02	##	##########CREATE TABLE	
F311-03	##	CREATE VIEW	

###	###	##	####
F311-04	##	CREATE VIEW: WITH CHECK OPTION	
F311-05	##	GRANT#	
F312		MERGE#	
F313		##MERGE#	
F314		DELETE#####MERGE#	
F321		#####	
F341		########	
F361		###########	
F381		########	
F382		#######	
F383		Set column not null#	
F384		Drop identity property#	
F385		Drop column generation expression#	
F386		Set identity column generation#	
F387		ALTER TABLE#: ALTER COLUMN#	
F388		ALTER TABLE#: ADD/DROP CONSTRAINT#	
F391		#####	
F392		###Unicode#####	
F393		######Unicode#####	
F394		#############	
F401		########	
F402		LOB###################	
F404		#############	
F405		NATURAL JOIN	
F406		FULL OUTER JOIN	
F407		CROSS JOIN	
F411		#####	##############
F421		####	
F431		###################	
F432		####NEXT###FETCH	
F433		FETCH FIRST	
F434		FETCH LAST	
F435		FETCH PRIOR	
F436		FETCH ABSOLUTE	
F437		FETCH RELATIVE	
F438		############	
F441		###########	
F442		#############	
F471	##	##########	
F481	##	##NULL##	
F491		####	

###	###	##	####
F501	##	########	
F501-01	##	SQL_FEATURES###	
F501-02	##	SQL_SIZING###	
F502		############	
F531		######	
F555		######	
F561		#####	
F571		####	
F591		######	
F611		#######	
F641		#############	
F651		########	
F661		######	
F672		#######	
F690		######	
F692		########	
F701		######	
F711		#####ALTER	
F731		##INSERT##	
F751		###CHECK##	
F761		#######	
F762		CURRENT_CATALOG	
F763		CURRENT_SCHEMA	
F771		####	
F781		######	
F791		Insensitive####	
F801		#######	
F850		#############ORDER BY	
F851		##########ORDER BY	
F852		##########ORDER BY	
F855		###############ORDER BY	
F856		###############FETCH FIRST	
F857		#############FETCH FIRST	
F858		##########FETCH FIRST	
F859		##########FETCH FIRST	
F860		##FETCH FIRST######	
F861		#############OFFSET	
F862		##########OFFSET	
F863		###############OFFSET	
F864		##########OFFSET	
F865		OFFSET########	

###	###	##	####
F867		FETCH FIRST#: WITH TIES#####	
F868		##############ORDER BY	
F869		SQL####	
S071		####SQL##########	
S090		########	
S092		#########	
S095		#############	
S096		#########	
S098		ARRAY_AGG	
S099		####	
S111		########ONLY	
S201		######SQL#######	
S203		#######	
S204		############	
S211		###########	
S301		##UNNEST	
S404		TRIM_ARRAY	
T031		BOOLEAN####	
T054		GREATEST#LEAST	NULL#########
T055		##########	
T056		####TRIM##	
T061		UCS####	
T071		BIGINT####	
T081		#############	
T121		########WITH(RECURSIVE###)	
T122		########WITH(RECURSIVE###)	
T131		#######	
T132		###############	
T133		#######	
T141		SIMILAR##	
T151		DISTINCT##	
T152		####DISTINCT##	
T171		########LIKE#	
T172		########AS#######	
T173		##########LIKE#	
T174		###	
T177		#########: #############	
T178		###: #############	
T191		#####RESTRICT	
T201		###############	
T212		#######	

###	###	##	####
T213		INSTEAD OF###	
T214		BEFORE###	
T215		AFTER###	
T216		######################	
T217		TRIGGER##	
T241		START TRANSACTION#	
T261		#########	
T271		######	
T281		####SELECT##	
T285		######	
T312		OVERLAY##	
T321-01	##	################	
T321-02	##	#########################	
T321-03	##	######	
T321-04	##	CALL#	
T321-05	##	RETURN#	
T321-06	##	ROUTINES###	
T321-07	##	PARAMETERS###	
T323		###################	
T325		####SQL#######	
T331		#####	
T332		#####	
T341		SQL###########SQL######################	
T351		######	
T431		########	
T432		#######GROUPING SETS	
T433		#####GROUPING##	
T434		GROUP BY DISTINCT	
T441		ABS###MOD##	
T461		##BETWEEN##	
T491		LATERAL######	
T501		##EXISTS##	
T521		CALL###########	
T523		SQL###########INOUT###############	
T524		CALL#######################	
T525		SQL######################	
T551		###################	
T581		###########	
T591		NULL###########UNIQUE##	
T611		##OLAP##	
T612		##OLAP##	

###	###	##	####
T613		######	
T614		NTILE##	
T615		LEAD###LAG##	
T617		FIRST_VALUE###LAST_VALUE##	
T620		WINDOW#: GROUPS#####	
T621		######	
T622		####	
T623		#######	
T624		######	
T626		ANY_VALUE	
T627		COUNT DISTINCT#############	
T631	##	#######IN##	
T651		SQL######SQL#####	
T653		########SQL#####	
T655		########	
T661		#10#######	
T662		###############	
T670		############	
T803		#######JSON	
T811		##SQL/JSON####	
T812		SQL/JSON: JSON_OBJECTAGG	
T813		SQL/JSON: ORDER BY###JSON_ARRAYAGG	
T814		JSON_OBJECT#JSON_OBJECTAGG#####	
T821		##SQL/JSON########	
T822		SQL/JSON: IS JSON WITH UNIQUE KEYS##	
T823		SQL/JSON: PASSING#	
T825		SQL/JSON: ON EMPTY#ON ERROR#	
T826		ON ERROR###ON EMPTY#######	
T827		JSON_TABLE: ###NESTED COLUMNS#	
T828		JSON_QUERY	
T829		JSON_QUERY: #########	
T830		SQL/JSON##############	
T831		SQL/JSON####: #####	
T832		SQL/JSON####: ######	
T833		SQL/JSON####: #####	
T834		SQL/JSON####: ##############	
T835		SQL/JSON####: ######	
T836		SQL/JSON####: starts with##	
T837		SQL/JSON####: regex_like##	
T840		SQL/JSON######16#######	
T851		SQL/JSON: ###################	

###	###	##	####
T865		SQL/JSON######: bigint()	
T866		SQL/JSON######: boolean()	
T867		SQL/JSON######: date()	
T868		SQL/JSON######: decimal()	
T869		SQL/JSON######: ########decimal()	
T870		SQL/JSON######: integer()	
T871		SQL/JSON######: number()	
T872		SQL/JSON######: string()	
T873		SQL/JSON######: time()	
T874		SQL/JSON######: time_tz()	
T875		SQL/JSON######: time##	
T876		SQL/JSON######: timestamp()	
T877		SQL/JSON######: timestamp_tz()	
T879		#####JSON	jsonb##
T880		########JSON	jsonb##
X010		XML#	
X011		XML####	
X014		XML####	
X016		##XML#	
X020		XMLConcat	
X031		XMLElement	
X032		XMLForest	
X034		XMLAgg	
X035		XMLAgg: ORDER BY#####	
X036		XMLComment	
X037		XMLPI	
X038		XMLText	supported except for RETURNING
X040		##########	
X041		##########: NULL###	
X042		##########: NULL#nil#####	
X043		##########: ###############	
X044		##########: ############	
X045		##########: ########	
X046		##########: #######	
X047		##########: #########	
X048		##########: ######BASE64###	
X049		##########: ######HEX###	
X050		##########	
X051		##########: NULL###	
X052		##########: NULL#nil#####	

###	###	##	####
X053		##########: ###############	
X054		##########: ############	
X055		##########: ######	
X056		##########: #######	
X057		##########: #########	
X058		##########: ######BASE64###	
X059		##########: ######HEX###	
X060		XMLParse: ########CONTENT#####	
X061		XMLParse: ########DOCUMENT#####	
X069		XMLSerialize: INDENT	
X070		XMLSerialize: ###########CONTENT#####	
X071		XMLSerialize: ###########DOCUMENT#####	
X072		XMLSerialize: ########	
X090		XML####	
X120		SQL########XML#####	
X121		##########XML#####	
X221		####(BY VALUE)XML#####	
X301		XMLTable: ###########	
X302		XMLTable: ########	
X303		XMLTable: ###########	
X304		XMLTable: ###########	XML DOCUMENT## ########
X400		###########	
X410		#########: XML#	

D.2.

###SQL:2023#################PostgreSQL############ ########################

###	###	##	####
B011		####Ada	
B013		####COBOL	
B014		####Fortran	
B015		####MUMPS	
B016		####Pascal	
B017		####PL/I	
B030		#####SQL	
B031		####SQL	
B032		####SQL	
B033		####SQL###########	
B034		###########	
B035		#######	
B036		describe input#	

###	###	##	####
B041		####SQL########	
B051		######	
B111		####### Ada	
B112		####### C	
B113		####### COBOL	
B114		####### Fortran	
B115		####### MUMPS	
B116		####### Pascal	
B117		####### PL/I	
B121		###### Ada	
B122		###### C	
B123		###### COBOL	
B124		###### Fortran	
B125		###### MUMPS	
B126		###### Pascal	
B127		###### PL/I	
B200		#########(PTF)	
B201		2####PTF###############	
B202		PTF########	
B203		2#############	
B204		PRUNE WHEN EMPTY	
B205		######	
B206		PTF########	
B207		##########	
B208		PTF####################	
B209		PTF###	
B211		####### Ada: VARCHAR###NUMERIC#####	
B221		###### Ada: VARCHAR###NUMERIC#####	
F054		DATE##########TIMESTAMP	
F120		GET DIAGNOSTICS#	
F121		######	
F122		######	
F123		######	
F124		SET TRANSACTION#: DIAGNOSTICS SIZE#	
F263		##CASE#############	
F291		UNIQUE##	
F301		########CORRESPONDING	
F403		###########	
F451		#######	
F461		########	
F492		##############	

###	###	##	####
F521		##	
F671		CHECK#######	######
F673		CHECK####SQL################	
F693		SQL###################	
F695		######	
F696		###########	
F721		######	#############
F741		##MATCH#	########
F812		#######	
F813		#######	
F821		###########	
F831		#########	
F832		################	
F833		#############	
F841		LIKE_REGEX##	regexp_like()########
F842		OCCURRENCES_REGEX##	regexp_matches()######
F843		POSITION_REGEX##	regexp_instr()#######
F844		SUBSTRING_REGEX##	regexp_substr()#####
F845		TRANSLATE_REGEX##	regexp_replace()#####
F846		###########8#######	
F847		######	
F866		FETCH FIRST#: PERCENT#####	
R010		#######: FROM#	
R020		#######: WINDOW#	
R030		#######: ########	
S011	##	Distinct####	
S011-01	##	USER_DEFINED_TYPES###	
S023		######	
S024		######	
S025		######	
S026		########	
S027		#################	
S028		####UDT########	
S041		#####	
S043		#####	
S051		#########	######
S081		#####	

###	###	##	####
S091		########	######
S093		######	
S094		######	
S097		#######	
S151		###	pg_typeof()###
S161		#####	
S162		#########	
S202		########SQL#######	
S231		##########	
S232		########	
S233		##########	
S241		####	
S242		######	
S251		########	
S261		#######	
S271		##########	
S272		###########	
S274		########	
S275		##########	
S281		#######	
S291		###########	
S401		##########	
S402		##########	
S403		ARRAY_MAX_CARDINALITY	
T011		###############	
T021		BINARY###VARBINARY####	
T022		BINARY###VARBINARY###########	
T023		##2#####	
T024		2###########	
T039		CLOB#####: #####	
T040		CLOB###	
T041		##LOB########	
T042		##LOB########	
T043		##T	
T044		##P	
T045		BLOB####	
T046		CLOB####	
T047		BLOB#### POSITION#OCTET_LENGTH#TRIM#SUBSTRING	
T048		BLOB###	
T049		BLOB#####: #####	

###	###	##	####
T050		CLOB#### POSITION#CHAR_LENGTH#OCTET_LENGTH#LOWER#TRIM#UPPER#SUBSTRING	
T051		##	
T053		######################	
T062		#####	
T076		DECFLOAT####	
T101		##nullability###	
T111		#############	
T175		###	########
T176		#########	NEXT VALUE FOR# #######
T180		###############	
T181		Application-time period####	
T200		###DDL	##################
T211		#######	
T218		###############################	######
T231		Sensitive####	
T251		SET TRANSACTION#: LOCAL#####	
T262		#############	
T272		###########	
T301		#####	######
T321	##	##SQL########	######
T322		###########	
T324		SQL##################	
T326		######	
T471		#######	
T472		DESCRIBE CURSOR	
T495		################	#####
T502		Period##	
T511		#########	
T522		SQL###########IN###############	######DEFAULT## ###########
T561		##########	
T571		#####SQL######	
T572		#######SQL######	
T601		###########	
T616		LEAD###LAG####null#######	
T618		NTH_VALUE##	################### #######
T619		###########	
T625		LISTAGG	

###	###	##	####
T641		######	###########
T652		SQL########SQL#	
T654		##########SQL#	
T801		JSON####	
T802		##JSON####	
T824		JSON_TABLE: ###PLAN#	
T838		JSON_TABLE: PLAN DEFAULT#	
T839		#####/###datetime###########	
T860		SQL/JSON#######: #####	
T861		SQL/JSON#######: ###########JSON#######	
T862		SQL/JSON#######: ##############	
T863		SQL/JSON#######: ##########################	
T864		SQL/JSON#######	
T878		SQL/JSON######: timestamp##	
T881		######JSON	jsonb#########
T882		###############JSON	
M001		######	
M002		SQL/CLI#########	
M003		####SQL#########	
M004		#########	######
M005		##########	
M006		GetSQLString####	
M007		TransmitRequest	
M009		GetOpts###GetStatistics####	
M010		############	###API
M011		Ada#########	
M012		C#########	
M013		COBOL#########	
M014		Fortran#########	
M015		M#########	
M016		Pascal#########	
M017		PL/I#########	
M018		Ada####################	
M019		C####################	###API
M020		COBOL####################	
M021		Fortran####################	
M022		MUMPS####################	
M023		Pascal####################	
M024		PL/I####################	
M030		SQL############	

###	###	##	####
M031		##############	
X012		XML######	
X013		XML#####	
X015		XML#######	
X025		XMLCast	
X030		XMLDocument	
X065		XMLParse: ###########CONTENT#####	
X066		XMLParse: ###########DOCUMENT#####	
X068		XMLSerialize: BOM	
X073		XMLSerialize: BLOB#######CONTENT#####	
X074		XMLSerialize: BLOB#######DOCUMENT#####	
X075		XMLSerialize: BLOB#####	
X076		XMLSerialize: VERSION	
X077		XMLSerialize: ###ENCODING#####	
X078		XMLSerialize: ###XML##	
X080		XML##########	
X081		#########XML######	
X082		DML####XML######	
X083		DDL####XML######	
X084		#######XML######	
X085		############	
X086		XMLTable####XML######	
X091		XML#######	
X096		XMLExists	XPath 1.0##
X100		XML###########: CONTENT#####	
X101		XML###########: DOCUMENT#####	
X110		XML###########: VARCHAR####	
X111		XML###########: CLOB####	
X112		XML###########: BLOB####	
X113		XML###########: STRIP WHITESPACE#####	
X114		XML###########: PRESERVE WHITESPACE#### #	
X131		#########XMLBINARY#	
X132		DML####XMLBINARY#	
X133		DDL####XMLBINARY#	
X134		#######XMLBINARY#	
X135		##########XMLBINARY#	
X141		IS VALID##: ########	
X142		IS VALID##: ACCORDING TO#	
X143		IS VALID##: ELEMENT#	
X144		IS VALID##: ######	

###	###	##	####
X145		######IS VALID##	
X151		DOCUMENT#######IS VALID##	
X152		CONTENT#######IS VALID##	
X153		SEQUENCE#######IS VALID##	
X155		IS VALID##: ELEMENT####NAMESPACE	
X157		IS VALID##: ELEMENT####NO NAMESPACE	
X160		##XML################	
X161		##XML################	
X170		XML####null#########	
X171		NIL ON NO CONTENT#####	
X181		XML(DOCUMENT(UNTYPED))#	
X182		XML(DOCUMENT(ANY))#	
X190		XML(SEQUENCE)#	
X191		XML(DOCUMENT(XMLSCHEMA))#	
X192		XML(CONTENT(XMLSCHEMA))#	
X200		XMLQuery	
X201		XMLQuery: RETURNING CONTENT	
X202		XMLQuery: RETURNING SEQUENCE	
X203		XMLQuery: ###########	
X204		XMLQuery: XQuery######	
X205		XMLQuery: EMPTY ON EMPTY#####	
X206		XMLQuery: NULL ON EMPTY#####	
X211		XML 1.1####	
X222		#####(BY REF)XML#####	####BY REF#### ############BY VALUE######
X231		XML(CONTENT(UNTYPED))#	
X232		XML(CONTENT(ANY))#	
X241		XML######RETURNING CONTENT	
X242		XML######RETURNING SEQUENCE	
X251		XML(DOCUMENT(UNTYPED))######XML#	
X252		XML(DOCUMENT(ANY))######XML#	
X253		XML(CONTENT(UNTYPED))######XML#	
X254		XML(CONTENT(ANY))######XML#	
X255		XML(SEQUENCE)######XML#	
X256		XML(DOCUMENT(XMLSCHEMA))######XML#	
X257		XML(CONTENT(XMLSCHEMA))######XML#	
X260		XML#: ELEMENT#	
X261		XML#: ELEMENT####NAMESPACE	
X263		XML#: ELEMENT###NO NAMESPACE	
X264		XML#: ######	

###	###	##	####
X271		XMLValidate: ########	
X272		XMLValidate: ACCORDING TO#	
X273		XMLValidate: ELEMENT#	
X274		XMLValidate: ######	
X281		XMLValidate DOCUMENT#######	
X282		XMLValidate CONTENT#######	
X283		XMLValidate SEQUENCE#######	
X284		XMLValidate: ELEMENT####NAMESPACE	
X286		XMLValidate: ELEMENT###NO NAMESPACE	
X300		XMLTable	XPath 1.0##
X305		XMLTable: XQuery######	

D.3. XML####SQL/XML####

SQL:2006#ISO/IEC 9075-14 (SQL/XML)#XML####################### PostgreSQL#XML ####################################2003############# ##:

- #################XML################document###content######XML############ #XML#########`XML(SEQUENCE)`########PostgreSQL##document###content######### ##`xml`######## ####sequence##############
- PostgreSQL#SQL:2006##################################XML Query####XPath 1.0# ###########
- ######PostgreSQL#`RETURNING`#####`RETURNING`################ ################ #################

###########################

D.3.1. ######XPath 1.0######

PostgreSQL#####`xpath()`#`xpath_exists()`#XPath######XML########## PostgreSQL## ####XQuery#########`XMLEXISTS`#`XMLTABLE`######XPath############### ########## #PostgreSQL##XPath 1.0#######libxml2##############

XQuery###XPath#####2.0###XQuery##############XPath########## ########## XPath 1.0###################XPath 1.0#####################

#################### SQL##############XQuery###XPath################PostgreSQL# #####XPath######1.0##########

D.3.1.1. XQuery#XPath#####

XPath######XQuery###:

- XQuery#######XPath#############XML################## XPath################## #############XML#####################################
- XQuery############################
- XQuery######################

###XPath#####################for-each#sort###############parse-xml#####
##############XPath 3.0############

D.3.1.2. XPath#1.0##########

XQuery#XPath 2.0#############XPath 1.0#################

- XQuery/XPath########XML#######################sequence#XPath 1.0########
  1.0#########0####XML####################
- ####################XQuery/XPath#########XPath 1.0#########################
  ################

> ##
>
> libxml2#########################PostgreSQL################ ############
> ################XPath 1.0############

- XQuery/XPath#XML Schema####################################XPath 1.0######
  ##3#####boolean#double#string#######
- XPath 1.0############# if (hat) then hat/@size else "no hat"####XQuery/
  XPath####XPath 1.0#########
- XPath 1.0################## "cat" < "dog"#"cat" > "dog"#####2##NaN#####
  ############ ####=#!=################
- XPath 1.0###XQuery/XPath########################## sale/@hatsize =
  7#sale/@customer = "alice"################################sale#######
  ##sale/@taxable = false()####################### taxable#####sale#####
  ##########
- XQuery/XPath###
  ## #####XPath 1.0##################
  ## #####PostgreSQL#XPath##########################xml####################
  #####

#################### XQuery#2.0###XPath###XPath 1.0###################W3C#####
#######[1]#######[2]################################### ################XPath 1.0#
###############

D.3.1.3. SQL#XML###############

SQL:2006#######SQL#####XML########################## ###########XQuery/XPath
################XPath1.0#######################

PostgreSQL#SQL######XML#xmlelement#######XML##SQL#xmltable###########
##################PostgreSQL####XML#####XPath 1.0######SQL##################
################## ##

###9.8##########################
###

D.3.2.

###libxml2##################PostgreSQL######################

[1] https://www.w3.org/TR/2010/REC-xpath-functions-20101214/#xpath1-compatibility
[2] https://www.w3.org/TR/xpath20/#id-backwards-compatibility

D.3.2.1. #######BY VALUE####

SQL####SQL##XML###XML#######################2################ BY REF#### #XML#######ID#####BY VALUE##XML#########ID######## ##################### ##

##############*x*#XML#######SQL:2006#######2###########true#false#######

```
SELECT XMLQUERY('$a is $b' PASSING BY REF x AS a, x AS b NULL ON EMPTY);
SELECT XMLQUERY('$a is $b' PASSING BY VALUE x AS a, x AS b NULL ON EMPTY);
```

PostgreSQL##XMLEXISTS#XMLTABLE###BY VALUE#BY REF############## xml####### ######################ID##############BY VALUE###

D.3.2.2.

XPath#######XPath## ##############

D.3.2.3. XML(SEQUENCE)####

PostgreSQL#xml#####DOCUMENT###CONTENT############# XQuery/XPath############ #XML####################XPath 1.0#####XML################CONTENT########### ###### #########DOCUMENT####PostgreSQL#XPath################XML#######

##E

###########PostgreSQL########################## ############################## ## ################ ##

#######################Git################ `pgsql-committers`########[1]######## ################# #####################Web#######[2]#######

############################## ##

E.1. ####17

#####: 2024-??-??, AS OF 2024-06-17

E.1.1.

######PostgreSQL 17###########################

- #############

###########PostgreSQL 17###########################

E.1.2. #####17####

#######################################pg_dumpall##################pg_upgrade#### ################# ##############################18.6##########

########## 17 ## ####################

- ##################search_path################ (Jeff Davis)

  ####################(`ANALYZE`# `CLUSTER`#`REFRESH` `MATERIALIZED VIEW`#`REINDEX`####`VACUUM`)#################### ############################## ##

- ######`ago`#`interval`#################### (Joseph Koshakow)

  ##

- ##########old_snapshot_threshold####### (Thomas Munro)

  ##snapshot #################### #######################PostgreSQL################

- ##`SET REGION AUTHORIZATION`####### (Joseph Koshakow)

  ##`SET SESSION AUTHORIZATION`######## ##############################

- ################################### (Nathan Bossart)

  ########`db_user_namespace`################

[1] https://www.postgresql.org/list/pgsql-committers/
[2] https://git.postgresql.org/gitweb/?p=postgresql.git;a=summary

- # # # # # # Windows # wal_sync_method # `fsync_writethrough` # # # # # # # (Thomas Munro)

  # # # # # # # # # # Windows # `fsync` # # # # # #

- # # # # # # 2 # # WAL # (Kyotaro Horiguchi, Andres Freund, Bruce Momjian)

  # # # # # # # `pg_walfile_name()` # `pg_walfile_name_offset()` # # # # # LSN # # # # # # # # # # # # # # # # # # #

- ########IN######################## (Andy Fan, Tom Lane)
- ########################UNION ALL####LIMIT############# (Andy Fan, David Rowley)
- ## (Tom Lane)
- ######GROUP BY########ORDER BY################ (Andrei Lepikhov, Teodor Sigaev)

  ###############enable_group_by_reordering#############
- ######ALL#######UNION#MergeAppend############# (David Rowley)
- ######MergeAppend################################## (Alexander Kuzmenkov)
- ######GiST#######SP-GiST########################### (Miroslav Bendik)

  ###############GiST###SP-GiST#################ORDER BY#########
- ######range#################pg_stats######### (Egor Rogov, Soumyadeep Chakraborty)

E.1.3.1.2.

- #############IN####btree################################ (Peter Geoghegan, Matthias van de Meent)
- #################BRIN################### (Tomas Vondra, Matthias van de Meent)

E.1.3.1.3.

- ### (Melanie Plageman, Heikki Linnakangas)

  ######WAL####################################
- ################################### (Masahiko Sawada, John Naylor)

  #########maintenance_work_mem#autovacuum_work_mem####################1#######
  #########
- ############################ (Melanie Plageman)
- ##########vacuum_buffer_usage_limit#2############ (Thomas Munro)
- ### (Nathan Bossart)
- ##########WAL############# (Bharath Rupireddy)
- ## (Melih Mutlu)
- ##############################io_combine_limit########### (Thomas Munro, Andres Freund, Melanie Plageman, Nazir Bilal Yavuz)

E.1.3.1.4.

- #############pg_stat_checkpointer####### (Bharath Rupireddy, Anton A. Melnikov, Alexander Korotkov)

  ############pg_stat_bgwriter##########################
- ########################## (Atsushi Torikoshi, Bharath Rupireddy)

######pg_stat_reset_shared()#######pg_stat_reset_shared(NULL)################ ############ pg_stat_reset_shared('slru')#pg_stat_reset_slru()###########SLRU# ###################pg_stat_reset_slru(NULL)#########

- #######################WAL#########Andres Freund#######
- ###########log_connections########### (Jacob Champion)
- ##################walsender######################### (Bharath Rupireddy)

  ###############log_replication_commands###########

- ############################pg_wait_events######## (Bertrand Drouvot)

  ##########pg_stat_activity#########################

- ############################## (Thomas Munro)
- ################################### (Sami Imseih)

  ################pg_stat_progress_vacuum# #indexes_total#indexes_processed######

E.1.3.1.5.

- ############################## (Nathan Bossart)

  ###############MAINTAIN#####################pg_maintain################## ########### ########VACUUM#ANALYZE#REINDEX#REFRESH MATERIALIZED VIEW#CLUSTER####LOCK TABLE###

- ######pg_monitor######to##pg_current_logfile()########## (Pavlo Golub, Nathan Bossart)

E.1.3.1.6.

- ##################allow_alter_system######ALTER SYSTEM# (Jelte Fennema-Nio, Gabriele Bartolini)
- ######ALTER SYSTEM################################ (Tom Lane)

  #########GRANT ON PARAMETER#######

- ###########transaction_timeout################# (Andrey Borodin, Japin Li, Junwang Zhao, Alexander Korotkov)
- ## (Jeff Davis)

  ##########C###C.UTF-8###########

- ###########huge_page_size##########POSTGRES#huge pages############ (Justin Pryzby)

  ######huge_pages#try###############

- ##################################### (Daniel Gustafsson)

  ######event_triggers################################

- ######SLRU###################### (Andrey Borodin, Dilip Kumar, Alvaro Herrera)

  ############## ##commit_timestamp_buffers#multixact_member_buffers#multixact_offset_buffers#notify_buffers#serializable_bu

commit_timestamp_buffers#transaction_buffers#subtransaction_buffers#####shared_buffers#

E.1.3.1.7. Streaming Replication and Recovery

- ################################## (Robert Haas, Jakub Wartak, Tomas Vondra)

  ################pg_basebackup####`--incremental`#################### ######## ###pg_combinebackup####################################

- ######WAL################ (Robert Haas, Nathan Bossart, Hubert Depesz Lubaczewski)

  ################LSN## ######### #summarize_wal#wal_summary_keep_time####### ##`pg_available_wal_summaries()`#`pg_wal_summary_contents()`### #`pg_get_wal_summarizer_state()`##########

- ## (Amul Sul)

  ############WAL##############

- ######pg_basebackup##########`PostgreSQL.auto.conf`#############`dbname`###### ######## (Vignesh C, Hayato Kuroda)

- #########`pg_`########`_slots.invocation_reason`######################### (Shveta Malik, Bharath Rupireddy)

- #########`pg_`########`_slots.inactive_since`######################## (Bharath Rupireddy)

- ########`pg_sync_replication_slots()`########################### (Hou Zhijie, Shveta Malik, Ajin Cherian, Peter Eisentraut)

- ######`failover`######################### (Hou Zhijie, Shveta Malik)

E.1.3.1.8. Logical Replication

- ##pg_createsubscriber####### (Euler Taveira)

- ######pg_upgrade############################## (Hayato Kuroda, Hou Zhijie, Vignesh C, Julien Rouhaud, Shlok Kyal)

  ## ################17############### ### PostgreSQL

- ############################## (Hou Zhijie, Shveta Malik, Ajin Cherian)

  ################5#####to`pg_create_logical_replication_slot()`######## ##

- ###########sync_replication_slots################################### (Shveta Malik, Hou Zhijie, Peter Smith)

- ################################`CREATE/ALTER SUBSCRIPTION`####### (Shveta Malik, Hou Zhijie, Ajin Cherian)

- ##hash######################### (Hayato Kuroda)

  ############btree#####################

- ################################### (Masahiko Sawada)

- ##(Vignesh C)#

  ######################

- ######flush######pg_logical_emit_message()######## (Michael Paquier)

  ######################

- ## ############ #Hou Zhijie#Shveta Malik#

  ###############synchronized_standby_slots###

- ######pg_stat_subscription################## (Peter Smith)

E.1.3.2.

- ###################COPYON_ERROR ignore######## (Damir Belyalov, Atsushi Torikoshi, Alex Shulgin, Jian He, Yugo Nagata)

  ###############ON_ERROR stop###

- ######COPY FROM################COPY#####LOG_VERBOSITY####### (Bharath Rupireddy)

- ######COPY FROM############################ (Atsushi Torikoshi)

  ####################pg_stat_progress_###.tuples_#############

- ######COPY FROM##############null#####null###################### (Zhang Mingli)

- ###########################EXPLAIN####### (Ashutosh Bapat)

  ##############MEMORY#######

- ######EXPLAIN#####SERIALIZE############ ######################## (Stepan Rutz, Matthias van de Meent)

- ##########I/O#########/##############EXPLAIN#BUFFERS##(Nazir Bilal Yavuz)#######

- ######EXPLAIN########################### (Tom Lane, Dean Rasheed)

- ######JITdeform_counter####EXPLAIN####### (Dmitry Dolgov)

- ######ALTER TABLE ... MERGE PARTITIONS########################### (Dmitry Koval)

- ######ALTER TABLE ... SPLIT PARTITION######################### (Dmitry Koval)

- ##################ID######## (Ashutosh Bapat)

- ############################ (Paul A. Jungwirth)

  ##

- ################################### (Justin Pryzby, Soumyadeep Chakraborty, Michael Paquier)

- ######ALTER TABLE## (Peter Eisentraut)

  ############ALTER TABLE ... SET STATISTICS DEFAULT### SET STATISTICS -1##################

- ######ALTER TABLE################ (Amul Sul)

  ########ALTER TABLE ... ALTER COLUMN ... SET EXPRESSION###

- ######ALTER TABLE.. SET ACCESS METHOD#DEFAULT######## (Michael Paquier)
- ###################################### (Konstantin Knizhnik, Mikhail Gribkov)
- ######REINDEX################## (Garrett Thornburg, Jian He)
- ######################CLUSTER#################### (Nathan Bossart)

E.1.3.3.

- ######################+/-infinity######## (Joseph Koshakow, Jian He, Ashutosh Bapat)
- ###########################ALTER TYPE########ENUM########## (Tom Lane)

  ########################

E.1.3.4. MERGE

- ######MERGE################### (Dean Rasheed)
- ######WHEN NOT MATCHED BY SOURCE#MERGE####### (Dean Rasheed)

  ##############WHEN NOT MATCHED###############

- ######MERGE#RETURNING############# (Dean Rasheed)

  #########RETURNING##merge_action()#######DML#########

E.1.3.5.

- ###########################JSONJSON_TABLE()###### (Nikita Glukhov, Teodor Sigaev, Oleg Bartunov, Alexander Korotkov, Andrew Dunstan, Amit Langote, Jian He)

  ###########SELECT####FROM################

- ######SQL/JSON########JSON()# JSON_SCALAR()#JSON_SERIALIZE()####### (Nikita Glukhov, Teodor Sigaev, Oleg Bartunov, Alexander Korotkov, Andrew Dunstan, Amit Langote)
- ######SQL/JSON#####JSON_EXISTS()#JSON_QUERY()####JSON_VALUE()###### (Nikita Glukhov, Teodor Sigaev, Oleg Bartunov, Alexander Korotkov, Andrew Dunstan, Amit Langote, Peter Eisentraut, Jian He)
- ######JSON####JSON###########jsonpath###########

  ######jsonpath#######.bigint()#.boolean()#.date()#.decimal([precision [,
  scale]])#.integer()#.number()#.string()#.time()#.time_tz()#.timestamp()#.timest
  ##

- ######Addto_timestamp()############### (Tom Lane)

  ######TZ##########################OF################

- ##################AS LOCAL############# (Vik Fearing)

  ##

- ########uuid_extract_timestamp()#uuid_extract_version()###UUID###### ### (Andrey Borodin)
- ##############(Dean Rasheed)###################

  #########random(min, max)#####integer#bigint#numeric########
- ##################################### (Eric Radman, Nathan Bossart)

  #########to_bin()#to_oct()###
- ######Unicode########### (Jeff Davis)

  ########unicode_version()#Unicode#########icu_unicode_version()#ICU## #######unicode_assigned()########Unicode################
- ############XML############################xmltext()#######
- ########################to_regtypemod()######## (David Wheeler, Erik Wienhold)
- ######pg_basetype()#####a############### (Steve Chavez)
- ########pg_column_toast_chunk_id()###a##TOAST########## (Yugo Nagata)

  ########TOAST#############NULL######

E.1.3.6. PL/pgSQL

- ######plpgsql%TYPE###%ROWTYPE########################### (Quan Zongliang, Pavel Stehule)
- ######plpgsql%TYPE########################### (Tom Lane)

E.1.3.7. libpq

- ######libpq######################### (Joe Conway)

  ###########PQchangePassword()###########################
- ##################libpq###### (Jelte Fennema-Nio)

  ######## ##PQclosePrepared()#PQclosePortal()#PQsendClosePrepared()#PQsendClosePortal()# ##
- ####################################libpqAPI############################### (Jelte Fennema-Nio)

  ##
- ######libpq##PQsocketPoll()#################Tristan Partin#Tom Lane############ ###
- ######libpq##PQsendPipelineSync()######################### (Anton Kirilov)

  ######PQpipelineSync()##
- ######libpq##PQsetChunkedRowsMode()########################## (Daniel Vérité)
- ############################TLS######### (Greg Stark, Heikki Linnakangas, Peter Eisentraut, Michael Paquier, Daniel Gustafsson)

#####################sslnegotiation=direct######ALPN######PostgreSQL17# #############

E.1.3.8. psql

- #######psql################# (Erik Wienhold, Laurenz Albe)

  #########\dp#########(none)############## #############

- ######################\pset null# (Erik Wienhold, Laurenz Albe)

  ########\pset null##########

- ######psql#\watch######################### (Greg Sabino Mullane)

  ###########min_rows###

- ######psql################C##########################

- ######psql##FETCH_COUNT##SELECT################# (Daniel Vérité)

- #######psql###################å####Gilles Darold#Christoph Heiss#Steve Chavez#Vignesh C#Pavel Borisov#Jian He##

E.1.3.9.

- #############WAL#########pg_walsummary######## (Robert Haas)

- ######pg_############################ (Tom Lane)

  ##

- ######Addpg_###option--exclude-extension# (Ayush Vatsa)

- ######pg_####pg_dumpall####pg_##############Pavel Stehule#Daniel Gustafsson### include/###############

  #############--filter######

- #####--sync-################################## (Justin Pryzby, Nathan Bossart)

  ###################initdb#pg_basebackup#pg_checksums#pg_dump#pg_rewind### ##pg_upgrade###

- ######pg_########--########-########## ################(Tom Lane)######### ###########

  ##

- ######pgbench############-d##-########## (Greg Sabino Mullane)

  ##########-d###################### ######--dbname############

- #######################pgbench#####--exit-on-abort######## (Yugo Nagata)

- ######################pgbench####\syncpipeline######## #############

- ######pg_archivecleanup########################## (Atsushi Torikoshi)

  ###########--clean-backup-history###

- ######pg_archivecleanup############### (Atsushi Torikoshi)

#############--debug#--dry-run#--strip-extension#

- ######pg_basebackup#pg_receivewal###########dbname############## (Jelte Fennema-Nio)

  ##

- ######pg_upgrade#####--###-####-########## (Thomas Munro)

  #########Linux#FreeBSD###########

- ######reindexdb--## (Maxim Orlov, Svetlana Derevyanko, Alexander Korotkov)

- ######reindexdb#vacuumdb####clusterdb#################################### ##### ######### (Nathan Bossart)

  ##############--all############

E.1.3.10.

- ######OpenSSL1.0.1########## (Michael Paquier)

- ######Allow tests to pass inOpenSSLFIPS################# (Peter Eisentraut)

- ################CPU AVX-512######### (Paul Amonson, Nathan Bossart, Ants Aasma)

- ######RequireLLVM#####10## (Thomas Munro)

- ######64-#####LoongArchCPU####CRC######## (Xudong Yang)

- ######AIX########## (Heikki Linnakangas)

- ######Microsoft Visual Studio###PostgreSQL############### (Michael Paquier)

  ######Meson#Visual Studio####################

- ######configure#####--disable-thread-safety######## #Thomas Munro#Heikki Linnakangas#

  ##

- ########configure#####--with-CC# (Heikki Linnakangas)

  ######CC##

- ######################receive####null###################### (David Rowley)

- #########JSON################JSON########## (Andrew Dunstan)

- #########-###README#####Markdown####### (Nathan Bossart)

- ###############-###INSTALL########## (Tom Lane)

- ######make#distprep########### (Peter Eisentraut)

- ####################make########### (Peter Eisentraut)

- ################################### (Michael Paquier)

  ##

- ######emit_log_hook################################### (Kambam Vinay, Michael Paquier)
- ################jsonpath####### (David Wheeler)

E.1.3.11.

- ## (Richard Guo, Etsuro Fujita)

  ##
- ######postgres_fdw#####(Alexander Pyhalov)##EXISTS###IN###################
- ############################## (David Rowley, Umair Shahid)

  ########################
- ######dblink################## (Noah Misch)
- ######ltree########################### (Tommy Pavlicek)

  ############ltree#########################
- ######unaccent######################### (Michael Paquier)

  ######unaccent.rules###############
- ##############--checkunique######################amcheck######Anastasia Lubennikova#Pavel Borisov#Maxim Orlov)#
- ######OpenSSLFIPS####citext############## (Peter Eisentraut)
- ######OpenSSLFIPS####pgcrypto############## (Peter Eisentraut)
- ##########SPI############### (Bharath Rupireddy)
- ######adminpackcontrib extension####### (Daniel Gustafsson)

  ####################pgAdmin III########
- ######ALTER OPERATOR######################## (Tommy Pavlicek)

  #################
- ############################## (Masahiro Ikeda)

  ######postgres_fdw###dblink###################
- ######pg_buffercache##pg_buffercache_evict()######################### (Palak Chaturvedi, Thomas Munro)

  ##################

E.1.3.11.1. pg_stat_statements

- ######pg_stat_statements##CALL#################### (Sami Imseih)
- ######pg_stat_statements############################## (Greg Sabino Mullane)

  ###############SAVEPOINT#RELEASE SAVEPOINT####ROLLBACK TO SAVEPOINT# #######################
- ######pg_stat_statements############GID############### (Michael Paquier)

###############PREPARE TRANSACTION#COMMIT PREPARED####ROLLBACK PREPARED################

- ######DEALLOCATE#pg_stat_statements####### (Dagfinn Ilmari Mannsåker, Michael Paquier)

  ######DEALLOCATE###########pg_stat_statements#######

- ######pg_stat_statements###############/#################### (Nazir Bilal Yavuz)

  #################_blk_read_time#####_blk_write_time###

- ######JITdeform_########pg_stat_statements####### (Dmitry Dolgov)

- #############4#####(minmax_only)#pg_stat_statements_reset()##### min/max #################### (Andrei Zubkov)

  ################false######

- ######pg_stat_statements#stats_since#minmax_stats_since############# ####min/############### (Andrei Zubkov)

E.1.4.

###

E.2.

####################https://www.postgresql.org/docs/release/######

##F

##########PostgreSQL####contrib#################################### ### ##PostgreSQL### ######################### ################################### ####################

#######contrib################################## ##G#######################

#############"world"##(#### 2##)# ######## ################################

make
make install

###########contrib########### ########1################################### ################## #########################

make check

################

make installcheck

PostgreSQL##################

PostgreSQL##############################postgresql-contrib################## ###############

#######1######################## ##########SQL########################### ###CREATE EXTENSION############## ####

```
CREATE EXTENSION extension_name;
```

#######################SQL## ########## ################################template1####################

##################trusted############CREATE EXTENSION######################## ########## #######################CREATE#################### ############### ############### ##

################################

btree_gin	fuzzystrmatch	ltree	tcn
btree_gist	hstore	pgcrypto	tsm_system_rows
citext	intarray	pg_trgm	tsm_system_time
cube	isn	seg	unaccent
dict_int	lo	tablefunc	uuid-ossp

########CREATE EXTENSION#### #SCHEMA *schema_name*######### ## #public###

#####################################shared_preload_libraries#################### ######## ###########################

F.1. amcheck —

amcheck##################################

#####[80.790960]#B-Tree################################## ################### ## ##################B-Tree############ #######################text#B-Tree##################################### ### ##SQL####################### # ########################## ######B## ## ## ############ ########B-##B-################### ########################## ## ##################SQL################ ########################## ############ ##search_path#####pg_####,pg_temp#######

################################## ############################## #####B-Tree## #############B-Tree######1################## ###################36.16.3########

###################B-Tree################verify_heapam#################1### ############### #######verify_heapam## #############

#########amcheck## ###### ##

F.1.1.

bt_index_check(index regclass, heapallindexed boolean, checkunique boolean) returns void

bt_index_check######B-Tree############################## #######

```
test=# SELECT bt_index_check(index => c.oid, heapallindexed =>
 i.indisunique),
               c.relname,
               c.relpages
FROM pg_index i
JOIN pg_opclass op ON i.indclass[0] = op.oid
JOIN pg_am am ON op.opcmethod = am.oid
JOIN pg_class c ON i.indexrelid = c.oid
JOIN pg_namespace n ON c.relnamespace = n.oid
WHERE am.amname = 'btree' AND n.nspname = 'pg_catalog'
-- Don't check temp tables, which may be from another session:
AND c.relpersistence != 't'
-- Function may throw an error when this is omitted:
AND c.relkind = 'i' AND i.indisready AND i.indisvalid
ORDER BY c.relpages DESC LIMIT 10;
 bt_index_check |             relname             | relpages
----------------+---------------------------------+----------
                | pg_depend_reference_index       |       43
                | pg_depend_depender_index        |       40
                | pg_proc_proname_args_nsp_index  |       31
                | pg_description_o_c_o_index      |       21
                | pg_attribute_relid_attnam_index |       14
                | pg_proc_oid_index               |       10
                | pg_attribute_relid_attnum_index |        9
                | pg_amproc_fam_proc_index        |        5
                | pg_amop_opr_fam_index           |        5
```

```
                    | pg_amop_fam_strat_index         |        5
(10 rows)
```

#############test#########10############################## ################
############################
################   ##
#bt_index_check##############

#####[73.312565]#bt_index_check##################################
#AccessShareLock######   ############SELECT######################
bt_index_check####################*heapallindexed*#true##############
################################   ##########################
##bt_index_check###
########   ######bt_######_##############################
#AccessShareLock###### ##
bt_######_##############################*heapallindexed*##############
############################## *checkunique*##bt_######_###############
################1############### ##############################bt_#####
#_###-############
####

bt_index_parent_check(index regclass, heapallindexed boolean, rootdescend boolean, checkunique boolean) returns void

#####[77.753465]#bt_index_parent_check#########B-Tree###################
######## ########*heapallindexed*###true################################
######*rootdescend*#true###
######## bt_index_parent_check###########bt_index_check############
############ bt_index_parent_check##bt_index_check##################
####bt_index_check####bt_index_parent_check######################
######################## bt_index_parent_check#######################
################## ######bt_######_#_###############B-#################
############## ######*heapallindexed*################################
####### *checkunique*##bt_######_#_##############################1####
############ ######*rootdescend*#######################################
#################### bt_######_#_###############bt_######_#########
############## bt_######_#_######bt_######_######################:bt_
######_#########bt_######_#_#######################################/#
#################### bt_######_#_#######################################
###########

bt_index_parent_check#############ShareLock##############
#ShareLock################## #######INSERT#UPDATE#DELETE#############
########VACUUM##
##

bt_index_parent_check############################## ################
####################B-Tree######################B-Tree####################
######################### bt_index_check#####bt_index_parent_check##
##

bt_index_check#bt_index_parent_check######DEBUG1#DEBUG2#########
#################### ########PostgreSQL##############################
####### ## #########

SET client_min_messages = DEBUG1;
```
```

```
#################psql#######################################
```

verify_heapam(relation regclass, on_error_stop boolean, check_toast boolean, skip text, startblock bigint, endblock bigint, blkno OUT bigint, offnum OUT integer, attnum OUT integer, msg OUT text) returns setof record

```
#########################################################################
#################################

###################

on_error_stop

    ################################

    #########

check_toast

    ####TOAST##############TOAST###############

    #################### ###TOAST###########################TOAST########
    ############################ ##########################

    #########

skip

    none#######################################################   ########
    #all-visible#all-frozen#none###

    ######none###

startblock

    ######################################## startblock##################
    ##################

    ########################

endblock

    ######################################## endblock####################
    ################

    ######################

###########verify_heapam##############

blkno

    ##############

offnum

    #############

attnum

    ##########################################
```

```
msg
```

#################

F.1.2. #####heapallindexed##

B-Tree#####heapallindexed###true###
########### #########CREATE INDEX###
###########################
##########(fingerprint)####### heapallindexed#################################
###

###heapallindexed###################### ###################### ###########
##############heapallindexed##################

########maintenance_work_mem################ ############################
###2%######################2############# ##################################
####### ##
############################

F.1.3. amcheck#######

amcheck## ##############

- ################################

  ##################################### text############################B-Tree###
  ########## ## #################
  ################################## ##
  #### ##

  ## ####################
  ############################ PostgreSQL##################################
  #########23.1#23.2########

- ######################################heapallindexed###########

  ## ########################

- ####PostgreSQL###

  ########### PostgreSQL###
  ####### ## ####################
  #############amcheck############### ##############31.1########

- ##

  amcheck### ###
  ##amcheck################################## #############amcheck########
  ######################################

- #####RAM#########################

  PostgreSQL#####################################(ECC)##################RAM#
  ################ ####ECC###
  ###############

  heapallindexed############1####################### ####################
  ######################################

##
##################################

###########
################ ###########TOAST########TOAST############################
#####ID##########################ID##############

PostgreSQL###
###############

###
##verify_heapam############################## ##############################
####################################
####

#####amcheck######################################

F.1.4.

amcheck############################ amcheck##############################
#####amcheck############################

amcheck######################## ######################## amcheck###########
##pageinspect################ REINDEX######################

F.2. auth_delay —

auth_delay## ###DoS########
##DoS##################

#############postgresql.conf# shared_preload_libraries####################

F.2.1.

auth_delay.milliseconds(integer)

   ##############################0###

#########postgresql.conf##################################

```
# postgresql.conf
shared_preload_libraries = 'auth_delay'

auth_delay.milliseconds = '500'
```

F.2.2.

KaiGai Kohei <kaigai@ak.jp.nec.com>

F.3. auto_explain —

auto_explain#########EXPLAIN#################################### ######## ################################

########SQL################## ################### ########################

```
LOAD 'auto_explain';
```

########################## ######### ##postgresql.conf#session_preload_libraries###shared_preload_libraries#auto_explain# ################################## ###################################### ### ########################

F.3.1.

auto_explain######################### ################################### ##auto_explain.log_min_duration######################

auto_explain.log_min_duration(integer)

auto_explain.log_min_duration#################################### # ##0################## -1################### ####250ms######250ms######## ######### ##########################

auto_explain.log_parameter_max_length(integer)

auto_explain.log_parameter_max_length######################## -1#### ######################## 0################## ############################### ### ########################

auto_explain.log_analyze(boolean)

auto_explain.log_analyze#############################EXPLAIN##### ##EXPLAIN ANALYZE######## ################## ####################

> ##
>
> ## ## ################### auto_explain.log_timing#################### ###################

auto_explain.log_buffers(boolean)

auto_explain.log_buffers## EXPLAIN#BUFFERS########## auto_explain.log_analyze################## ############## ################## ##########################

auto_explain.log_wal(boolean)

auto_explain.log_wal################WAL################### EXPLAIN#WAL########## auto_explain.log_analyze#################### ########## ################## ##########################

auto_explain.log_timing(boolean)

auto_explain.log_timing## EXPLAIN#TIMING########## ##

    ##
    auto_explain.log_analyze########################### ########### #######
    ###################

auto_explain.log_triggers (boolean)

    auto_explain.log_triggers#######################################
    auto_explain.log_analyze################################ #############
    ###### #########################

auto_explain.log_verbose (boolean)

    auto_explain.log_verbose#######################################
    EXPLAIN#VERBOSE########### ################## ####################

auto_explain.log_settings (boolean)

    auto_explain.log_settings##
    ## ## ################## ######
    ###############

auto_explain.log_format (enum)

    auto_explain.log_format#####EXPLAIN######### ######
    #text#xml#json#yaml### ######text### #########################

auto_explain.log_level (enum)

    auto_explain.log_level##auto_explain####################### ####
    #DEBUG5#DEBUG4#DEBUG3#DEBUG2#DEBUG1#INFO#NOTICE#WARNING#LOG### ####
    ##LOG### ####################

auto_explain.log_nested_statements (boolean)

    auto_explain.log_nested_statements##################################
    #### ######################### ################## #########################
    ##

auto_explain.sample_rate (real)

    auto_explain.sample_rate####auto_explain################################
    ## ######1######################### #######################################
    ###### ####################

##postgresql.conf############# #
##################

```
# postgresql.conf
session_preload_libraries = 'auto_explain'

auto_explain.log_min_duration = '3s'
```

F.3.2.

```
postgres=# LOAD 'auto_explain';
postgres=# SET auto_explain.log_min_duration = 0;
postgres=# SET auto_explain.log_analyze = true;
postgres=# SELECT count(*)
           FROM pg_class, pg_index
           WHERE oid = indrelid AND indisunique;
```

```
#######################


LOG:  duration: 3.651 ms  plan:
  Query Text: SELECT count(*)
              FROM pg_class, pg_index
              WHERE oid = indrelid AND indisunique;
  Aggregate  (cost=16.79..16.80 rows=1 width=0) (actual time=3.626..3.627
 rows=1 loops=1)
    ->  Hash Join  (cost=4.17..16.55 rows=92 width=0) (actual
 time=3.349..3.594 rows=92 loops=1)
          Hash Cond: (pg_class.oid = pg_index.indrelid)
          ->  Seq Scan on pg_class  (cost=0.00..9.55 rows=255 width=4)
 (actual time=0.016..0.140 rows=255 loops=1)
          ->  Hash  (cost=3.02..3.02 rows=92 width=4) (actual
 time=3.238..3.238 rows=92 loops=1)
                Buckets: 1024  Batches: 1  Memory Usage: 4kB
                ->  Seq Scan on pg_index  (cost=0.00..3.02 rows=92
 width=4) (actual time=0.008..3.187 rows=92 loops=1)
                      Filter: indisunique
```

F.3.3.

<itagaki.takahiro.at.oss.ntt.co.jp>

F.4. basebackup_to_shell — pg_basebackup #####"shell"##

basebackup_to_shell##shell#########basebackup########### ####
##pg_basebackup --target=shell#####################pg_basebackup --target=shell:*DETAIL_STRING*#######################################tar###### ############## ############################

###
###############shared_preload_libraries###local_preload_libraries#################

F.4.1.

basebackup_to_shell.command(string)

> ## #######%f################## ###########:base.tar## #######%d################################### #### ####%d######################## ################ ########################## ######## %%#####################%########## %############################## ##################

basebackup_to_shell.required_role(string)

> shell########################## ##########################shell######## ###########

F.4.2.

Robert Haas <rhaas@postgresql.org>

F.5. basic_archive — WAL############

basic_archive############### ############WAL########################### # ## ###############49###########

###############archive_library#######archive_mode##############

F.5.1.

basic_archive.archive_directory(string)

> ####WAL####################### ####################### ###############
> WAL########### ###archive_mode##################################WAL#####
> ###########

#########postgresql.conf############### ###################

```
# postgresql.conf
archive_mode = 'on'
archive_library = 'basic_archive'
basic_archive.archive_directory = '/path/to/archive/directory'
```

F.5.2.

############################archtemp################### ###################
####################### ##
#########

F.5.3.

Nathan Bossart

F.6. bloom —

bloom##########[1]######################

######################
###

######################
########### ### ############
##

####btree#######
##btree##############################
##
##

F.6.1.

bloom#######WITH##################

length

   ############################## 16############## ######80########4096###

col1 – col32

   ###################### ############################## ######2########4095#
   ## ##################################

F.6.2.

################

```
CREATE INDEX bloomidx ON tbloom USING bloom (i1,i2,i3)
       WITH (length=80, col1=2, col2=2, col3=4);
```

#########80##################i1#i2#2#####i3#4########### length#col1#col2####
#########################

############################ #############btree############## ###########btree####
##################################

```
=# CREATE TABLE tbloom AS
   SELECT
     (random() * 1000000)::int as i1,
     (random() * 1000000)::int as i2,
     (random() * 1000000)::int as i3,
     (random() * 1000000)::int as i4,
     (random() * 1000000)::int as i5,
     (random() * 1000000)::int as i6
   FROM
  generate_series(1,10000000);
SELECT 10000000
```

#####################################

[1] https://en.wikipedia.org/wiki/Bloom_filter

```
=# EXPLAIN ANALYZE SELECT * FROM tbloom WHERE i2 = 898732 AND i5 = 123451;
                                                 QUERY PLAN
-------------------------------------------------------------------
-----------------------------------
 Seq Scan on tbloom  (cost=0.00..2137.14 rows=3 width=24) (actual
 time=16.971..16.971 rows=0 loops=1)
   Filter: ((i2 = 898732) AND (i5 = 123451))
   Rows Removed by Filter: 100000
 Planning Time: 0.346 ms
 Execution Time: 16.988 ms
(5 rows)
```

###btree#####################################

```
=# CREATE INDEX btreeidx ON tbloom (i1, i2, i3, i4, i5, i6);
CREATE INDEX
=# SELECT pg_size_pretty(pg_relation_size('btreeidx'));
 pg_size_pretty
----------------
 3976 kB
(1 row)
=# EXPLAIN ANALYZE SELECT * FROM tbloom WHERE i2 = 898732 AND i5 = 123451;
                                                 QUERY PLAN
-------------------------------------------------------------------
-----------------------------------
 Seq Scan on tbloom  (cost=0.00..2137.00 rows=2 width=24) (actual
 time=12.805..12.805 rows=0 loops=1)
   Filter: ((i2 = 898732) AND (i5 = 123451))
   Rows Removed by Filter: 100000
 Planning Time: 0.138 ms
 Execution Time: 12.817 ms
(5 rows)
```

##########################btree##################

```
=# CREATE INDEX bloomidx ON tbloom USING bloom (i1, i2, i3, i4, i5, i6);
CREATE INDEX
=# SELECT pg_size_pretty(pg_relation_size('bloomidx'));
 pg_size_pretty
----------------
 1584 kB
(1 row)
=# EXPLAIN ANALYZE SELECT * FROM tbloom WHERE i2 = 898732 AND i5 = 123451;
                                                       QUERY PLAN
-------------------------------------------------------------------
-----------------------------------------------------
 Bitmap Heap Scan on tbloom  (cost=1792.00..1799.69 rows=2 width=24)
 (actual time=0.388..0.388 rows=0 loops=1)
   Recheck Cond: ((i2 = 898732) AND (i5 = 123451))
   Rows Removed by Index Recheck: 29
   Heap Blocks: exact=28
   ->  Bitmap Index Scan on bloomidx  (cost=0.00..1792.00 rows=2 width=0)
 (actual time=0.356..0.356 rows=29 loops=1)
         Index Cond: ((i2 = 898732) AND (i5 = 123451))
 Planning Time: 0.099 ms
 Execution Time: 0.408 ms
(8 rows)
```

btree## btree#########
######################## ##################

```
=# CREATE INDEX btreeidx1 ON tbloom (i1);
```

```
CREATE INDEX
=# CREATE INDEX btreeidx2 ON tbloom (i2);
CREATE INDEX
=# CREATE INDEX btreeidx3 ON tbloom (i3);
CREATE INDEX
=# CREATE INDEX btreeidx4 ON tbloom (i4);
CREATE INDEX
=# CREATE INDEX btreeidx5 ON tbloom (i5);
CREATE INDEX
=# CREATE INDEX btreeidx6 ON tbloom (i6);
CREATE INDEX
=# EXPLAIN ANALYZE SELECT * FROM tbloom WHERE i2 = 898732 AND i5 = 123451;
                                                           QUERY PLAN
-----------------------------------------------------------------------
-------------------------------------------------------
 Bitmap Heap Scan on tbloom  (cost=24.34..32.03 rows=2 width=24) (actual
 time=0.028..0.029 rows=0 loops=1)
   Recheck Cond: ((i5 = 123451) AND (i2 = 898732))
   ->  BitmapAnd  (cost=24.34..24.34 rows=2 width=0) (actual
 time=0.027..0.027 rows=0 loops=1)
         ->  Bitmap Index Scan on btreeidx5  (cost=0.00..12.04 rows=500
 width=0) (actual time=0.026..0.026 rows=0 loops=1)
               Index Cond: (i5 = 123451)
         ->  Bitmap Index Scan on btreeidx2  (cost=0.00..12.04 rows=500
 width=0) (never executed)
               Index Cond: (i2 = 898732)
 Planning Time: 0.491 ms
 Execution Time: 0.055 ms
(9 rows)
```

#########
btree#######2MB#####################12MB######################8######

F.6.3.

######text#####
#################

```
CREATE OPERATOR CLASS text_ops
DEFAULT FOR TYPE text USING bloom AS
    OPERATOR    1   =(text, text),
    FUNCTION    1   hashtext(text);
```

F.6.4.

- #########int4#text###################
- =################### ##############################
- bloom########UNIQUE###########
- bloom########NULL###############

F.6.5.

Teodor Sigaev <teodor@postgrespro.ru>, Postgres Professional, Moscow, Russia

Alexander Korotkov <a.korotkov@postgrespro.ru>, Postgres Professional, Moscow, Russia

Oleg Bartunov <obartunov@postgrespro.ru>, Postgres Professional, Moscow, Russia

F.7. btree_gin — GIN#######B-tree##

btree_gin##############B-tree##########GIN################
##int2#int4#int8#float4#float8#timestamp with time zone#timestamp without time zone#time with time zone#time without time zone#date#interval#oid#money#"char"#varchar#text#bytea#bit#varbit#macaddr#macaddr8
#####enum####

###################B-tree#######################B-tree#######################
######GIN##########GIN################### ####GIN############B-tree##
################################AND#######################################
##########GIN####################

#########trusted#########################CREATE#######################

F.7.1.

```
CREATE TABLE test (a int4);

-- #########
CREATE INDEX testidx ON test USING GIN (a);

-- #####
SELECT * FROM test WHERE a < 10;
```

F.7.2.

Teodor Sigaev#<teodor@stack.net>#### Oleg Bartunov#<oleg@sai.msu.su>## ####
#http://www.sai.msu.su/~megera/oddmuse/index.cgi/Gin########

F.8. btree_gist — GiST#######B-tree##

btree_gist##############B-tree##########GiST#####################
##int2#int4#int8#float4#float8#numeric#timestamp with time zone#timestamp without time zone#time with time zone#time without time zone#date#interval#oid#money#char#varchar#text#bytea#bit#varbit#macaddr#macaddr8#inet#cidr#uuid#bool#######enum####

###################B-tree#######################B-tree#######################
###############B-tree####################### ##############GiST########
################################GiST################# ####GiST##########GiST#
##################

####B-tree#########btree_gist#<>############################## #############
#################

#################btree_gist######<->###################GiST##############
#####int2#int4#int8#float4#float8#timestamp with time zone#timestamp without time zone#time without time zone#date#interval#oid#money#######

#########trusted#######################CREATE#######################

F.8.1.

btree#####btree_gist#######

```
CREATE TABLE test (a int4);

-- #########
CREATE INDEX testidx ON test USING GIST (a);

-- #####
SELECT * FROM test WHERE a < 10;

-- #####: "42"#####10###########
SELECT *, a <-> 42 AS dist FROM test ORDER BY a <-> 42 LIMIT 10;

#########1################################
```

```
=> CREATE TABLE zoo (
  cage   INTEGER,
  animal TEXT,
  EXCLUDE USING GIST (cage WITH =, animal WITH <>)
);

=> INSERT INTO zoo VALUES(123, 'zebra');
INSERT 0 1
=> INSERT INTO zoo VALUES(123, 'zebra');
INSERT 0 1
=> INSERT INTO zoo VALUES(123, 'lion');
ERROR:  conflicting key value violates exclusion constraint
 "zoo_cage_animal_excl"
DETAIL:  Key (cage, animal)=(123, lion) conflicts with existing key (cage,
 animal)=(123, zebra).
=> INSERT INTO zoo VALUES(124, 'lion');
INSERT 0 1
```

F.8.2.

Teodor Sigaev#<teodor@stack.net>## Oleg Bartunov#<oleg@sai.msu.su>## Janko Richter (<jankorichter@yahoo.de>)####Paul Jungwirth (<pj@illuminatedcomputing.com>)# #####http://www.sai.msu.su/~megera/postgres/gist/# #######

F.9. citext —

citext################citext######### ################lower######## ####
##text#########

###
####################(23.2.2.4##)############### ########################### ##

#########trusted#########################CREATE#######################

F.9.1.

PostgreSQL##lower################

```
SELECT * FROM tab WHERE lower(col) = LOWER(?);
```

#########################

- ####SQL######## ################lower########################
- lower####################################
- UNIQUE###PRIMARY KEY### ########### ##

citext######SQL######lower############## ##########################
citext#text############## #################LC_CTYPE################# ######
####lower############### ##
#########

F.9.2.

#########

```
CREATE TABLE users (
    nick CITEXT PRIMARY KEY,
    pass TEXT   NOT NULL
);

INSERT INTO users VALUES ( 'larry',  sha256(random()::text::bytea) );
INSERT INTO users VALUES ( 'Tom',    sha256(random()::text::bytea) );
INSERT INTO users VALUES ( 'Damian', sha256(random()::text::bytea) );
INSERT INTO users VALUES ( 'NEAL',   sha256(random()::text::bytea) );
INSERT INTO users VALUES ( 'Bjørn',  sha256(random()::text::bytea) );

SELECT * FROM users WHERE nick = 'Larry';
```

SELECT###nick##larry###########Larry####################

F.9.3.

citext##########lower########################### #######lower###############
######################

####################################citext#################################### ##########
#####~###~*##citext################ ########################### !~#!~*#####
#LIKE####~~#~~*#!~~#!~~*########### ############################text#######
########

###citext####################################

- regexp_match()
- regexp_matches()
- regexp_replace()
- regexp_split_to_array()
- regexp_split_to_table()
- replace()
- split_part()
- strpos()
- translate()

#######RegExp#########################c#################################### ##
##text###############

F.9.4.

- citext##########################LC_CTYPE####### ########################### ######## Unicode#################################### ########################### ###########citext################## ##################################### ####################################
- PostgreSQL 9.1###COLLATE###citext##################### #####citext######## ##########################COLLATE######## ##################LC_CTYPE##### ################COLLATE "default"############# ################COLLATE# ############################
- ########B-tree##citext#text######## #### ###text###B-Tree############# ###citext#######################lower### ##############
- citext## #### ###text############################lower########### ##################### ################ ##citext## ############################text################## ########2############## ###########################
- citext###############search_path(#####public)# ########################### #######text##############
- ##########################1#########2################################ ##### ################################## ############citext################### ##

F.9.5.

David E. Wheeler <david@kineticode.com>

Donald Fraser###citext#############

F.10. cube —

##################cube########## ######cube##############cube##############
###

#########trusted#########################CREATE#######################

F.10.1.

F.1#cube############## *x#y*###############

#F.1 cube#####

####	##
x	1###################
(*x*)	##
x1,*x2*,...,*xn*	n####### #######0############
(*x1*,*x2*,...,*xn*)	##
(*x*),(*y*)	*x##y*##########1###### ##########
[(*x*),(*y*)]	##
(*x1*,...,*xn*),(*y1*,...,*yn*)	############n#######
[(*x1*,...,*xn*),(*y1*,...,*yn*)]	##

################## ####### — ######################cube############## ######
###cube##################################

############### #####[(*x*),(*y*)]#[(*x*), (*y*)]######

F.10.2.

######64################### ########16#######################

F.10.3.

F.2##cube###################

#F.2 cube####

##
cube && cube → boolean   #########
cube @> cube → boolean   1#######2##########
cube <@ cube → boolean   1#######2###########
cube -> integer → float8   #####1######*n*##########
cube ~> integer → float8   ##########*n*########## *n* = 2 * *k* - 1 #*k*#######*n* = 2 * *k* #*k*########### ##*n*####   ################### ######KNN-GiST##################

###	##
cube <-> cube → float8	2###################
cube <#> cube → float8	2#########(L-1##)########
cube <=> cube → float8	2###########(L-#####)########

##########cube#### 9.1################## ##############################2###### ######### #######cube#####B-tree###########################cube####UNIQUE### ###########

cube#######cube#####GiST################## cube#GiST#######WHERE### #=#&&#@>#<@##########################

####cube#GiST#######ORDER BY####<->#<#>#<=>############################### #### ####3#######(0.5, 0.5, 0.5)#######################

```
SELECT c FROM test ORDER BY c <-> cube(array[0.5,0.5,0.5]) LIMIT 1;
```

~>### ####1################ ##

```
SELECT c FROM test ORDER BY c ~> 1 LIMIT 5;
```

#######1##################2#####################################

```
SELECT c FROM test ORDER BY c ~> 3 DESC LIMIT 5;
```

F.3###########

#F.3 cube###

##	## #
cube (float8) → cube	########1############ cube(1) → (1)
cube (float8, float8) → cube	1############ cube(1, 2) → (1),(2)
cube (float8[]) → cube	################0########### cube(ARRAY[1,2,3]) → (1, 2, 3)
cube (float8[], float8[]) → cube	2## cube(ARRAY[1,2], ARRAY[3,4]) → (1, 2),(3, 4)
cube (cube, float8) → cube	## ############################## ####

`cube('(1,2),(3,4)'::cube, 5) → (1, 2, 5),(3, 4, 5)`

`cube` (`cube`, `float8`, `float8`) → `cube`
######################### ################################ 	
`cube('(1,2),(3,4)'::cube, 5, 6) → (1, 2, 5),(3, 4, 6)`

`cube_dim` (`cube`) → `integer`
############
`cube_dim('(1,2),(3,4)') → 2`

`cube_ll_coord` (`cube`, `integer`) → `float8`
#######*n*##########
`cube_ll_coord('(1,2),(3,4)', 2) → 2`

`cube_ur_coord` (`cube`, `integer`) → `float8`
#######*n*##########
`cube_ur_coord('(1,2),(3,4)', 2) → 4`

`cube_is_point` (`cube`) → `boolean`
################2###############
`cube_is_point(cube(1,1)) → t`

`cube_distance` (`cube`, `cube`) → `float8`
2############## #########################
`cube_distance('(1,2)', '(3,4)') → 2.8284271247461903`

`cube_subset` (`cube`, `integer[]`) → `cube`
## ########################## 	
#######################
`cube_subset(cube('(1,3,5),(6,7,8)'), ARRAY[2]) → (3),(7)`
`cube_subset(cube('(1,3,5),(6,7,8)'), ARRAY[3,2,1,1]) → (5, 3, 1, 1),(8, 7, 6, 6)`

`cube_union` (`cube`, `cube`) → `cube`
2###############
`cube_union('(1,2)', '(3,4)') → (1, 2),(3, 4)`

`cube_inter` (`cube`, `cube`) → `cube`
2################
`cube_inter('(1,2)', '(3,4)') → (3, 4),(1, 2)`

`cube_enlarge` (*c* `cube`, *r* `double`, *n* `integer`) → `cube`
###*n*##########*r*############## ################## ###########*r*#####
### #####*r*##########*r*####### ####################(###*r* < 0######### 	
#)################## *n*#######################(*r* > 0)###*n*##############
##########################0######## ################################
##
`cube_enlarge('(1,2),(3,4)', 0.5, 3) → (0.5, 1.5, -0.5),(3.5, 4.5, 0.5)`

F.10.4.

```
select cube_union('(0,5,2),(2,3,1)', '0');
cube_union
-------------------
(0, 0, 0),(2, 5, 2)
(1 row)
```

##########################

```
select cube_inter('(0,-1),(1,1)', '(-2),(2)');
cube_inter
-------------
(0, 0),(1, 0)
(1 row)
```

###0############### ########
####

```
cube_union('(0,5,2),(2,3,1)','(0,0,0),(0,0,0)');
cube_inter('(0,-1),(1,1)','(-2,0),(2,0)');
```

##########################2############# ###############(box,point)############
###########

```
select cube_contains('(0,0),(1,1)', '0.5,0.5');
cube_contains
--------------
t
(1 row)
```

F.10.5.

#########sql/cube.sql##################

################100############# ###cubedata.h############################

F.10.6.

###: Gene Selkov, Jr. <selkovjr@mcs.anl.gov>, Mathematics and Computer Science Division, Argonne National Laboratory.

GiST (http://gist.cs.berkeley.edu/)###(gist)#########Joe Hellerstein## (https://dsf.berkeley.edu/jmh/) ####### ###Illustra################Andy Dong####### ########################
########################Postgres######### ######################Argonne Lab#U.S. Department of Energy########

2002#8#/9##Bruno Wolff III <bruno@wolff.to>######################## ###########
##########################

2006#7##Joshua Reich <josh@root.net>############### ######cube(float8[], float8[])#########V0#######V1##########################

F.11. dblink — ##PostgreSQL###########

dblink#################PostgreSQL#######################

######dblink##################extension#########

DblinkConnect

    ##############################

DblinkGetConnect

    ##

DblinkGetResult

    ################################

postgres_fdw##

dblink_connect

dblink_connect — ######################

##

```
dblink_connect(text connstr) returns text
dblink_connect(text connname, text connstr) returns text
```

##

dblink_connect()######PostgreSQL################ ##################libpq###### ######### ########################## ################################1####### ### ####################################

############################ ######################dblink_fdw############## ## ###CREATE SERVER#CREATE USER MAPPING#########

##

connname

##

connstr

###hostaddr=127.0.0.1 port=5432 dbname=mydb user=postgres password=mypasswd####libpq############ ###32.1.1######### ############

###

######## #####OK##################################

##

###search_path######## ################## ######*connstr*##-csearch_path=############### ####### ##dblink############# ###SQL###########################

#################GSSAPI##############dblink_connect####### ############## #####dblink_connect_u#############

##dblink##

# #

```
SELECT dblink_connect('dbname=postgres options=-csearch_path=');
 dblink_connect
----------------
 OK
(1 row)

SELECT dblink_connect('myconn', 'dbname=postgres options=-csearch_path=');
 dblink_connect
----------------
 OK
```

```
(1 row)


-- ##########FOREIGN DATA WRAPPER####
-- ####################################
--       ######dblink_connect():################
--
 ----------------------------------------------------------------------
--       ERROR:  password is required
--       DETAIL:  Non-superuser cannot connect if the server does not
 request a password.
--       HINT:  Target server's authentication method must be changed.

CREATE SERVER fdtest FOREIGN DATA WRAPPER dblink_fdw OPTIONS (hostaddr
 '127.0.0.1', dbname 'contrib_regression');

CREATE USER regress_dblink_user WITH PASSWORD 'secret';
CREATE USER MAPPING FOR regress_dblink_user SERVER fdtest OPTIONS (user
 'regress_dblink_user', password 'secret');
GRANT USAGE ON FOREIGN SERVER fdtest TO regress_dblink_user;
GRANT SELECT ON TABLE foo TO regress_dblink_user;

\set ORIGINAL_USER :USER
\c - regress_dblink_user
SELECT dblink_connect('myconn', 'fdtest');
 dblink_connect
----------------
 OK
(1 row)

SELECT * FROM dblink('myconn', 'SELECT * FROM foo') AS t(a int, b text, c
 text[]);
 a  | b |       c
----+---+---------------
  0 | a | {a0,b0,c0}
  1 | b | {a1,b1,c1}
  2 | c | {a2,b2,c2}
  3 | d | {a3,b3,c3}
  4 | e | {a4,b4,c4}
  5 | f | {a5,b5,c5}
  6 | g | {a6,b6,c6}
  7 | h | {a7,b7,c7}
  8 | i | {a8,b8,c8}
  9 | j | {a9,b9,c9}
 10 | k | {a10,b10,c10}
(11 rows)

\c - :ORIGINAL_USER
REVOKE USAGE ON FOREIGN SERVER fdtest FROM regress_dblink_user;
REVOKE SELECT ON TABLE foo FROM regress_dblink_user;
DROP USER MAPPING FOR regress_dblink_user SERVER fdtest;
DROP USER regress_dblink_user;
DROP SERVER fdtest;
```

dblink_connect_u

dblink_connect_u — ##########################

##

```
dblink_connect_u(text connstr) returns text
dblink_connect_u(text connname, text connstr) returns text
```

##

```
dblink_connect_u()#####################################dblink_connect()#
#####

##########################################PostgreSQL#########################
#######################  ####################################~/.pgpass######
############################## ##############################################
#### ####dblink_connect_u()############################PUBLIC#############
############# ########################dblink_connect_u()#EXECUTE########
############ ##################### #############~/.pgpass##################
############################

#######dblink_connect()#########
```

dblink_disconnect

dblink_disconnect — ######################

##

```
dblink_disconnect() returns text
dblink_disconnect(text connname) returns text
```

##

dblink_disconnect()#dblink_connect()################ ################## ##

##

connname

##############

###

######## #####OK##################################

#

```
SELECT dblink_disconnect();
 dblink_disconnect
-------------------
 OK
(1 row)

SELECT dblink_disconnect('myconn');
 dblink_disconnect
-------------------
 OK
(1 row)
```

dblink

dblink — ####################

##

```
dblink(text connname, text sql [, bool fail_on_error]) returns setof
 record
dblink(text connstr, text sql [, bool fail_on_error]) returns setof record
dblink(text sql [, bool fail_on_error]) returns setof record
```

##

dblink####################SELECT#########SQL####################

2##text###dblink_connect###

##

connname

   ########### #########################

connstr

   ##dblink_connect##############

sql

   ###select * from foo####################SQL#######

fail_on_error

   ## ####################NOTICE####################

###

####################### dblink###record############### ## #####PostgreSQL################# #########

```
SELECT *
    FROM dblink('dbname=mydb options=-csearch_path=',
                'select proname, prosrc from pg_proc')
      AS t1(proname name, prosrc text)
    WHERE proname LIKE 'bytea%';
```

FROM################################ ###########SQL################PostgreSQL####### ####################*############WHERE###proname############## ##########################FROM######################## ################### #####dblink############## ###########FROM#########################

##

################dblink######################### ### #########

```
CREATE VIEW myremote_pg_proc AS
  SELECT *
    FROM dblink('dbname=postgres options=-csearch_path=',
                'select proname, prosrc from pg_proc')
    AS t1(proname name, prosrc text);

SELECT * FROM myremote_pg_proc WHERE proname LIKE 'bytea%';
```

#

```
SELECT * FROM dblink('dbname=postgres options=-csearch_path=',
                     'select proname, prosrc from pg_proc')
  AS t1(proname name, prosrc text) WHERE proname LIKE 'bytea%';
  proname   |   prosrc
------------+------------
 byteacat   | byteacat
 byteaeq    | byteaeq
 bytealt    | bytealt
 byteale    | byteale
 byteagt    | byteagt
 byteage    | byteage
 byteane    | byteane
 byteacmp   | byteacmp
 bytealike  | bytealike
 byteanlike | byteanlike
 byteain    | byteain
 byteaout   | byteaout
(12 rows)

SELECT dblink_connect('dbname=postgres options=-csearch_path=');
 dblink_connect
----------------
 OK
(1 row)

SELECT * FROM dblink('select proname, prosrc from pg_proc')
  AS t1(proname name, prosrc text) WHERE proname LIKE 'bytea%';
  proname   |   prosrc
------------+------------
 byteacat   | byteacat
 byteaeq    | byteaeq
 bytealt    | bytealt
 byteale    | byteale
 byteagt    | byteagt
 byteage    | byteage
 byteane    | byteane
 byteacmp   | byteacmp
 bytealike  | bytealike
 byteanlike | byteanlike
 byteain    | byteain
 byteaout   | byteaout
(12 rows)

SELECT dblink_connect('myconn', 'dbname=regression options=-
csearch_path=');
 dblink_connect
----------------
 OK
(1 row)

SELECT * FROM dblink('myconn', 'select proname, prosrc from pg_proc')
  AS t1(proname name, prosrc text) WHERE proname LIKE 'bytea%';
```

```
  proname   |   prosrc
------------+------------
 bytearecv  | bytearecv
 byteasend  | byteasend
 byteale    | byteale
 byteagt    | byteagt
 byteage    | byteage
 byteane    | byteane
 byteacmp   | byteacmp
 bytealike  | bytealike
 byteanlike | byteanlike
 byteacat   | byteacat
 byteaeq    | byteaeq
 bytealt    | bytealt
 byteain    | byteain
 byteaout   | byteaout
(14 rows)
```

dblink_exec

dblink_exec — ####################

##

```
dblink_exec(text connname, text sql [, bool fail_on_error]) returns text
dblink_exec(text connstr, text sql [, bool fail_on_error]) returns text
dblink_exec(text sql [, bool fail_on_error]) returns text
```

##

dblink_exec############################SQL########

2##text## ######################### ### ############dblink_connect##

##

connname

############ ##########################

connstr

##dblink_connect##############

sql

###insert into foo values(0, 'a', '{"a0","b0","c0"}')############# #######SQL########

fail_on_error

####################NOTICE################ERROR######

###

################ERROR######

#

```
SELECT dblink_connect('dbname=dblink_test_standby');
 dblink_connect
----------------
 OK
(1 row)

SELECT dblink_exec('insert into foo values(21, ''z'',
 ''{"a0","b0","c0"}'');');
   dblink_exec
-----------------
 INSERT 943366 1
(1 row)

SELECT dblink_connect('myconn', 'dbname=regression');
```

```
 dblink_connect
----------------
 OK
(1 row)

SELECT dblink_exec('myconn', 'insert into foo values(21, ''z'',
 ''{"a0","b0","c0"}'');');
   dblink_exec
------------------
 INSERT 6432584 1
(1 row)

SELECT dblink_exec('myconn', 'insert into pg_class values
 (''foo'')',false);
NOTICE:  sql error
DETAIL:  ERROR:  null value in column "relnamespace" violates not-null
 constraint

 dblink_exec
-------------
 ERROR
(1 row)
```

dblink_open

dblink_open — ##################

##

```
dblink_open(text cursorname, text sql [, bool fail_on_error]) returns text
dblink_open(text connname, text cursorname, text sql [, bool
 fail_on_error]) returns text
```

##

```
dblink_open()####################                                ######
#dblink_fetch()#dblink_close()############
```

##

connname

   ########### ##########################

cursorname

   ################

sql

   ###select * from pg_class####################SELECT####

fail_on_error

   ##   ####################
   NOTICE################ERROR######

###

```
######OK###ERROR######
```

##

```
#######################################################dblink_open##########
###############BEGIN#####     ##############dblink_close#################
dblink_open#dblink_close###dblink_exec################################
## ###dblink_close###dblink_disconnect##################################
#######
```

# #

```
SELECT dblink_connect('dbname=postgres options=-csearch_path=');
 dblink_connect
----------------
 OK
(1 row)

SELECT dblink_open('foo', 'select proname, prosrc from pg_proc');
 dblink_open
```

```
-------------
 OK
(1 row)
```

dblink_fetch

dblink_fetch — ##############################

##

```
dblink_fetch(text cursorname, int howmany [, bool fail_on_error]) returns
 setof record
dblink_fetch(text connname, text cursorname, int howmany [, bool
 fail_on_error]) returns setof record
```

##

dblink_fetch#dblink_open##########################

##

connname

############ ##########################

cursorname

################

howmany

############ #############*howmany*######################## ##################
###########

fail_on_error

####################
NOTICE####################

###

######################## ##############dblink############################

##

##################FROM######################## ################################
############ ######FETCH################################

#

```
SELECT dblink_connect('dbname=postgres options=-csearch_path=');
 dblink_connect
----------------
 OK
(1 row)

SELECT dblink_open('foo', 'select proname, prosrc from pg_proc where
 proname like ''bytea%''');
 dblink_open
-------------
```

```
 OK
(1 row)

SELECT * FROM dblink_fetch('foo', 5) AS (funcname name, source text);
 funcname |  source
----------+----------
 byteacat | byteacat
 byteacmp | byteacmp
 byteaeq  | byteaeq
 byteage  | byteage
 byteagt  | byteagt
(5 rows)

SELECT * FROM dblink_fetch('foo', 5) AS (funcname name, source text);
 funcname  |  source
-----------+-----------
 byteain   | byteain
 byteale   | byteale
 bytealike | bytealike
 bytealt   | bytealt
 byteane   | byteane
(5 rows)

SELECT * FROM dblink_fetch('foo', 5) AS (funcname name, source text);
  funcname  |   source
------------+------------
 byteanlike | byteanlike
 byteaout   | byteaout
(2 rows)

SELECT * FROM dblink_fetch('foo', 5) AS (funcname name, source text);
 funcname | source
----------+--------
(0 rows)
```

dblink_close

dblink_close — ####################

##

```
dblink_close(text cursorname [, bool fail_on_error]) returns text
dblink_close(text connname, text cursorname [, bool fail_on_error])
 returns text
```

##

dblink_close#####dblink_open###############

##

connname

############ ##########################

cursorname

#############

fail_on_error

####################
NOTICE################ERROR######

###

######OK###ERROR######

##

dblink_open##dblink_close##
###COMMIT#######

#

```
SELECT dblink_connect('dbname=postgres options=-csearch_path=');
 dblink_connect
----------------
 OK
(1 row)

SELECT dblink_open('foo', 'select proname, prosrc from pg_proc');
 dblink_open
-------------
 OK
(1 row)

SELECT dblink_close('foo');
 dblink_close
--------------
 OK
```

```
(1 row)
```

dblink_get_connections

dblink_get_connections — ########dblink#############

##

```
dblink_get_connections() returns text[]
```

##

dblink_get_connections###########dblink####################

###

####################NULL###

#

```
SELECT dblink_get_connections();
```

dblink_error_message

dblink_error_message — ########################

##

```
dblink_error_message(text connname) returns text
```

##

dblink_error_message##################################

##

connname

##########

###

################ ###################OK######

##

```
#########dblink_send_query##################################################
######             ##########dblink_error_message#####dblink_is_busy###
#dblink_get_result################## ######################################
###
```

# #

```
SELECT dblink_error_message('dtest1');
```

dblink_send_query

dblink_send_query — ########################

##

```
dblink_send_query(text connname, text sql) returns int
```

##

dblink_send_query#################################### ################## ########

#################dblink_is_busy#################### ##### ##dblink_get_result############# ###dblink_cancel_query################ ###########

##

connname

##########

sql

###select * from pg_class####################SQL####

###

##############1###### #######0######

#

```
SELECT dblink_send_query('dtest1', 'SELECT * FROM foo WHERE f1 < 3');
```

dblink_is_busy

dblink_is_busy — ######################

##

```
dblink_is_busy(text connname) returns int
```

##

dblink_is_busy#####################

##

connname

##########

###

##############1######0###### #####0#######dblink_get_result################
##

#

```
SELECT dblink_is_busy('dtest1');
```

dblink_get_notify

dblink_get_notify — ################

##

```
dblink_get_notify() returns setof (notify_name text, be_pid int, extra
 text)
dblink_get_notify(text connname) returns setof (notify_name text, be_pid
 int, extra text)
```

##

dblink_get_notify## dblink#### #########dblink_exec#####LISTEN################ ###LISTEN#NOTIFY########

##

connname

#################

###

setof (notify_name text, be_pid int, extra text)###################

# #

```
SELECT dblink_exec('LISTEN virtual');
 dblink_exec
-------------
 LISTEN
(1 row)

SELECT * FROM dblink_get_notify();
 notify_name | be_pid | extra
-------------+--------+-------
(0 rows)

NOTIFY virtual;
NOTIFY

SELECT * FROM dblink_get_notify();
 notify_name | be_pid | extra
-------------+--------+-------
 virtual     |   1229 |
(1 row)
```

dblink_get_result

dblink_get_result — ################

##

```
dblink_get_result(text connname [, bool fail_on_error]) returns setof
 record
```

##

dblink_get_result#####dblink_send_query###################### ##########
##########dblink_get_result###########

##

connname

##########

fail_on_error

####################
NOTICE####################

###

#############SQL################################ ################dblink#######
####################

##############SQL##################################1###1####### ##########
#FROM##############################

##

dblink_send_query#1######################## ##############################
################ ########################

dblink_send_query#dblink_get_result#######dblink#######################
################ ## ##########
#dblink_open#################################### ########dblink()######## #
################################

# #

```
contrib_regression=# SELECT dblink_connect('dtest1',
 'dbname=contrib_regression');
 dblink_connect
----------------
 OK
(1 row)

contrib_regression=# SELECT * FROM
contrib_regression-# dblink_send_query('dtest1', 'select * from foo where
 f1 < 3') AS t1;
 t1
```

```
----
  1
(1 row)

contrib_regression=# SELECT * FROM dblink_get_result('dtest1') AS t1(f1
 int, f2 text, f3 text[]);
 f1 | f2 |     f3
----+----+------------
  0 | a  | {a0,b0,c0}
  1 | b  | {a1,b1,c1}
  2 | c  | {a2,b2,c2}
(3 rows)

contrib_regression=# SELECT * FROM dblink_get_result('dtest1') AS t1(f1
 int, f2 text, f3 text[]);
 f1 | f2 | f3
----+----+----
(0 rows)

contrib_regression=# SELECT * FROM
contrib_regression-# dblink_send_query('dtest1', 'select * from foo where
 f1 < 3; select * from foo where f1 > 6') AS t1;
 t1
----
  1
(1 row)

contrib_regression=# SELECT * FROM dblink_get_result('dtest1') AS t1(f1
 int, f2 text, f3 text[]);
 f1 | f2 |     f3
----+----+------------
  0 | a  | {a0,b0,c0}
  1 | b  | {a1,b1,c1}
  2 | c  | {a2,b2,c2}
(3 rows)

contrib_regression=# SELECT * FROM dblink_get_result('dtest1') AS t1(f1
 int, f2 text, f3 text[]);
 f1 | f2 |      f3
----+----+---------------
  7 | h  | {a7,b7,c7}
  8 | i  | {a8,b8,c8}
  9 | j  | {a9,b9,c9}
 10 | k  | {a10,b10,c10}
(4 rows)

contrib_regression=# SELECT * FROM dblink_get_result('dtest1') AS t1(f1
 int, f2 text, f3 text[]);
 f1 | f2 | f3
----+----+----
(0 rows)
```

dblink_cancel_query

dblink_cancel_query — ##########################

##

```
dblink_cancel_query(text connname) returns text
```

##

```
dblink_cancel_query##############################  ######################
##################################  ##################################  ##
##dblink_get_result##############################
```

##

connname

#########

###

```
##############OK##############################
```

#

```
SELECT dblink_cancel_query('dtest1');
```

dblink_get_pkey

dblink_get_pkey — ##############################

##

```
dblink_get_pkey(text relname) returns setof dblink_pkey_results
```

##

dblink_get_pkey################################### ########################
################

##

relname

###foo#myschema.mytab##################### ###"FooBar"###############
############################## #####################

###

#####1####### ########################### ###################

```
CREATE TYPE dblink_pkey_results AS (position int, colname text);
```

position####1##*N*###### ######################################

# #

```
CREATE TABLE foobar (
    f1 int,
    f2 int,
    f3 int,
    PRIMARY KEY (f1, f2, f3)
);
CREATE TABLE

SELECT * FROM dblink_get_pkey('foobar');
 position | colname
----------+---------
        1 | f1
        2 | f2
        3 | f3
(3 rows)
```

dblink_build_sql_insert

dblink_build_sql_insert — ####################################INSERT#######

##

```
dblink_build_sql_insert(text relname,
                        int2vector primary_key_attnums,
                        integer num_primary_key_atts,
                        text[] src_pk_att_vals_array,
                        text[] tgt_pk_att_vals_array) returns text
```

##

dblink_build_sql_insert###
###INSERT SQL########### ###
################2##################

##

relname

###foo#myschema.mytab#################### ###"FooBar"################
############################ ####################

primary_key_attnums

###1 2##################1#######

num_primary_key_atts

#############

src_pk_att_vals_array

################################ ################### ######################
##############

tgt_pk_att_vals_array

####INSERT########################### ###################

###

####SQL#############

##

PostgreSQL 9.0#####*primary_key_attnums*#######SELECT * FROM relname######
#################### ############################ ############################
############

#

```
SELECT dblink_build_sql_insert('foo', '1 2', 2, '{"1", "a"}', '{"1",
 "b''a"}');
```

```
             dblink_build_sql_insert
-----------------------------------------------
 INSERT INTO foo(f1,f2,f3) VALUES('1','b''a','1')
(1 row)
```

dblink_build_sql_delete

dblink_build_sql_delete — ######################DELETE#######

##

```
dblink_build_sql_delete(text relname,
                        int2vector primary_key_attnums,
                        integer num_primary_key_atts,
                        text[] tgt_pk_att_vals_array) returns text
```

##

dblink_build_sql_delete## ####################DELETE SQL##########

##

relname

###foo#myschema.mytab#################### ###"FooBar"################ ############################ #####################

primary_key_attnums

###1 2##################1#######

num_primary_key_atts

#############

tgt_pk_att_vals_array

####DELETE######################### ###################

###

####SQL#############

##

PostgreSQL 9.0#####*primary_key_attnums*#######SELECT * FROM relname###### #################### ########################### ############################ ############

#

```
SELECT dblink_build_sql_delete('"MyFoo"', '1 2', 2, '{"1", "b"}');
           dblink_build_sql_delete
----------------------------------------------
 DELETE FROM "MyFoo" WHERE f1='1' AND f2='b'
(1 row)
```

dblink_build_sql_update

dblink_build_sql_update — ######################UPDATE#######

##

```
dblink_build_sql_update(text relname,
                        int2vector primary_key_attnums,
                        integer num_primary_key_atts,
                        text[] src_pk_att_vals_array,
                        text[] tgt_pk_att_vals_array) returns text
```

##

dblink_build_sql_update###
###UPDATE SQL########## ##
#################2################# ##UPDATE####################### ####
#dblink_build_sql_insert#######################################

##

relname

###foo#myschema.mytab#################### ###"FooBar"################
############################## #####################

primary_key_attnums

###1 2##################1#######

num_primary_key_atts

#############

src_pk_att_vals_array

################################ ################### #######################
##############

tgt_pk_att_vals_array

####UPDATE########################### ###################

###

####SQL#############

##

PostgreSQL 9.0#####*primary_key_attnums*#######SELECT * FROM relname######
#################### ############################ ############################
############

#

```
SELECT dblink_build_sql_update('foo', '1 2', 2, '{"1", "a"}', '{"1",
 "b"}');
                   dblink_build_sql_update
-------------------------------------------------------------
 UPDATE foo SET f1='1',f2='b',f3='1' WHERE f1='1' AND f2='b'
(1 row)
```

F.12. dict_int —

dict_int###################### ##
##

#########trusted#########################CREATE#######################

F.12.1.

#####3##############

- maxlen########################### #######6###
- rejectlong################################ rejectlong#false############
  #######maxlen####### rejectlong#true############################## ######
  ########### ################################
- absval##########+#####-########################## ######false### true##
  ##maxlen#################

F.12.2.

dict_int##############intdict_template##########################
#intdict######### #######################

```
mydb# ALTER TEXT SEARCH DICTIONARY intdict (MAXLEN = 4, REJECTLONG =
 true);
ALTER TEXT SEARCH DICTIONARY
```

##############################

#####################

```
mydb# select ts_lexize('intdict', '12345678');
 ts_lexize
-----------
 {123456}
```

##################12################################ ###########

```
ALTER TEXT SEARCH CONFIGURATION english
    ALTER MAPPING FOR int, uint WITH intdict;
```

F.13. dict_xsyn —

dict_xsyn############################ ################################### ###############

F.13.1.

dict_xsyn#################

- matchorig######################### ######true###
- matchsynonyms######################## ######false###
- keeporig######################### ######true###
- keepsynonyms######################## ######true###
- rules###################### ######$SHAREDIR/tsearch_data/#$SHAREDIR# PostgreSQL## #####.rules############ ####rules##############

rules############

- ##############1#################### ####################

  word syn1 syn2 syn3

- #################### ################### ##############

####$SHAREDIR/tsearch_data/#########xsyn_sample.rules#########

F.13.2.

dict_xsyn##############xsyn_template################################### #xsyn######### ##########################

```
mydb# ALTER TEXT SEARCH DICTIONARY xsyn (RULES='my_rules',
 KEEPORIG=false);
ALTER TEXT SEARCH DICTIONARY
```

```
################################
```

```
######################
```

```
mydb=# SELECT ts_lexize('xsyn', 'word');
      ts_lexize
-----------------------
 {syn1,syn2,syn3}

mydb# ALTER TEXT SEARCH DICTIONARY xsyn (RULES='my_rules', KEEPORIG=true);
ALTER TEXT SEARCH DICTIONARY

mydb=# SELECT ts_lexize('xsyn', 'word');
      ts_lexize
-----------------------
 {word,syn1,syn2,syn3}

mydb# ALTER TEXT SEARCH DICTIONARY xsyn (RULES='my_rules', KEEPORIG=false,
 MATCHSYNONYMS=true);
```

```
ALTER TEXT SEARCH DICTIONARY

mydb=# SELECT ts_lexize('xsyn', 'syn1');
      ts_lexize
------------------------
 {syn1,syn2,syn3}

mydb# ALTER TEXT SEARCH DICTIONARY xsyn (RULES='my_rules', KEEPORIG=true,
 MATCHORIG=false, KEEPSYNONYMS=false);
ALTER TEXT SEARCH DICTIONARY

mydb=# SELECT ts_lexize('xsyn', 'syn1');
      ts_lexize
------------------------
 {word}

##############12################################ ###########


ALTER TEXT SEARCH CONFIGURATION english
    ALTER MAPPING FOR word, asciiword WITH xsyn, english_stem;
```

F.14. earthdistance —

earthdistance###############2############# #########cube########### 2#####
######################point#############

######################### #############PostGIS[1]##################

cube######earthdistance#########################(####################
#CREATE EXTENSION#CASCADE###############)#

##
earthdistance#cube##############################CREATE########### ######################### ######earthdistance###################### ################################## ###########earthdistance######### ##############################

F.14.1. cube#########

#######x#y#z#######3###################cube############## cube###earth###### ##### ##

######earth()########## ########### #####1################################## ######################################

######################### ####################earth()#180/pi()########### #######

##############################2##
##

########## F.4########

#F.4 cube#####earthdistance###

##	##
earth () → float8	############
sec_to_gc (float8) → float8	###2########################
gc_to_sec (float8) → float8	###2########################
ll_to_earth (float8, float8) → earth	############1#######2################
latitude (earth) → float8	#################
longitude (earth) → float8	#################

[1] https://postgis.net/

##
earth_distance (earth, earth) → float8 ####2############
earth_box (earth, float8) → cube ##################cube#@>########################## ################ ######### #####earth_distance######2####################

F.14.2. point#########

########2###point######################### ####1###########2################ ###### #######X#####Y################(##, ##)#####################

F.5#######1############

#F.5 point#####earthdistance####

##
point <@> point → float8 ###########2###########

########cube### earth()############### ###########

#######1###########±180######################### cube#########################

F.15. file_fdw —

file_fdw##file_fdw###### ##COPY FROM################ ######## ### COPY ######### ##########################

#######################################

filename

##############################filename#program################ ######

program

##############################COPY FROM PROGRAM################ program#filename#######################

format

#################COPY#FORMAT##########

header

##################COPY#HEADER##########

delimiter

#################COPY#DELIMITER##########

quote

#################COPY#QUOTE##########

escape

##################COPY#ESCAPE##########

null

####NULL###########COPY#NULL##########

encoding

####################COPY#ENCODING##########

COPY##HEADER## ######## ###########COPY#######################################TRUE##########

###

force_not_null

##############NULL###(##########null######)########## ## ##COPY#FORCE_NOT_NULL#######################

force_null

############## #####NULL#################################NULL###### ###### ######NULL######################NULL###### ####COPY#FORCE_NULL###### #################

COPY#FORCE_QUOTE#####file_fdw###############

##############################file_fdw#######################################
#####

###
#######################pg_read_server_files#########
#pg_execute_server_program########################## ####################
######################################

program## ####################
##########################
################################

file_fdw#############EXPLAIN############################## COSTS OFF#####
#####(######)###############

#F.1 PostgreSQL CSV

file_fdw#########PostgreSQL########################## ############
#pglog.csv###CSV#################### ###file_fdw#################

```
CREATE EXTENSION file_fdw;
```

###############

```
CREATE SERVER pglog FOREIGN DATA WRAPPER file_fdw;
```

###################### CREATE FOREIGN TABLE#################CSV###########
#################

```
CREATE FOREIGN TABLE pglog (
  log_time timestamp(3) with time zone,
  user_name text,
  database_name text,
  process_id integer,
  connection_from text,
  session_id text,
  session_line_num bigint,
  command_tag text,
  session_start_time timestamp with time zone,
  virtual_transaction_id text,
  transaction_id bigint,
  error_severity text,
  sql_state_code text,
  message text,
  detail text,
  hint text,
  internal_query text,
  internal_query_pos integer,
  context text,
  query text,
  query_pos integer,
  location text,
  application_name text,
  backend_type text,
  leader_pid integer,
  query_id bigint
) SERVER pglog
OPTIONS ( filename 'log/pglog.csv', format 'csv' );
```

######## ######################### ##

#F.2 Create a Foreign Table with an Option on a Column

######null#force_################################

```
CREATE FOREIGN TABLE films (
 code char(5) NOT NULL,
 title text NOT NULL,
 rating text OPTIONS (force_null 'true')
) SERVER film_server
OPTIONS ( filename 'films/db.csv', format 'csv' );
```

F.16. fuzzystrmatch —

fuzzystrmatch################################

##
#####soundex#metaphone#dmetaphone#dmetaphone_alt##UTF-8########## ############### #########daitch_mokotoff###levenshtein#########

#########trusted#########################CREATE#######################

F.16.1. Soundex

Soundex## ####1880##1900##1910########## ######### Soundex################################

fuzzystrmatch#Soundex###########2##########

```
soundex(text) returns text
difference(text, text) returns int
```

soundex#######Soundex######### difference###2#######Soundex############### ########## Soundex####4##############0##4########## 0##############4########## ##### ####################similarity####################

###############

```
SELECT soundex('hello world!');

SELECT soundex('Anne'), soundex('Ann'), difference('Anne', 'Ann');
SELECT soundex('Anne'), soundex('Andrew'), difference('Anne', 'Andrew');
SELECT soundex('Anne'), soundex('Margaret'), difference('Anne',
 'Margaret');

CREATE TABLE s (nm text);

INSERT INTO s VALUES ('john');
INSERT INTO s VALUES ('joan');
INSERT INTO s VALUES ('wobbly');
INSERT INTO s VALUES ('jack');

SELECT * FROM s WHERE soundex(nm) = soundex('john');

SELECT * FROM s WHERE difference(s.nm, 'john') > 2;
```

F.16.2. Daitch-Mokotoff Soundex

##Soundex########Daitch-Mokotoff Soundex############################ ####Daitch-Mokotoff Soundex################################ ########################

- ####4##############6##########
- ##############7#####10###############
- 2########1############1###############
- ##

##########Daitch-Mokotoff soundex#########

daitch_mokotoff(*source* text) returns text[]

#####################1##############################

Daitch-Mokotoff soundex####6##############*source*####################

#############

```
SELECT daitch_mokotoff('George');
 daitch_mokotoff
-----------------
 {595000}

SELECT daitch_mokotoff('John');
 daitch_mokotoff
-----------------
 {160000,460000}

SELECT daitch_mokotoff('Bierschbach');
                      daitch_mokotoff
-----------------------------------------------------------
 {794575,794574,794750,794740,745750,745740,747500,747400}

SELECT daitch_mokotoff('Schwartzenegger');
 daitch_mokotoff
-----------------
 {479465}
```

#######################&&################ ############## ######GIN############
64.4################

```
CREATE TABLE s (nm text);
CREATE INDEX ix_s_dm ON s USING gin (daitch_mokotoff(nm)) WITH (fastupdate
 = off);

INSERT INTO s (nm) VALUES
  ('Schwartzenegger'),
  ('John'),
  ('James'),
  ('Steinman'),
  ('Steinmetz');

SELECT * FROM s WHERE daitch_mokotoff(nm) &&
 daitch_mokotoff('Swartzenegger');
SELECT * FROM s WHERE daitch_mokotoff(nm) && daitch_mokotoff('Jane');
SELECT * FROM s WHERE daitch_mokotoff(nm) && daitch_mokotoff('Jens');
```

#12##################

```
CREATE FUNCTION soundex_tsvector(v_name text) RETURNS tsvector
BEGIN ATOMIC
  SELECT to_tsvector('simple',
                     string_agg(array_to_string(daitch_mokotoff(n), ' '),
 ' '))
  FROM regexp_split_to_table(v_name, '\s+') AS n;
END;

CREATE FUNCTION soundex_tsquery(v_name text) RETURNS tsquery
BEGIN ATOMIC
```

```
  SELECT string_agg('(' || array_to_string(daitch_mokotoff(n), '|') ||
 ')', '&')::tsquery
  FROM regexp_split_to_table(v_name, '\s+') AS n;
END;

CREATE TABLE s (nm text);
CREATE INDEX ix_s_txt ON s USING gin (soundex_tsvector(nm)) WITH
 (fastupdate = off);

INSERT INTO s (nm) VALUES
  ('John Doe'),
  ('Jane Roe'),
  ('Public John Q.'),
  ('George Best'),
  ('John Yamson');

SELECT * FROM s WHERE soundex_tsvector(nm) @@ soundex_tsquery('john');
SELECT * FROM s WHERE soundex_tsvector(nm) @@ soundex_tsquery('jane doe');
SELECT * FROM s WHERE soundex_tsvector(nm) @@ soundex_tsquery('john
 public');
SELECT * FROM s WHERE soundex_tsvector(nm) @@ soundex_tsquery('besst,
 giorgio');
SELECT * FROM s WHERE soundex_tsvector(nm) @@ soundex_tsquery('Jameson
 John');
```

#############soundex## ############## ####### 5.4##########

F.16.3. #########(Levenshtein)

#####2##################(Levenshtein distance)#######

```
levenshtein(source text, target text, ins_cost int, del_cost int, sub_cost
 int) returns int
levenshtein(source text, target text) returns int
levenshtein_less_equal(source text, target text, ins_cost int, del_cost
 int, sub_cost int, max_d int) returns int
levenshtein_less_equal(source text, target text, max_d int) returns int
```

source###target#255########NULL############## ############################## ########## #####2################################ #############1######

levenshtein_less_equal####################levenshtein######### ######max_d# #####levenshtein_less_equal####################max_d############### max_d#######levenshtein###########

#########

```
test=# SELECT levenshtein('GUMBO', 'GAMBOL');
 levenshtein
-------------
           2
(1 row)

test=# SELECT levenshtein('GUMBO', 'GAMBOL', 2, 1, 1);
 levenshtein
-------------
           3
(1 row)
```

```
test=# SELECT levenshtein_less_equal('extensive', 'exhaustive', 2);
 levenshtein_less_equal
------------------------
                       3
(1 row)

test=# SELECT levenshtein_less_equal('extensive', 'exhaustive', 4);
 levenshtein_less_equal
------------------------
                       4
(1 row)
```

F.16.4. Metaphone

Metaphone##Soundex#################################### 2#######################
####

##############metaphone#########

```
metaphone(source text, max_output_length int) returns text
```

source#255#####NULL############# max_output_length###metaphone###########
#####################

#########

```
test=# SELECT metaphone('GUMBO', 4);
 metaphone
-----------
 KM
(1 row)
```

F.16.5. Double Metaphone

Double Metaphone####################primary###alternate####2#####################
######primary####alternate########
##

```
dmetaphone(source text) returns text
dmetaphone_alt(source text) returns text
```

##################

#########

```
test=# SELECT dmetaphone('gumbo');
 dmetaphone
------------
 KMP
(1 row)
```

F.17. hstore — hstore##/#####

#######################PostgreSQL########hstore########## ################## ############################## #################

#########trusted#########################CREATE#######################

F.17.1. hstore#####

############hstore#######################*key => value*############## ######## ##

```
k => v
foo => bar, baz => whatever
"1-a" => "anything at all"
```

#################################### ######=>################# ############ ##=#>################# ##

hstore########## ########hstore######hstore##1########## ############### ##

```
SELECT 'a=>1,a=>2'::hstore;
  hstore
----------
 "a"=>"1"
```

##SQL#NULL################# #########

```
key => NULL
```

NULL################### null#######NULL#######################

> ##
>
> ###########hstore## ###### #hstore#################### ################################ (standard_conforming_strings#############)######################## ###### ##############4.1.2.1#########

##

F.17.2. hstore#######

hstore############### F.6###### F.7######

#F.6 hstore####

###	## # ##
hstore -> text → text	####################NULL###### 'a=>x, b=>y'::hstore -> 'a' → x
hstore -> text[] → text[]	

#####################NULL###### 'a=>x, b=>y, c=>z'::hstore -> ARRAY['c','a'] → {"z","x"}
hstore || hstore → hstore 2##hstore###### 'a=>b, c=>d'::hstore || 'c=>x, d=>q'::hstore → "a"=>"b", "c"=>"x", "d"=>"q"
hstore ? text → boolean hstore####### 'a=>1'::hstore ? 'a' → t
hstore ?& text[] → boolean hstore############## 'a=>1,b=>2'::hstore ?& ARRAY['a','b'] → t
hstore ?| text[] → boolean hstore################ 'a=>1,b=>2'::hstore ?| ARRAY['b','c'] → t
hstore @> hstore → boolean ######### 'a=>b, b=>1, c=>NULL'::hstore @> 'b=>1' → t
hstore <@ hstore → boolean ########### 'a=>c'::hstore <@ 'a=>b, b=>1, c=>NULL' → f
hstore - text → hstore ############ 'a=>1, b=>2, c=>3'::hstore - 'b'::text → "a"=>"1", "c"=>"3"
hstore - text[] → hstore ######(##)####### 'a=>1, b=>2, c=>3'::hstore - ARRAY['a','b'] → "c"=>"3"
hstore - hstore → hstore ############################ 'a=>1, b=>2, c=>3'::hstore - 'a=>4, b=>2'::hstore → "a"=>"1", "c"=>"3"
anyelement #= hstore → anyelement ##(############)#######hstore############ ROW(1,3) #= 'f1=>11'::hstore → (11,3)
%% hstore → text[] hstore#################### %% 'a=>foo, b=>bar'::hstore → {a,foo,b,bar}
%# hstore → text[] hstore######2########## %# 'a=>foo, b=>bar'::hstore → {{a,foo},{b,bar}}

#F.7 hstore###

```
##
      ##
      #
```

```
hstore(record) → hstore
      #########hstore######
      hstore(ROW(1,2)) → "f1"=>"1", "f2"=>"2"
```

```
hstore(text[]) → hstore
      ####hstore#################2############
      hstore(ARRAY['a','1','b','2']) → "a"=>"1", "b"=>"2"
      hstore(ARRAY[['c','3'],['d','4']]) → "c"=>"3", "d"=>"4"
```

```
hstore(text[], text[]) → hstore
      ###########hstore######
      hstore(ARRAY['a','b'], ARRAY['1','2']) → "a"=>"1", "b"=>"2"
```

```
hstore(text, text) → hstore
      hstore############
      hstore('a', 'b') → "a"=>"b"
```

```
akeys(hstore) → text[]
      hstore###############
      akeys('a=>1,b=>2') → {a,b}
```

```
skeys(hstore) → setof text
      hstore###############
      skeys('a=>1,b=>2') →


      a
      b
```

```
avals(hstore) → text[]
      hstore##############
      avals('a=>1,b=>2') → {1,2}
```

```
svals(hstore) → setof text
      hstore##############
      svals('a=>1,b=>2') →


      1
      2
```

```
hstore_to_array(hstore) → text[]
      hstore############################
      hstore_to_array('a=>1,b=>2') → {a,1,b,2}
```

```
hstore_to_matrix(hstore) → text[]
      hstore######2##############
      hstore_to_matrix('a=>1,b=>2') → {{a,1},{b,2}}
```

```
hstore_to_json(hstore) → json
      #null#####JSON##########hstore#json#######
      #####hstore##json####################
```

hstore_to_json('"a key"=>1, b=>t, c=>null, d=>12345, e=>012345, f=>1.234, g=>2.345e+4') → {"a key": "1", "b": "t", "c": null, "d": "12345", "e": "012345", "f": "1.234", "g": "2.345e+4"}
hstore_to_jsonb(hstore) → jsonb #null#####JSON##########hstore#jsonb####### #####hstore##jsonb#################### hstore_to_jsonb('"a key"=>1, b=>t, c=>null, d=>12345, e=>012345, f=>1.234, g=>2.345e+4') → {"a key": "1", "b": "t", "c": null, "d": "12345", "e": "012345", "f": "1.234", "g": "2.345e+4"}
hstore_to_json_loose(hstore) → json hstore#json################################2##JSON############ hstore_to_json_loose('"a key"=>1, b=>t, c=>null, d=>12345, e=>012345, f=>1.234, g=>2.345e+4') → {"a key": 1, "b": true, "c": null, "d": 12345, "e": "012345", "f": 1.234, "g": 2.345e+4}
hstore_to_jsonb_loose(hstore) → jsonb hstore#jsonb################################2##JSON############ hstore_to_jsonb_loose('"a key"=>1, b=>t, c=>null, d=>12345, e=>012345, f=>1.234, g=>2.345e+4') → {"a key": 1, "b": true, "c": null, "d": 12345, "e": "012345", "f": 1.234, "g": 2.345e+4}
slice(hstore, text[]) → hstore ###########hstore############ slice('a=>1,b=>2,c=>3'::hstore, ARRAY['b','c','x']) → "b"=>"2", "c"=>"3"
each(hstore) → setof record(*key* text, *value* text) hstore###################### select * from each('a=>1,b=>2') → key | value -----+------- a | 1 b | 2
exist(hstore, text) → boolean hstore####### exist('a=>1', 'a') → t
defined(hstore, text) → boolean hstore#######NULL###### defined('a=>NULL', 'a') → f
delete(hstore, text) → hstore ################## delete('a=>1,b=>2', 'b') → "a"=>"1"
delete(hstore, text[]) → hstore

#
##(##)################ delete('a=>1,b=>2,c=>3', ARRAY['a','b']) → "c"=>"3"
delete (hstore, hstore) → hstore #2####################### delete('a=>1,b=>2', 'a=>4,b=>2'::hstore) → "a"=>"1"
populate_record (anyelement, hstore) → anyelement ##(############)######hstore############ populate_record(ROW(1,2), 'f1=>42'::hstore) → (42,2)

############hstore################################## text################## ################################## #########

```
CREATE TABLE mytable (h hstore);
INSERT INTO mytable VALUES ('a=>b, c=>d');
SELECT h['a'] FROM mytable;
 h
---
 b
(1 row)

UPDATE mytable SET h['c'] = 'new';
SELECT h FROM mytable;
          h
----------------------
 "a"=>"b", "c"=>"new"
(1 row)
```

###NULL#########hstore####################NULL##### (###############->## ############) ###NULL###
#hstore###########

F.17.3.

hstore#@>#?#?&###?|#####GiST###GIN############## #########

```
CREATE INDEX hidx ON testhstore USING GIST (h);

CREATE INDEX hidx ON testhstore USING GIN (h);
```

gist_hstore_ops GiST########/#################### ############siglen##### ############## ###########16###### ##########1##2024####### ################## ##########(################################)#############

######32###############

```
CREATE INDEX hidx ON testhstore USING GIST (h gist_hstore_ops(siglen=32));
```

hstore####=######btree###hash############## #####hstore###UNIQUE######## ##GROUP BY#ORDER BY#DISTINCT##############hstore#################### ## ################################ =######################

```
CREATE INDEX hidx ON testhstore USING BTREE (h);
```

```
CREATE INDEX hidx ON testhstore USING HASH (h);
```

F.17.4.

```
##########################

UPDATE tab SET h['c'] = '3';

####################

UPDATE tab SET h = h || hstore('c', '3');

1##############################################

UPDATE tab SET h = h || hstore(array['q', 'w'], array['11', '12']);

#########

UPDATE tab SET h = delete(h, 'k1');

record#hstore######

CREATE TABLE test (col1 integer, col2 text, col3 text);
INSERT INTO test VALUES (123, 'foo', 'bar');

SELECT hstore(t) FROM test AS t;
                   hstore
---------------------------------------------
 "col1"=>"123", "col2"=>"foo", "col3"=>"bar"
(1 row)

hstore########record#######

CREATE TABLE test (col1 integer, col2 text, col3 text);

SELECT * FROM populate_record(null::test,
                              '"col1"=>"456", "col2"=>"zzz"');
 col1 | col2 | col3
------+------+------
  456 | zzz  |
(1 row)

hstore####################

CREATE TABLE test (col1 integer, col2 text, col3 text);
INSERT INTO test VALUES (123, 'foo', 'bar');

SELECT (r).* FROM (SELECT t #= '"col3"=>"baz"' AS r FROM test t) s;
 col1 | col2 | col3
------+------+------
  123 | foo  | baz
(1 row)
```

F.17.5.

```
###########hstore################### ########################### #############
######################
```

```
##########

SELECT * FROM each('aaa=>bq, b=>NULL, ""=>1');

#############

CREATE TABLE stat AS SELECT (each(h)).key, (each(h)).value FROM
 testhstore;

###########

SELECT key, count(*) FROM
  (SELECT (each(h)).key FROM testhstore) AS stat
  GROUP BY key
  ORDER BY count DESC, key;
    key    | count
-----------+-------
 line      |   883
 query     |   207
 pos       |   203
 node      |   202
 space     |   197
 status    |   195
 public    |   194
 title     |   190
 org       |   189
..................
```

F.17.6.

PostgreSQL 9.0##hstore################### (##########)######################/#### ################

```
################################################## ############################## ################## ######UPDATE######################################

UPDATE tablename SET hstorecol = hstorecol || '';

#################

ALTER TABLE tablename ALTER hstorecol TYPE hstore USING hstorecol || '';

ALTER TABLE##############ACCESS EXCLUSIVE########## ######################
##########
```

F.17.7.

PL/Perl###PL/Python#####hstore###################### PL/Perl############PL/Perl#### #hstore_plperl###################hstore_plperlu######## ################## ############hstore###Perl############## PL/Python######hstore_plpython3u#### #### ########hstore###Python############

```
                                ##

 ######hstore######################### ################################
 ####################################
```

F.17.8.

Oleg Bartunov <oleg@sai.msu.su>, Moscow, Moscow University, Russia

Teodor Sigaev <teodor@sigaev.ru>, Moscow, Delta-Soft Ltd., Russia

######Andrew Gierth <andrew@tao11.riddles.org.uk>,United Kingdom##########

F.18. intagg —

intagg#################### ###########################intagg######### ## ####################################

F.18.1.

###########################int_array_aggregate(integer)###### ######### #######array_agg#######

#####setof integer###int_array_enum(integer[])##### ################### ### ################### ################unnest#######

F.18.2.

#####[93.023256]##############1############ #################2################ ######

```
CREATE TABLE left_table  (id INT PRIMARY KEY, ...);
CREATE TABLE right_table (id INT PRIMARY KEY, ...);
CREATE TABLE many_to_many(id_left  INT REFERENCES left_table,
                          id_right INT REFERENCES right_table);
```

##############

```
SELECT right_table.*
FROM right_table JOIN many_to_many ON (right_table.id =
 many_to_many.id_right)
WHERE many_to_many.id_left = item;
```

###SQL##############

#####[95.890411]#########one_to_many################################## ###### ## ################### ####### ##

```
CREATE TABLE summary AS
  SELECT id_left, int_array_aggregate(id_right) AS rights
  FROM many_to_many
  GROUP BY id_left;
```

#########1########################## ##################################### #### ############# ############

```
SELECT id_left, int_array_enum(rights) FROM summary WHERE id_left = item;
```

##int_array_enum########################

```
SELECT id_left, id_right FROM many_to_many WHERE id_left = item;
```

#####[95.852535]##########################1##################one_to_many#### ##

#####[95.475113]#########EXPLAIN####8488##############329############## ####### #one_to_many############################

```
SELECT id_right, count(id_right) FROM
  ( SELECT id_left, int_array_enum(rights) AS id_right
    FROM summary
    JOIN (SELECT id FROM left_table
          WHERE id = item) AS lefts
    ON (summary.id_left = lefts.id)
  ) AS list
  GROUP BY id_right
  ORDER BY count DESC;
```

F.19. intarray —

intarray######NULL############################ ############################
##

###NULL################################

########################### ##

#########trusted#########################CREATE########################

F.19.1. intarray#########

intarray############## F.8####### F.9######

#F.8 intarray##

icount (integer[]) → integer ############ icount('{1,2,3}'::integer[]) → 3
sort (integer[], *dir* text) → integer[] ################# *dir*#asc###desc############### sort('{1,3,2}'::integer[], 'desc') → {3,2,1}
sort (integer[]) → integer[] sort_asc (integer[]) → integer[] ######### sort(array[11,77,44]) → {11,44,77}
sort_desc (integer[]) → integer[] ######### sort_desc(array[11,77,44]) → {77,44,11}
uniq (integer[]) → integer[] ############ ##################sort########## uniq('{1,2,2,3,1,1}'::integer[]) → {1,2,3,1} uniq(sort('{1,2,3,2,1}'::integer[])) → {1,2,3}
idx (integer[], *item* integer) → integer *item*#######################0###### idx(array[11,22,33,22,11], 22) → 2
subarray (integer[], *start* integer, *len* integer) → integer[] *start*#######*len*############### subarray('{1,2,3,2,1}'::integer[], 2, 3) → {2,3,2}
subarray (integer[], *start* integer) → integer[] *start*################## subarray('{1,2,3,2,1}'::integer[], 2) → {2,3,2,1}
intset (integer) → integer[] ##############

#
intset(42) → {42}

#F.9 intarray###

##
integer[] && integer[] → boolean ############1###########
integer[] @> integer[] → boolean ###############
integer[] <@ integer[] → boolean ##################
integer[] → integer
integer[] # integer → integer ##############################0###### (idx########)
integer[] + integer → integer[] ##############
integer[] + integer[] → integer[] ########
integer[] - integer → integer[] ######################
integer[] - integer[] → integer[] ####################
integer[] | integer → integer[] ###########
integer[] | integer[] → integer[] ###########
integer[] & integer[] → integer[] #############
integer[] @@ query_int → boolean #############(####)
query_int ~~ integer[] → boolean #############(@@######)

###&&#@>#<@######NULL## PostgreSQL################### ###########################

@@###~~###############query_int######################### ##########&(###)#|(###)#! (##)##################################### ###1&(2|3)########1####2#3### #############

F.19.2.

intarray#&&#@>#@@##################################

2##########GiST################### gist__int_ops########################### ####### ###gist__intbig_ops## ########## ################RD-tree###########

gist__int_ops######################## ############numranges############### ########### numranges#######100### #####1##253##### GiST#################### ####################(################################)#############

gist__intbig_ops######################## ############siglen############### #### ###########16###### ##########1##2024####### ########################### (################################)#############

###########GIN######gin__int_ops#####<@###################

GiST###GIN#########################GiST#GIN###############

F.19.3.

```
-- #####(message)#1######(section)######
CREATE TABLE message (mid INT PRIMARY KEY, sections INT[], ...);


-- ######32##################
CREATE INDEX message_rdtree_idx ON message USING GIST (sections
 gist__intbig_ops (siglen = 32));


-- #1 OR 2######### - OVERLAP###
SELECT message.mid FROM message WHERE message.sections && '{1,2}';


-- #1 AND 2######### - CONTAINS###
SELECT message.mid FROM message WHERE message.sections @> '{1,2}';


-- ###QUERY######
SELECT message.mid FROM message WHERE message.sections @@
 '1&2'::query_int;
```

F.19.4.

###########contrib/intarray/bench#########################PostgreSQL####### #### (DBD::Pg###################) ############

```
cd .../contrib/intarray/bench
createdb TEST
psql -c "CREATE EXTENSION intarray" TEST
./create_test.pl | psql TEST
./bench.pl
```

bench.pl################### ########################

F.19.5.

Teodor Sigaev (<teodor@sigaev.ru>)#Oleg Bartunov (<oleg@sai.msu.su>)########## ####### ##########http://www.sai.msu.su/~megera/postgres/gist/########## Andrey Oktyabrski ################################

F.20. isn — #######ISBN#EAN#UPC#######

isn#######EAN13#UPC#ISBN (##)#ISMN (##)#ISSN (#####)##

#########trusted#########################CREATE#########################

F.20.1.

F.10#isn####################

#F.10 isn####

####	##
EAN13	############ ##EAN13############
ISBN13	######### ###EAN13############
ISMN13	######### ###EAN13############
ISSN13	############ ###EAN13############
ISBN	######### #################
ISMN	######### #################
ISSN	############ #################
UPC	########

############

1. ISBN13#ISMN13#ISSN13######EAN13#####
2. EAN13#####ISBN13#ISMN13#ISSN13################### (#######)#
3. ###ISBN13###ISBN##########
4. ###ISMN13###ISMN##########
5. ###ISSN13###ISSN##########
6. UPC###EAN13###############EAN13#####0###############
7. ####UPC#ISBN#ISMN#ISSN#EAN13############

################64######################### ##

ISBN#ISMN#ISSN###################ISxN 10######### #############ISxN 13######## EAN13#ISBN13#ISMN13#ISSN13#####ISxN#######EAN13#########

F.20.2.

isn#############################

- ISBN13 <=> EAN13

- ISMN13 <=> EAN13
- ISSN13 <=> EAN13
- ISBN <=> EAN13
- ISMN <=> EAN13
- ISSN <=> EAN13
- UPC <=> EAN13
- ISBN <=> ISBN13
- ISMN <=> ISMN13
- ISSN <=> ISSN13

EAN13## ################### ###### ############################

F.20.3.

isn############################B-tree####################### ##### F.11######## ############ ######isn#################1########

#F.11 isn Functions

##
isn_weak (boolean) → boolean   weak###################
isn_weak () → boolean   weak##############
make_valid (isn) → isn   ########################
is_valid (isn) → boolean   ##############

weak################################### ####################################

##weak#################### ####ISBN### ### ##############OCR################################### ## ############ ################################

weak###!####### #0-11-000322-5!####### ######is_valid################## make_valid####### ########

#########!###########weak############################

##############################?########## ############################

F.20.4.

```
---######
SELECT isbn('978-0-393-04002-9');
SELECT isbn13('0901690546');
SELECT issn('1436-4522');


--#####
-- ####################ean13##################
-- ####################: select isbn(ean13('0220356483481'));
-- ###########
SELECT upc(ean13('0220356483481'));
SELECT ean13(upc('220356483481'));


--ISBN######################
CREATE TABLE test (id isbn);
INSERT INTO test VALUES('9780393040029');


--##################('?'###)
INSERT INTO test VALUES('220500896?');
INSERT INTO test VALUES('978055215372?');

SELECT issn('3251231?');
SELECT ismn('979047213542?');


--weak########
SELECT isn_weak(true);
INSERT INTO test VALUES('978-0-11-000533-4');
INSERT INTO test VALUES('9780141219307');
INSERT INTO test VALUES('2-205-00876-X');
SELECT isn_weak(false);

SELECT id FROM test WHERE NOT is_valid(id);
UPDATE test SET id = make_valid(id) WHERE id = '2-205-00876-X!';

SELECT * FROM test;

SELECT isbn13(id) FROM test;
```

F.20.5.

##

- https://www.isbn-international.org/
- https://www.issn.org/
- https://www.ismn-international.org/
- https://www.wikipedia.org/

###########################

- https://www.gs1.org/standards/id-keys
- https://en.wikipedia.org/wiki/List_of_ISBN_identifier_groups
- https://www.isbn-international.org/content/isbn-users-manual
- https://en.wikipedia.org/wiki/International_Standard_Music_Number

- https://www.ismn-international.org/ranges.html

##################ISBN#ISMN#ISSN##

F.20.6.

Germán Méndez Bravo (Kronuz), 2004–2006

#######Garrett A. Wollman#`isbn_issn`##############

F.21. lo —

lo###############LO#BLOB########################## lo#####lo_manage##########

#########trusted##########################CREATE#########################

F.21.1.

JDBC###########1##ODBC####################BLOB############################ #################BLOB########################

PostgreSQL############### ################################ #######OID######## ############################OID####################### ######################## ##############################

######PostgreSQL######################JDBC#ODBC############################## ####################################

lo########LO################################ ################################ ########lo_unlink######## ## #########

############oid###########lo########### ##########################OID###### ######################### ###############lo############### ######################### ## ###BLOB##lo#########ODBC###### ##################

F.21.2.

############

```
CREATE TABLE image (title text, raster lo);

CREATE TRIGGER t_raster BEFORE UPDATE OR DELETE ON image
    FOR EACH ROW EXECUTE FUNCTION lo_manage(raster);
```

##########################BEFORE UPDATE OR DELETE############# ############# ############## BEFORE UPDATE OF *column_name*################################ #### ###########lo#################################### ######################## ###########

F.21.3.

- ## DROP TABLE###DELETE FROM *table*# ################

  TRUNCATE############

  ##vacuumlo################ lo_manage#############vacuumlo###############

- ## ############################ ##########

F.21.4.

Peter Mount <peter@retep.org.uk>

F.22. ltree —

################################### ltree########## ######################

#########trusted#######################CREATE######################

F.22.1.

######################### ######################### ####C######A-Za-z0-9_-########### #######1000##############

###42#Personal_Services###

#########L1.L2.L3############0### ##### #####65535###############

##'Top.Countries.Europe.Russia'

ltree#####################

- ltree############
- lquery##ltree####################### ################### ######*##0####### ####### ###################################### ##########

```
foo          ###foo###############
*.foo.*      foo#######################
*.foo        foo########################

##############################################
```

```
*{n}         ###n############
*{n,}        #####n############
*{n,m}       #####n##########m################
*{,m}        ##m#######################*{0,m}
foo{n,m}     #####n##########m#######foo#######
foo{,}       ##########foo#######

##########################################(###{,})#######################1#(###{1})###

#########################lquery######################
```

```
@            ##################a@#A#######
*            #######################foo*#foobar#######
%            ######################

%##########       ##################       ###foo_bar%#foo_bar_baz#####
#foo_barbaz#######       *############################       ###foo_bar
%*#foo1_bar2_baz#######foo1_br2_baz#######

##################|############################################  ########
######! (##)#################################### ##########################
###################(##################################)#

#######lquery#######
```

```
Top.*{0,2}.sport*@.!football|tennis{1,}.Russ*|Spain
a.  b.     c.      d.                    e.
```

#######################

a. Top########

b. ###0##2########

c. ###sport#########################

d. ####football##tennis######1##########

e. Russ############Spain############

- ltxtquery#ltree#################### ltxtquery#########@#*#%########## ##### #######lquery###### ###& (###)#| (###)#! (##)################ ##lquery### ###ltxtquery##########################

  ltxtquery#######

  ```
  Europe & Russia*@ & !Transportation
  ```

  ###Europe####Russia############################### ####Transportation#### ########### ################### ###%###

  ################

###ltxtquery#######################ltree#lquery#######

F.22.2.

ltree##########=#<>#<#>#<=#>=###### ## ##### F.12################

#F.12 ltree###

##
ltree @> ltree → boolean ####################
ltree <@ ltree → boolean ####################
ltree ~ lquery → boolean lquery ~ ltree → boolean ltree#lquery#########
ltree ? lquery[] → boolean lquery[] ? ltree → boolean ltree#########lquery#########
ltree @ ltxtquery → boolean ltxtquery @ ltree → boolean ltree#ltxtquery#########
ltree || ltree → ltree

ltree########
ltree || text → ltree text || ltree → ltree #####ltree##########
ltree[] @> ltree → boolean ltree <@ ltree[] → boolean ###ltree##############
ltree[] <@ ltree → boolean ltree @> ltree[] → boolean ###ltree##############
ltree[] ~ lquery → boolean lquery ~ ltree[] → boolean ###lquery###############
ltree[] ? lquery[] → boolean lquery[] ? ltree[] → boolean ltree#######lquery###############
ltree[] @ ltxtquery → boolean ltxtquery @ ltree[] → boolean ###ltxtquery###############
ltree[] ?@> ltree → ltree ltree#######################NULL######
ltree[] ?<@ ltree → ltree ltree#######################NULL######
ltree[] ?~ lquery → ltree lquery######################NULL######
ltree[] ?@ ltxtquery → ltree ltxtquery######################NULL######

###<@#@>#@#~########^<@#^@>#^@#^~###### ######################## ############ #####

######### F.13######

#F.13 ltree##

subltree (ltree, *start* integer, *end* integer) → ltree *start*####*end*-1#####ltree#############0######## subltree('Top.Child1.Child2', 1, 2) → Child1
subpath (ltree, *offset* integer, *len* integer) → ltree *offset*####*len*##ltree########## *offset*########################### ## *len*####################### subpath('Top.Child1.Child2', 0, 2) → Top.Child1

subpath (ltree, *offset* integer) → ltree *offset*##########ltree########## *offset*########################## #### subpath('Top.Child1.Child2', 1) → Child1.Child2
nlevel (ltree) → integer ############# nlevel('Top.Child1.Child2') → 3
index (*a* ltree, *b* ltree) → integer *a*##*b*##################-1###### index('0.1.2.3.5.4.5.6.8.5.6.8', '5.6') → 6
index (*a* ltree, *b* ltree, *offset* integer) → integer *a*##*b*##################-1###### ###*offset*#########*offset*#######-*offset*################## index('0.1.2.3.5.4.5.6.8.5.6.8', '5.6', -4) → 9
text2ltree (text) → ltree text#ltree########
ltree2text (ltree) → text ltree#text########
lca (ltree [, ltree [, ...]]) → ltree ####################8############ lca('1.2.3', '1.2.3.4.5.6') → 1.2
lca (ltree[]) → ltree ###################### lca(array['1.2.3'::ltree,'1.2.3.4']) → 1.2

F.22.3.

ltree#######################################

- ltree####B-tree#######<#<=#=#>=#>
- #####[50.000000]#ltree### ###################ltree:=.
- ltree####GiST######(gist_ltree_ops######)# <#<=#=#>=#>#@>#<@#@#~#?

  gist_ltree_ops GiST################################ ############siglen### ################ ############8###### ####int##(########4###)###########2024## ######## ############################(################################)#### ########

  ###########8################

  ```
  CREATE INDEX path_gist_idx ON test USING GIST (path);
  ```

  ######100################

```
CREATE INDEX path_gist_idx ON test USING GIST (path
 gist_ltree_ops(siglen=100));
```

- ltree[]####GiST######gist__ltree_ops#######ltree[] <@ ltree#ltree @> ltree[]#@#~#?

  gist__ltree_ops GiST######gist_ltree_ops##############################gist__ltree_ops##siglen########28######

  ###########28################

```
CREATE INDEX path_gist_idx ON test USING GIST (array_path);
```

  ######100################

```
CREATE INDEX path_gist_idx ON test USING GIST (array_path
 gist__ltree_ops(siglen=100));
```

  ####################

F.22.4.

#######################contrib/ltree/ltreetest.sql#############

```
CREATE TABLE test (path ltree);
INSERT INTO test VALUES ('Top');
INSERT INTO test VALUES ('Top.Science');
INSERT INTO test VALUES ('Top.Science.Astronomy');
INSERT INTO test VALUES ('Top.Science.Astronomy.Astrophysics');
INSERT INTO test VALUES ('Top.Science.Astronomy.Cosmology');
INSERT INTO test VALUES ('Top.Hobbies');
INSERT INTO test VALUES ('Top.Hobbies.Amateurs_Astronomy');
INSERT INTO test VALUES ('Top.Collections');
INSERT INTO test VALUES ('Top.Collections.Pictures');
INSERT INTO test VALUES ('Top.Collections.Pictures.Astronomy');
INSERT INTO test VALUES ('Top.Collections.Pictures.Astronomy.Stars');
INSERT INTO test VALUES ('Top.Collections.Pictures.Astronomy.Galaxies');
INSERT INTO test VALUES ('Top.Collections.Pictures.Astronomy.Astronauts');
CREATE INDEX path_gist_idx ON test USING GIST (path);
CREATE INDEX path_idx ON test USING BTREE (path);
CREATE INDEX path_hash_idx ON test USING HASH (path);
```

#######################test#########

```
                        Top
                     /   |  \
             Science Hobbies Collections
                 /       |              \
        Astronomy   Amateurs_Astronomy Pictures
           /  \                            |
Astrophysics  Cosmology                Astronomy
                                        /  |    \
                                 Galaxies Stars Astronauts
```

#############

```
ltreetest=> SELECT path FROM test WHERE path <@ 'Top.Science';
                path
------------------------------------
```

```
 Top.Science
 Top.Science.Astronomy
 Top.Science.Astronomy.Astrophysics
 Top.Science.Astronomy.Cosmology
(4 rows)
```

################

```
ltreetest=> SELECT path FROM test WHERE path ~ '*.Astronomy.*';
                     path
-----------------------------------------------
 Top.Science.Astronomy
 Top.Science.Astronomy.Astrophysics
 Top.Science.Astronomy.Cosmology
 Top.Collections.Pictures.Astronomy
 Top.Collections.Pictures.Astronomy.Stars
 Top.Collections.Pictures.Astronomy.Galaxies
 Top.Collections.Pictures.Astronomy.Astronauts
(7 rows)

ltreetest=> SELECT path FROM test WHERE path ~ '*.!pictures@.Astronomy.*';
                path
-------------------------------------
 Top.Science.Astronomy
 Top.Science.Astronomy.Astrophysics
 Top.Science.Astronomy.Cosmology
(3 rows)
```

################

```
ltreetest=> SELECT path FROM test WHERE path @ 'Astro*% & !pictures@';
                path
-------------------------------------
 Top.Science.Astronomy
 Top.Science.Astronomy.Astrophysics
 Top.Science.Astronomy.Cosmology
 Top.Hobbies.Amateurs_Astronomy
(4 rows)

ltreetest=> SELECT path FROM test WHERE path @ 'Astro* & !pictures@';
                path
-------------------------------------
 Top.Science.Astronomy
 Top.Science.Astronomy.Astrophysics
 Top.Science.Astronomy.Cosmology
(3 rows)
```

################

```
ltreetest=> SELECT subpath(path,0,2)||'Space'||subpath(path,2) FROM test
 WHERE path <@ 'Top.Science.Astronomy';
                 ?column?
-----------------------------------------
 Top.Science.Space.Astronomy
 Top.Science.Space.Astronomy.Astrophysics
 Top.Science.Space.Astronomy.Cosmology
(3 rows)
```

###############SQL#########################

```
CREATE FUNCTION ins_label(ltree, int, text) RETURNS ltree
```

```
    AS 'select subpath($1,0,$2) || $3 || subpath($1,$2);'
    LANGUAGE SQL IMMUTABLE;

ltreetest=> SELECT ins_label(path,2,'Space') FROM test WHERE path <@
 'Top.Science.Astronomy';
                ins_label
-----------------------------------------
 Top.Science.Space.Astronomy
 Top.Science.Space.Astronomy.Astrophysics
 Top.Science.Space.Astronomy.Cosmology
(3 rows)
```

F.22.5.

`ltree_plpython3u`####PL/Python##`ltree`########## ############################ ##`ltree`###Python############ (###########################)

> ##
>
> ######`ltree`######################## ################################## ##################################

F.22.6.

######Teodor Sigaev (<`teodor@stack.net`>)#Oleg Bartunov (<`oleg@sai.msu.su`>)# ######### ###########http://www.sai.msu.su/~megera/postgres/gist/########## ############ Eugeny Rodichev######### #################

F.23. pageinspect —

pageinspect## ########### ####################

F.23.1.

get_raw_page(relname text, fork text, blkno bigint) returns bytea

get_raw_page##############################bytea############## ######### ########################### *fork*###########'main'#########'fsm'###### ###'vm'#########'init'###########

get_raw_page(relname text, blkno bigint) returns bytea

get_raw_page#################### get_raw_page(relname, 'main', blkno)######

page_header(page bytea) returns record

page_header######PostgreSQL############################

get_raw_page############################ ##########

```
test=# SELECT * FROM page_header(get_raw_page('pg_class', 0));
    lsn    | checksum | flags  | lower | upper | special | pagesize |
 version | prune_xid
-----------+----------+--------+-------+-------+---------+----------
+---------+-----------
 0/24A1B50 |        0 |      1 |   232 |   368 |    8192 |     8192 |
     4 |         0
```

#######PageHeaderData############### ###src/include/storage/bufpage.h#########

checksum## ##

page_checksum(page bytea, blkno bigint) returns smallint

page_checksum####################################

get_raw_page############################ ##########

```
test=# SELECT page_checksum(get_raw_page('pg_class', 0), 0);
 page_checksum
---------------
         13443
```

####################(##############)################################

#################page_header###checksum############## ##

fsm_page_contents(page bytea) returns text

fsm_page_contents##FSM################ ##########

```
test=# SELECT fsm_page_contents(get_raw_page('pg_class', 'fsm', 0));
```

###################################1######## ########################### ## ######################"next###"################

FSM##################src/backend/storage/freespace/README#########

F.23.2.

heap_page_items(page bytea) returns setof record

heap_page_items###################### ################################ ### #############MVCC#####################################

get_raw_page################################ ##########

```
test=# SELECT * FROM heap_page_items(get_raw_page('pg_class', 0));
```

#################src/include/storage/itemid.h###src/include/access/htup_details.h#########

heap_tuple_infomask_flags###############t_infomask###t_infomask2## #################

tuple_data_split(rel_oid oid, t_data bytea, t_infomask integer, t_infomask2 integer, t_bits text [, do_detoast bool]) returns bytea[]

tuple_data_split##################################

```
test=# SELECT tuple_data_split('pg_class'::regclass,
 t_data, t_infomask, t_infomask2, t_bits) FROM
 heap_page_items(get_raw_page('pg_class', 0));
```

#####heap_page_items######################

do_detoast#true#############TOAST###### #######false###

heap_page_item_attrs(page bytea, rel_oid regclass [, do_detoast bool]) returns setof record

heap_page_item_attrs#heap_page_items###############*do_detoast*###### #false######TOAST######################

get_raw_page############################## ##########

```
test=# SELECT * FROM heap_page_item_attrs(get_raw_page('pg_class', 0),
 'pg_class'::regclass);
```

heap_tuple_infomask_flags(t_infomask integer, t_infomask2 integer) returns record

heap_tuple_infomask_flags##heap_page_items#####t_infomask## #t_infomask2################################## ######################### ############ ##########

```
test=# SELECT t_ctid, raw_flags, combined_flags
         FROM heap_page_items(get_raw_page('pg_class', 0)),
           LATERAL heap_tuple_infomask_flags(t_infomask, t_infomask2)
```

```
        WHERE t_infomask IS NOT NULL OR t_infomask2 IS NOT NULL;
```

#####heap_page_items#####################

########## HEAP_XMIN_FROZEN######raw##################################

################src/include/access/htup_details.h#########

F.23.3. B-tree##

bt_metap(relname text) returns record

bt_metap#B-tree######################## ##########

```
test=# SELECT * FROM bt_metap('pg_cast_oid_index');
-[ RECORD 1 ]-------------+-------
magic                     | 340322
version                   | 4
root                      | 1
level                     | 0
fastroot                  | 1
fastlevel                 | 0
last_cleanup_num_delpages | 0
last_cleanup_num_tuples   | 230
allequalimage             | f
```

bt_page_stats(relname text, blkno bigint) returns record

bt_page_stats##B-tree########################## ##########

```
test=# SELECT * FROM bt_page_stats('pg_cast_oid_index', 1);
-[ RECORD 1 ]-+-----
blkno         | 1
type          | l
live_items    | 224
dead_items    | 0
avg_item_size | 16
page_size     | 8192
free_size     | 3668
btpo_prev     | 0
btpo_next     | 0
btpo_level    | 0
btpo_flags    | 3
```

bt_multi_page_stats(relname text, blkno bigint, blk_count bigint) returns setof record

bt_multi_page_stats#bt_page_stats###########*blkno*#####*blk_count*### ################## *blk_count*######*blkno*##index################### ####### #######

```
test=# SELECT * FROM bt_multi_page_stats('pg_proc_oid_index', 5, 2);
-[ RECORD 1 ]-+-----
blkno         | 5
type          | l
live_items    | 367
dead_items    | 0
avg_item_size | 16
page_size     | 8192
free_size     | 808
btpo_prev     | 4
```

```
btpo_next     | 6
btpo_level    | 0
btpo_flags    | 1
-[ RECORD 2 ]-+-----
blkno         | 6
type          | l
live_items    | 367
dead_items    | 0
avg_item_size | 16
page_size     | 8192
free_size     | 808
btpo_prev     | 5
btpo_next     | 7
btpo_level    | 0
btpo_flags    | 1
```

bt_page_items(relname text, blkno bigint) returns setof record

bt_page_items##B-tree########################## #########

```
test=# SELECT itemoffset, ctid, itemlen, nulls, vars, data, dead, htid,
 tids[0:2] AS some_tids
        FROM bt_page_items('tenk2_hundred', 5);
 itemoffset |   ctid    | itemlen | nulls | vars |          data
    | dead |  htid  |      some_tids
------------+-----------+---------+-------+------
+-------------------------+------+--------+---------------------
          1 | (16,1)    |      16 | f     | f    | 30 00 00 00 00 00 00
 00 |      |        |
          2 | (16,8292) |     616 | f     | f    | 24 00 00 00 00 00 00
 00 | f    | (1,6)  | {"(1,6)","(10,22)"}
          3 | (16,8292) |     616 | f     | f    | 25 00 00 00 00 00 00
 00 | f    | (1,18) | {"(1,18)","(4,22)"}
          4 | (16,8292) |     616 | f     | f    | 26 00 00 00 00 00 00
 00 | f    | (4,18) | {"(4,18)","(6,17)"}
          5 | (16,8292) |     616 | f     | f    | 27 00 00 00 00 00 00
 00 | f    | (1,2)  | {"(1,2)","(1,19)"}
          6 | (16,8292) |     616 | f     | f    | 28 00 00 00 00 00 00
 00 | f    | (2,24) | {"(2,24)","(4,11)"}
          7 | (16,8292) |     616 | f     | f    | 29 00 00 00 00 00 00
 00 | f    | (2,17) | {"(2,17)","(11,2)"}
          8 | (16,8292) |     616 | f     | f    | 2a 00 00 00 00 00 00
 00 | f    | (0,25) | {"(0,25)","(3,20)"}
          9 | (16,8292) |     616 | f     | f    | 2b 00 00 00 00 00 00
 00 | f    | (0,10) | {"(0,10)","(0,14)"}
         10 | (16,8292) |     616 | f     | f    | 2c 00 00 00 00 00 00
 00 | f    | (1,3)  | {"(1,3)","(3,9)"}
         11 | (16,8292) |     616 | f     | f    | 2d 00 00 00 00 00 00
 00 | f    | (6,28) | {"(6,28)","(11,1)"}
         12 | (16,8292) |     616 | f     | f    | 2e 00 00 00 00 00 00
 00 | f    | (0,27) | {"(0,27)","(1,13)"}
         13 | (16,8292) |     616 | f     | f    | 2f 00 00 00 00 00 00
 00 | f    | (4,17) | {"(4,17)","(4,21)"}
(13 rows)
```

###B-tree######### ##6###TID### 100######## ###itemoffset#########(high key)######### ################### ##########ctid#######################TID###ctid################### tids################TID#######

##################ctid#################(downlink)######################### ## ctid## ######################### #################(minus infinity)###############

htid#######################TID###### #####ctid#######################
################################### ####################TID#########NULL
htid#########

#########btpo_next########0### ####
###################data###########ctid########################### #######
##data################ #######
#########ctid#######################

B-Tree################64.1.4.1######### ######################64.1.4.3########
##

bt_page_items(page bytea) returns setof record

####bt_page_items#bytea############## get_raw_page##################
########## ##########################

```
test=# SELECT itemoffset, ctid, itemlen, nulls, vars, data, dead, htid,
 tids[0:2] AS some_tids
        FROM bt_page_items(get_raw_page('tenk2_hundred', 5));
 itemoffset |   ctid    | itemlen | nulls | vars |          data
    | dead |  htid  |      some_tids
------------+-----------+---------+-------+------
+-------------------------+------+--------+--------------------
          1 | (16,1)    |      16 | f     | f    | 30 00 00 00 00 00 00
 00 |      |        |
          2 | (16,8292) |     616 | f     | f    | 24 00 00 00 00 00 00
 00 | f    | (1,6)  | {"(1,6)","(10,22)"}
          3 | (16,8292) |     616 | f     | f    | 25 00 00 00 00 00 00
 00 | f    | (1,18) | {"(1,18)","(4,22)"}
          4 | (16,8292) |     616 | f     | f    | 26 00 00 00 00 00 00
 00 | f    | (4,18) | {"(4,18)","(6,17)"}
          5 | (16,8292) |     616 | f     | f    | 27 00 00 00 00 00 00
 00 | f    | (1,2)  | {"(1,2)","(1,19)"}
          6 | (16,8292) |     616 | f     | f    | 28 00 00 00 00 00 00
 00 | f    | (2,24) | {"(2,24)","(4,11)"}
          7 | (16,8292) |     616 | f     | f    | 29 00 00 00 00 00 00
 00 | f    | (2,17) | {"(2,17)","(11,2)"}
          8 | (16,8292) |     616 | f     | f    | 2a 00 00 00 00 00 00
 00 | f    | (0,25) | {"(0,25)","(3,20)"}
          9 | (16,8292) |     616 | f     | f    | 2b 00 00 00 00 00 00
 00 | f    | (0,10) | {"(0,10)","(0,14)"}
         10 | (16,8292) |     616 | f     | f    | 2c 00 00 00 00 00 00
 00 | f    | (1,3)  | {"(1,3)","(3,9)"}
         11 | (16,8292) |     616 | f     | f    | 2d 00 00 00 00 00 00
 00 | f    | (6,28) | {"(6,28)","(11,1)"}
         12 | (16,8292) |     616 | f     | f    | 2e 00 00 00 00 00 00
 00 | f    | (0,27) | {"(0,27)","(1,13)"}
         13 | (16,8292) |     616 | f     | f    | 2f 00 00 00 00 00 00
 00 | f    | (4,17) | {"(4,17)","(4,21)"}
(13 rows)
```

##########################

F.23.4. BRIN##

brin_page_type(page bytea) returns text

brin_page_type####BRIN#################### #######BRIN#################
#########

```
test=# SELECT brin_page_type(get_raw_page('brinidx', 0));
 brin_page_type
----------------
 meta
```

brin_metapage_info(page bytea) returns record

brin_metapage_info#BRIN########################## ##########

```
test=# SELECT * FROM brin_metapage_info(get_raw_page('brinidx', 0));
   magic    | version | pagesperrange | lastrevmappage
------------+---------+---------------+----------------
 0xA8109CFA |       1 |             4 |              2
```

brin_revmap_data(page bytea) returns setof tid

brin_revmap_data#BRIN################################ ##########

```
test=# SELECT * FROM brin_revmap_data(get_raw_page('brinidx', 2)) LIMIT
 5;
  pages
---------
 (6,137)
 (6,138)
 (6,139)
 (6,140)
 (6,141)
```

brin_page_items(page bytea, index oid) returns setof record

brin_page_items#BRIN###################### ##########

```
test=# SELECT * FROM brin_page_items(get_raw_page('brinidx', 5),
                                     'brinidx')
       ORDER BY blknum, attnum LIMIT 6;
 itemoffset | blknum | attnum | allnulls | hasnulls | placeholder |
 empty |    value
------------+--------+--------+----------+----------+-------------
+-------+--------------
        137 |      0 |      1 | t        | f        | f           | f
   |
        137 |      0 |      2 | f        | f        | f           | f
   | {1 .. 88}
        138 |      4 |      1 | t        | f        | f           | f
   |
        138 |      4 |      2 | f        | f        | f           | f
   | {89 .. 176}
        139 |      8 |      1 | t        | f        | f           | f
   |
        139 |      8 |      2 | f        | f        | f           | f
   | {177 .. 264}
```

######BrinMemTuple######BrinValues############### ####src/include/access/brin_tuple.h########

F.23.5. GIN##

gin_metapage_info(page bytea) returns record

gin_metapage_info#GIN####################### ##########

```
test=# SELECT * FROM gin_metapage_info(get_raw_page('gin_index', 0));
-[ RECORD 1 ]----+-----------
pending_head     | 4294967295
pending_tail     | 4294967295
tail_free_size   | 0
n_pending_pages  | 0
n_pending_tuples | 0
n_total_pages    | 7
n_entry_pages    | 6
n_data_pages     | 0
n_entries        | 693
version          | 2
```

gin_page_opaque_info(page bytea) returns record

gin_page_opaque_info##########GIN########################## ##########

```
test=# SELECT * FROM gin_page_opaque_info(get_raw_page('gin_index',
 2));
 rightlink | maxoff |         flags
-----------+--------+------------------------
         5 |      0 | {data,leaf,compressed}
(1 row)
```

gin_leafpage_items(page bytea) returns setof record

#####[83.333333]#gin_leafpage_items#GIN############################## ### ####### ######gin_leafpage_items#######GIN########################## #### ###########

```
test=# SELECT first_tid, nbytes, tids[0:5] AS some_tids
        FROM gin_leafpage_items(get_raw_page('gin_test_idx', 2));
 first_tid | nbytes |                        some_tids
-----------+--------
+-------------------------------------------------------------
 (8,41)    |    244 | {"(8,41)","(8,43)","(8,44)","(8,45)","(8,46)"}
 (10,45)   |    248 |
 {"(10,45)","(10,46)","(10,47)","(10,48)","(10,49)"}
 (12,52)   |    248 |
 {"(12,52)","(12,53)","(12,54)","(12,55)","(12,56)"}
 (14,59)   |    320 |
 {"(14,59)","(14,60)","(14,61)","(14,62)","(14,63)"}
 (167,16)  |    376 |
 {"(167,16)","(167,17)","(167,18)","(167,19)","(167,20)"}
 (170,30)  |    376 |
 {"(170,30)","(170,31)","(170,32)","(170,33)","(170,34)"}
 (173,44)  |    197 |
 {"(173,44)","(173,45)","(173,46)","(173,47)","(173,48)"}
(7 rows)
```

F.23.6. GiST##

gist_page_opaque_info(page bytea) returns record

gist_page_opaque_info#NSN#rightlink#########GiST######################## ## ##########

```
test=# SELECT * FROM
 gist_page_opaque_info(get_raw_page('test_gist_idx', 2));
 lsn | nsn | rightlink | flags
```

```
-----+-----+-----------+--------
 0/1 | 0/0 |         1 | {leaf}
(1 row)
```

gist_page_items(page bytea, index_oid regclass) returns setof record

gist_page_items#GiST################################ ##########

```
test=# SELECT * FROM gist_page_items(get_raw_page('test_gist_idx', 0),
 'test_gist_idx');
 itemoffset |   ctid    | itemlen | dead |             keys
------------+-----------+---------+------
+-------------------------------
          1 | (1,65535) |      40 | f    | (p)=("(185,185),(1,1)")
          2 | (2,65535) |      40 | f    | (p)=("(370,370),(186,186)")
          3 | (3,65535) |      40 | f    | (p)=("(555,555),(371,371)")
          4 | (4,65535) |      40 | f    | (p)=("(740,740),(556,556)")
          5 | (5,65535) |      40 | f    | (p)=("(870,870),(741,741)")
          6 | (6,65535) |      40 | f    | (p)=("(1000,1000),
(871,871)")
(6 rows)
```

gist_page_items_bytea(page bytea) returns setof record

gist_page_items#############byteablob######## ########################
################## ##########

```
test=# SELECT * FROM
 gist_page_items_bytea(get_raw_page('test_gist_idx', 0));
 itemoffset |   ctid    | itemlen | dead |
         key_data
------------+-----------+---------+------
+-----------------------------------------
-------------------------------------------
          1 | (1,65535) |      40 | f    |
 \x00000100ffff28000000000000c0644000000000
00c06440000000000000f03f000000000000f03f
          2 | (2,65535) |      40 | f    |
 \x00000200ffff28000000000000c0744000000000
00c07440000000000000e06440000000000000e06440
          3 | (3,65535) |      40 | f    |
 \x00000300ffff28000000000000207f4000000000
00207f400000000000d07440000000000000d07440
          4 | (4,65535) |      40 | f    |
 \x00000400ffff28000000000000c0844000000000
00c08440000000000000307f40000000000000307f40
          5 | (5,65535) |      40 | f    |
 \x00000500ffff28000000000000f0894000000000
00f08940000000000000c88440000000000000c88440
          6 | (6,65535) |      40 | f    |
 \x00000600ffff28000000000000208f4000000000
00208f400000000000f88940000000000000f88940
          7 | (7,65535) |      40 | f    |
 \x00000700ffff28000000000000408f4000000000
00408f40000000000000288f40000000000000288f40
(7 rows)
```

F.23.7. Hash##

hash_page_type(page bytea) returns text

hash_page_type######HASH################### ##########

```
test=# SELECT hash_page_type(get_raw_page('con_hash_index', 0));
 hash_page_type
----------------
 metapage
```

hash_page_stats(page bytea) returns setof record

hash_page_stats#HASH################################## ##########

```
test=# SELECT * FROM hash_page_stats(get_raw_page('con_hash_index',
 1));
-[ RECORD 1 ]---+-----------
live_items      | 407
dead_items      | 0
page_size       | 8192
free_size       | 8
hasho_prevblkno | 4096
hasho_nextblkno | 8474
hasho_bucket    | 0
hasho_flag      | 66
hasho_page_id   | 65408
```

hash_page_items(page bytea) returns setof record

hash_page_items#HASH## ### #######

```
test=# SELECT * FROM hash_page_items(get_raw_page('con_hash_index', 1))
 LIMIT 5;
 itemoffset |   ctid    |    data
------------+-----------+------------
          1 | (899,77)  | 1053474816
          2 | (897,29)  | 1053474816
          3 | (894,207) | 1053474816
          4 | (892,159) | 1053474816
          5 | (890,111) | 1053474816
```

hash_bitmap_info(index oid, blkno bigint) returns record

hash_bitmap_info#HASH## #### ######

```
test=# SELECT * FROM hash_bitmap_info('con_hash_index', 2052);
 bitmapblkno | bitmapbit | bitstatus
-------------+-----------+-----------
          65 |         3 | t
```

hash_metapage_info(page bytea) returns record

hash_metapage_info#HASH########################## ##########

```
test=# SELECT magic, version, ntuples, ffactor, bsize, bmsize, bmshift,
test-#     maxbucket, highmask, lowmask, ovflpoint, firstfree, nmaps,
 procid,
test-#     regexp_replace(spares::text, '(,0)*}', '}') as spares,
test-#     regexp_replace(mapp::text, '(,0)*}', '}') as mapp
test-# FROM hash_metapage_info(get_raw_page('con_hash_index', 0));
-[ RECORD 1 ]-------------------------------------------------
------------------------------
magic     | 105121344
```

```
version   | 4
ntuples   | 500500
ffactor   | 40
bsize     | 8152
bmsize    | 4096
bmshift   | 15
maxbucket | 12512
highmask  | 16383
lowmask   | 8191
ovflpoint | 28
firstfree | 1204
nmaps     | 1
procid    | 450
spares    | {0,0,0,0,0,0,1,1,1,1,1,1,1,1,3,4,4,4,45,55,58,59,
508,567,628,704,1193,1202,1204}
mapp      | {65}
```

F.24. passwordcheck —

passwordcheck######CREATE ROLE###ALTER ROLE###################### ###### ####################################

###############postgresql.conf##shared_preload_libraries#'$libdir/passwordcheck'##################

################################## #############CrackLib[2]####### ####### #Makefile## 2############################ ##########CrackLib############### ##### CrackLib#################################### ######################### ###

##

##PostgreSQL######################## ###################################

##########################passwordcheck################# ########### #############passwordcheck######## #########################GSSAPI# #######################20#####

#################################passwordcheck####### ################# #####################################

[2] https://github.com/cracklib/cracklib

F.25. pg_buffercache — PostgreSQL#####

#####[62.085308]#pg_buffercache##
######pg_buffercache## #######
#######################

#####[77.607362]#########pg_buffercache_pages()##(pg_buffercache########
####)#pg_buffercache_summary()######pg_buffercache_usage_counts()#####
##############pg_buffercache_pages()###pg_buffercache######
###pg_buffercache_summary()###pg_buffercache_usage_counts()#
##pg_buffercache_evict()########

pg_buffercache_pages()#######1##############################
pg_buffercache###########################

pg_buffercache_summary()######################1#######

pg_buffercache_usage_counts()##

#####[85.427136]###################pg_monitor#################### GRANT#####
############## ##############################pg_monitor####################
##########GRANT######################

######pg_buffercache_evict()## ###
#######################

F.25.1. pg_buffercache###

################### F.14######

#F.14 pg_buffercache##

##	##
bufferid integer	1##shared_buffers#########ID
relfilenode oid #### pg_class.relfilenode#	###############
reltablespace oid #### pg_tablespace.oid#	############OID
reldatabase oid #### pg_database.oid#	############OID
relforknumber smallint	##############common/relpath.h##
relblocknumber bigint	############
isdirty boolean	##########
usagecount smallint	Clock-sweep#######
pinning_backends integer	

##
######################

##################1######## #########bufferid#############NULL###### ####### #####OID#########################

##############
#pg_class#################################### pg_class#################### ####OID###0####reldatabase#####################

###pg_buffercache########## ## ############################ ###

F.25.2. pg_buffercache_summary()##

################## F.15######

#F.15 pg_buffercache_summary()###

##
buffers_used int4 ###########
buffers_unused int4 ###########
buffers_dirty int4 ###########
buffers_pinned int4 ############
usagecount_avg float8 ##################

pg_buffercache_summary()############################## ##########
#pg_buffercache##############pg_buffercache_summary()########

pg_buffercache#######pg_buffercache_summary()#################### ### ####################################

F.25.3. The pg_buffercache_usage_counts() Function

################## F.16######

#F.16 pg_buffercache_usage_counts()###

##
usage_count int4 ###########
buffers int4 ##############
dirty int4

##	##
	##################
pinned	int4 ###################

pg_buffercache_usage_counts()##
########### ###########pg_buffercache############
##pg_buffercache_usage_counts()########

pg_buffercache#######pg_buffercache_usage_counts()##################
####################################

F.25.4. The pg_buffercache_evict Function

######pg_buffercache_evict()####pg_buffercache####bufferid############
true ################ false
########### false ###### ### #
#######################

F.25.5.

```
regression=# SELECT n.nspname, c.relname, count(*) AS buffers
             FROM pg_buffercache b JOIN pg_class c
             ON b.relfilenode = pg_relation_filenode(c.oid) AND
                b.reldatabase IN (0, (SELECT oid FROM pg_database
                                      WHERE datname = current_database()))
             JOIN pg_namespace n ON n.oid = c.relnamespace
             GROUP BY n.nspname, c.relname
             ORDER BY 3 DESC
             LIMIT 10;

  nspname   |        relname         | buffers
------------+------------------------+---------
 public     | delete_test_table      |     593
 public     | delete_test_table_pkey |     494
 pg_catalog | pg_attribute           |     472
 public     | quad_poly_tbl          |     353
 public     | tenk2                  |     349
 public     | tenk1                  |     349
 public     | gin_test_idx           |     306
 pg_catalog | pg_largeobject         |     206
 public     | gin_test_tbl           |     188
 public     | spgist_text_tbl        |     182
(10 rows)


regression=# SELECT * FROM pg_buffercache_summary();
 buffers_used | buffers_unused | buffers_dirty | buffers_pinned |
 usagecount_avg
--------------+----------------+---------------+----------------
+----------------
          248 |        2096904 |            39 |              0 |
 3.141129
(1 row)


regression=# SELECT * FROM pg_buffercache_usage_counts();
 usage_count | buffers | dirty | pinned
-------------+---------+-------+--------
```

```
                0 |   14650 |     0 |      0
                1 |    1436 |   671 |      0
                2 |     102 |    88 |      0
                3 |      23 |    21 |      0
                4 |       9 |     7 |      0
                5 |     164 |   106 |      0
(6 rows)
```

F.25.6.

Mark Kirkwood <markir@paradise.net.nz>

####: Neil Conway <neilc@samurai.com>

##########: Tom Lane <tgl@sss.pgh.pa.us>

F.26. pgcrypto —

pgcrypto#####PostgreSQL############

#########trusted#########################CREATE#######################

pgcrypto#OpenSSL######PostgreSQL#####OpenSSL###########################

F.26.1.

F.26.1.1. digest()

```
digest(data text, type text) returns bytea
digest(data bytea, type text) returns bytea
```

#####*data*############### *type*############# #######
#md5#sha1#sha224#sha256#sha384####sha512### ####OpenSSL#################
###########

#######16#####################encode()######### #########

```
CREATE OR REPLACE FUNCTION sha1(bytea) returns text AS $$
    SELECT encode(digest($1, 'sha1'), 'hex')
$$ LANGUAGE SQL STRICT IMMUTABLE;
```

F.26.1.2. hmac()

```
hmac(data text, key text, type text) returns bytea
hmac(data bytea, key bytea, type text) returns bytea
```

key#####*data*######MAC###### *type*#digest()########

digest()################################ ##################################
########

##

F.26.2.

crypt()###gen_salt()############################ crypt()########
##gen_salt()###########################

crypt()###############MD5#SHA1######################

1. ##### ####################################
2. ###################### ################################## ################
   ######
3. ################### ################################
4. ######## ##

crypt()################ F.17#######

#F.17 crypt()#############

######	########	#######	#######	###	##
bf	72	##	128	60	Blowfish## ##2a#
md5	###	###	48	34	MD5######
xdes	8	##	24	20	##DES
des	8	###	12	13	###UNIX crypt

F.26.2.1. crypt()

```
crypt(password text, salt text) returns text
```

password#crypt(3)############# ################gen_salt()#######*salt*#######
####################*salt*##############################

####################

```
UPDATE ... SET pswhash = crypt('new password', gen_salt('md5'));
```

#######

```
SELECT (pswhash = crypt('entered password', pswhash)) AS pswmatch
 FROM ... ;
```

############true######

F.26.2.2. gen_salt()

```
gen_salt(type text [, iter_count integer ]) returns text
```

crypt()######################### ###########crypt()####################

type######################## ########des#xdes#md5#bf###

#####################*iter_count*####################### ####################
################################ ##
iter_count################################ *iter_count*####
################# F.18########

#F.18 crypt()########

######	#####	##	##
xdes	725	1	16777215
bf	6	4	31

xdes################################

####################DES############1####4################################ ##
4######################## ##100################################

########################## F.19######## #####8################################
crypt-bf#########
######gen_salt#*iter_count*###

#F.19

######	1#########	[a-z]###	[A-Za-z0-9]####	md5############
crypt-bf/8	1792	4#	3927#	100k
crypt-bf/7	3648	2#	1929#	50k
crypt-bf/6	7168	1#	982#	25k
crypt-bf/5	13504	188#	521#	12.5k
crypt-md5	171584	15#	41#	1k
crypt-des	23221568	157.5#	108#	7
sha1	37774272	90#	68#	4
md5(####)	150085504	22.5#	17#	1

###

- Intel Mobile Core i3###########
- crypt-des###crypt-md5#########John the Ripper v1.6.38#-test##########
- md5########mdcrack 1.2######
- sha1####lcrack-20031130-beta######
- crypt-bf####1000##8################################ ######################## ######john -test#crypt-bf/5#13506 loops/sec#### ################pgcrypto# ###crypt-bf###John the Ripper##########################

################################## ###################################### ###### ################################### ##6## ####

F.26.3. PGP#####

########OpenPGP#(RFC 4880[3]################# #####################

######PGP########2#####################

- ############## ###################
- ##########################

####################

1. ##########String2Key(S2K)################ ##########crypt()############### ################### ###################
2. ##################################### ######S2K#######################
3. S2K###############S2K######################## ##############S2K############# ###############

###########

1. ######################
2. ####################################

[3] https://datatracker.ietf.org/doc/html/rfc4880

##########################

1. ################UTF-8################
2. ######################### ###random IV#############
3. ################SHA1##############
4. ################################

F.26.3.1. pgp_sym_encrypt()

```
pgp_sym_encrypt(data text, psw text [, options text ]) returns bytea
pgp_sym_encrypt_bytea(data bytea, psw text [, options text ]) returns
 bytea
```

##PGP##*psw*#*data*######## *options*###########################

F.26.3.2. pgp_sym_decrypt()

```
pgp_sym_decrypt(msg bytea, psw text [, options text ]) returns text
pgp_sym_decrypt_bytea(msg bytea, psw text [, options text ]) returns bytea
```

##########PGP###########

pgp_sym_decrypt#bytea################## ######################## ##########
#pgp_sym_decrypt_bytea################

options###########################

F.26.3.3. pgp_pub_encrypt()

```
pgp_pub_encrypt(data text, key bytea [, options text ]) returns bytea
pgp_pub_encrypt_bytea(data bytea, key bytea [, options text ]) returns
 bytea
```

##PGP##*key*#*data*######## ######################

options###########################

F.26.3.4. pgp_pub_decrypt()

```
pgp_pub_decrypt(msg bytea, key bytea [, psw text [, options text ]])
 returns text
pgp_pub_decrypt_bytea(msg bytea, key bytea [, psw text [, options text ]])
 returns bytea
```

###################### *key*############################### ######################
#######*psw*############# ##

pgp_pub_decrypt#bytea################## ######################## ##########
#pgp_pub_decrypt_bytea################

options###########################

F.26.3.5. pgp_key_id()

```
pgp_key_id(bytea) returns text
```

pgp_key_id#PGP###########ID######## ##################################ID#######

2#######ID##########

- SYMKEY

  ##################

- ANYKEY

  ####################ID########## ## ####### pgcrypto##################

########ID################### ################## ##############################
######ANYKEY##############################

F.26.3.6. armor(), dearmor()

```
armor(data bytea [ , keys text[], values text[] ]) returns text
dearmor(data text) returns bytea
```

PGP#ASCII########################### ASCII#########CRC###BASE64#############
#########

keys#values## #######1###########
####### keys#values##ASCII##############

F.26.3.7. pgp_armor_headers

```
pgp_armor_headers(data text, key out text, value out text) returns setof
 record
```

pgp_armor_headers()#*data*################ ########2############## ##########
############UTF-8########

F.26.3.8. PGP#########

######GnuPG########### ########################## ##################### #####
#####

```
pgp_sym_encrypt(data, psw, 'compress-algo=1, cipher-algo=aes256')
```

convert-crlf########################## #####PGP####################

#################compress-algo#unicode-mode##### #################

F.26.3.8.1. cipher-algo

############

#: bf, aes128, aes192, aes256, 3des, cast5
#####: aes128
####: pgp_sym_encrypt, pgp_pub_encrypt

F.26.3.8.2. compress-algo

############# PostgreSQL#zlib####################

#:
 0 - ###
 1 - ZIP##
 2 - ZLIB## (ZIP###########CRC######)
#####: 0
####: pgp_sym_encrypt, pgp_pub_encrypt

F.26.3.8.3. compress-level

############ ######################### 0###########

#: 0, 1-9
#####: 6
####: pgp_sym_encrypt, pgp_pub_encrypt

F.26.3.8.4. convert-crlf

######\n#\r\n#######\r\n #\n ######### RFC 4880###################\r\n ####### ################ ###RFC#########################

#: 0, 1
#####: 0
####: pgp_sym_encrypt, pgp_pub_encrypt, pgp_sym_decrypt, pgp_pub_decrypt

F.26.3.8.5. disable-mdc

####SHA-1######## ########################SHA-1###########RFC 4880########### PGP############### ###gnupg.org###pgp.com#########################

#: 0, 1
#####: 0
####: pgp_sym_encrypt, pgp_pub_encrypt

F.26.3.8.6. sess-key

################## ########################### ############################S2K#### #########

#: 0, 1
#####: 0
####: pgp_sym_encrypt

F.26.3.8.7. s2k-mode

####S2K#######

#:
  0 - #########!
  1 - ##############
  3 - #########
#####: 3
####: pgp_sym_encrypt

F.26.3.8.8. s2k-count

####S2K################# 1024###65011712###############

#####: 65536##253952######
####: s2k-mode=3#######pgp_sym_encrypt

F.26.3.8.9. s2k-digest-algo

S2K####################

#: md5, sha1
#####: sha1
####: pgp_sym_encrypt

F.26.3.8.10. s2k-cipher-algo

#######################

#: bf, aes, aes128, aes192, aes256
#####: cipher-algo###
####: pgp_sym_encrypt

F.26.3.8.11. unicode-mode

#######################UTF-8############## ##########UTF-8################# UTF-8############# ####################

#: 0, 1
#####: 0
####: pgp_sym_encrypt, pgp_pub_encrypt

F.26.3.9. GnuPG##########

############

```
gpg --gen-key
```

##########DSA#Elgamal####

RSA##############DSA###RSA################`gpg --edit-key`#####RSA######## ###############

#########

```
gpg --list-secret-keys
```

ASCII####################

```
gpg -a --export KEYID > public.key
```

ASCII####################

```
gpg -a --export-secret-keys KEYID > secret.key
```

PGP##############`dearmor()`############# ########################`-a`########## #####

###`man gpg`#The GNU Privacy Handbook[4]#https://www.gnupg.org/##################

F.26.3.10. PGP######

- ################ ##
- ###################### ###################################
- #################### ######################### ######GPG/PGP###`pgcrypto`##### ######### ##########################

F.26.4.

############################ PGP################## ######################

1. ######################
2. ##
3. #####IV################################
4. ##########

#####PGP##############################

```
encrypt(data bytea, key bytea, type text) returns bytea
decrypt(data bytea, key bytea, type text) returns bytea

encrypt_iv(data bytea, key bytea, iv bytea, type text) returns bytea
decrypt_iv(data bytea, key bytea, iv bytea, type text) returns bytea
```

`type`############################## *`type`*###############

```
algorithm [ - mode ] [ /pad: padding ]
```

###*`algorithm`*####1####

- `bf` — Blowfish
- `aes` — AES (Rijndael-128, -192 or -256)

[4] https://www.gnupg.org/gph/en/manual.html

##mode####1####

- cbc — #######################
- ecb — #######################

padding####1####

- pkcs — ##################
- none — ########################

###############

```
encrypt(data, 'fooz', 'bf')
encrypt(data, 'fooz', 'bf-cbc/pad:pkcs')
```

encrypt_iv###decrypt_iv###*iv*######CBC############## ECB######### ##########
######################### ##############################

F.26.5.

```
gen_random_bytes(count integer) returns bytea
```

###########*count*######### ######1024############### ########################
#######

```
gen_random_uuid() returns uuid
```

#####4(#####)UUID###### (##############################)

F.26.6.

F.26.6.1.

pgcrypto#####PostgreSQL#configure################### #############--with-zlib#--with-openssl###

ZLIB##############PGP#########################

pgcrypto#OpenSSL######## #######################

OpenSSL 3.0.0###################DES#Blowfish###############openssl.cnf########
#######################

F.26.6.2. NULL###

##SQL###########NULL##########NULL##### ################################
##

F.26.6.3.

pgcrypto######################### ####pgcrypto##############################
############### ##################

1. #######SSL#####

2. ####################

#######################################

###########[5]######### ####pgcrypto#######################################

F.26.7.

Marko Kreen <markokr@gmail.com>

pgcrypto###############

######	##	####
DES crypt	David Burren#	FreeBSD libcrypt
MD5 crypt	Poul-Henning Kamp	FreeBSD libcrypt
Blowfish crypt	Solar Designer	www.openwall.com

[5] https://en.wikipedia.org/wiki/Side-channel_attack

F.27. pg_freespacemap —

pg_freespacemap##############FSM############## pg_freespace############ ################### ## #####

##################pg_stat_scan_tables################### GRANT######### #########

F.27.1.

pg_freespace(rel regclass IN, blkno bigint IN) returns int2

FSM######blkno##############################

pg_freespace(rel regclass IN, blkno OUT bigint, avail OUT int2)

FSM############################## ##############(blkno bigint, avail int2)#1#######################

######################## ######BLCKSZ#1/256(#####BLCKSZ##32###)############## ####################################

################################

F.27.2.

```
postgres=# SELECT * FROM pg_freespace('foo');
 blkno | avail
-------+-------
     0 |     0
     1 |     0
     2 |     0
     3 |    32
     4 |   704
     5 |   704
     6 |   704
     7 |  1216
     8 |   704
     9 |   704
    10 |   704
    11 |   704
    12 |   704
    13 |   704
    14 |   704
    15 |   704
    16 |   704
    17 |   704
    18 |   704
    19 |  3648
(20 rows)

postgres=# SELECT * FROM pg_freespace('foo', 7);
 pg_freespace
--------------
         1216
(1 row)
```

F.27.3.

########## Mark Kirkwood <markir@paradise.net.nz>######## #####8.4###Heikki Linnakangas <heikki@enterprisedb.com>#######FSM################

F.28. pg_prewarm — #########################

pg_prewarm#########################PostgreSQL################################ ############ #######pg_prewarm################## ##pg_prewarm#shared_preload_libraries################## ################ #autoprewarm.blocks##2######### #####################

F.28.1.

```
pg_prewarm(regclass, mode text default 'buffer', fork text default 'main',
           first_block int8 default null,
           last_block int8 default null) RETURNS int8
```

###########(########)################ ## ################################main### #########################(NULL####### ###)# ################(NULL ##########################)# ####################

############################## prefetch## ### read################### ##prefetch## buffer########### #####################

###prefetch#read####OS#### ############ PostgreSQL#buffer## ## ## ## ################################## ###########

```
autoprewarm_start_worker() RETURNS void
```

################## ## #####

```
autoprewarm_dump_now() RETURNS int8
```

###autoprewarm.blocks####### ## ################### ####autoprewarm.blocks##############

F.28.2.

pg_prewarm.autoprewarm(boolean)

   ################################## ######on### #####################

pg_prewarm.autoprewarm_interval(integer)

   ###autoprewarm.blocks######### ######300#### 0######################### ###################

##########postgresql.conf############# ###############

```
# postgresql.conf
shared_preload_libraries = 'pg_prewarm'
```

```
pg_prewarm.autoprewarm = true
pg_prewarm.autoprewarm_interval = 300s
```

F.28.3.

Robert Haas <rhaas@postgresql.org>

F.29. pgrowlocks —

pgrowlocks####################################

##################pg_stat_scan_tables###############SELECT#############
####

F.29.1.

pgrowlocks(text) returns setof record

############### ##############################1####### ##### F.20######

#F.20 pgrowlocks####

##	#	##
locked_row	tid	###########ID#TID#
locker	xid	########################ID#######################ID#66.1###
multi	boolean	#########################
xids	xid[]	######################### ###ID##################
modes	text[]	Lock mode of lockers (more than one if multitransaction), an array of For Key Share, For Share, For No Key Update, No Key Update, For Update, Update.
pids	integer[]	####################ID## ################

pgrowlocks###########AccessShareLock################1########### ######### ############### ###########:

1. #####ACCESS EXCLUSIVE#############pgrowlocks#########
2. pgrowlocks################################ ################################ #############

pgrowlocks################### ####################################:

```
SELECT * FROM accounts AS a, pgrowlocks('accounts') AS p
  WHERE p.locked_row = a.ctid;
```

##################################

F.29.2.

```
=# SELECT * FROM pgrowlocks('t1');
 locked_row | locker | multi | xids  |     modes      |  pids
------------+--------+-------+-------+----------------+--------
 (0,1)      |    609 | f     | {609} | {"For Share"}  | {3161}
```

```
 (0,2)      |    609 | f     | {609} | {"For Share"}  | {3161}
 (0,3)      |    607 | f     | {607} | {"For Update"} | {3107}
 (0,4)      |    607 | f     | {607} | {"For Update"} | {3107}
(4 rows)
```

F.29.3.

Tatsuo Ishii

F.30. pg_stat_statements — SQL###########

pg_stat_statements##################SQL#############################

#####################
##postgresql.conf#shared_preload_libraries#pg_stat_statements############### ####### ### ############### #########ID##################compute_query_id######auto#on##############ID## ####################################

pg_stat_statements################################### ################### ###############pg_stat_statements#pg_stat_statements_info######## #pg_stat_statements_reset#pg_stat_statements###### ############## ##CREATE EXTENSION pg_stat_statements################

F.30.1. pg_stat_statements

####################pg_stat_statements#################### #######1###### ############ID####ID######ID#######################################SQL###### ### ######## F.21########

#F.21 pg_stat_statements##

# #	##
userid oid #### pg_authid.oid#	SQL##########OID
dbid oid #### pg_database.oid#	SQL##############OID
toplevel bool	############SQL########### #pg_stat_statements.track#top###### #######
queryid bigint	###########################
query text	####SQL#####
plans bigint	SQL########### #pg_stat_statements.track_planning##############
total_plan_time double precision	SQL#################### #pg_stat_statements.track_planning#### ###########
min_plan_time double precision	################### ######## pg_stat_statements.track_planning### #######minmax_only######true##################################
max_plan_time double precision	################### ######## pg_stat_statements.track_planning### #######minmax_only######true##################################
mean_plan_time double precision	SQL###################### #pg_stat_statements.track_planning### ############
stddev_plan_time double precision	

##
SQL########################## #pg_stat_statements.track_planning###############
calls bigint SQL########
total_exec_time double precision SQL###################
min_exec_time double precision ######################## #########pg_stat_statements_reset## #minmax_only######true################################
max_exec_time double precision ############################## #########pg_stat_statements_reset## #minmax_only######true##############################
mean_exec_time double precision SQL####################
stddev_exec_time double precision SQL########################
rows bigint SQL########################
shared_blks_hit bigint SQL########################
shared_blks_read bigint SQL####################
shared_blks_dirtied bigint SQL########################
shared_blks_written bigint SQL####################
local_blks_hit bigint SQL##########################
local_blks_read bigint SQL########################
local_blks_dirtied bigint SQL##########################
local_blks_written bigint SQL########################
temp_blks_read bigint SQL####################
temp_blks_written bigint SQL####################
shared_blk_read_time double precision #####[84.172662]#SQL################################## #track_io_timing#### ############ ########################## ####### #track_io_timing########### ########
shared_blk_write_time double precision #####[84.172662]#SQL################################## #track_io_timing#### ############ ########################## ####### #track_io_timing########### ########
local_blk_read_time double precision

##	##
	#####[84.782609]#SQL################################ #track_io_timing#### ########### ################################### ####### #track_io_timing ##################
local_blk_write_time double precision	#####[84.782609]#SQL################################ #track_io_timing#### ########### ########################### ####### #track_io_timing######### #########
temp_blk_read_time double precision	SQL############################### #track_io_timing###############
temp_blk_write_time double precision	SQL############################### #track_io_timing###############
wal_records bigint	SQL########WAL######
wal_fpi bigint	SQL########WAL###########
wal_bytes numeric	SQL##############WAL##
jit_functions bigint	SQL##JIT############
jit_generation_time double precision	SQL##JIT###################
jit_inlining_count bigint	#############
jit_inlining_time double precision	SQL#########################
jit_optimization_count bigint	SQL#########
jit_optimization_time double precision	SQL###################
jit_emission_count bigint	##########
jit_emission_time double precision	SQL#########################
jit_deform_count bigint	#####[70.588235]#SQL##JIT############ ###########JIT##################
jit_deform_time double precision	#####[67.777778]#SQL######################### ##########JIT########### #######
stats_since timestamp with time zone	############################
minmax_stats_since timestamp with time zone	##############/################### #min_plan_time#max_plan_time#min_exec_time#max_exec_time##

####################pg_read_all_stats############################SQL##### ######queryid########## ## #####

################SELECT#INSERT#UPDATE#DELETE# MERGE#######################
#######################pg_stat_statements############ ####################
##

> ##
>
> #####queryid##########compute_query_id############## queryid##########
> ####################################

##########################pg_stat_statements########$1##################
##############pg_stat_statements##########queryid######################
##

###############pg_stat_statements################ #######################
##################pg_stat_statements.max###############
pg_stat_statements_info#######F.30.2##############################

##############################pg_stat_statements################ #########
################
##############################

queryid##############################search_path##########################
#####################################

pg_stat_statements##dbid#userid##
######queryid############ ####queryid################################## #
##
##pg_stat_statements####################2##################################
################ ## ####PostgreSQL#######
########queryid####################

##WAL##################2#####################queryid############# ########
###queryid##
############# ####################

####queryid##PostgreSQL##########
################## ####################################

###$*n*####################
#$1######## ## ####PL/pgSQL#
##SELECT i + 1 INTO j####PL/pgSQL#
#SELECT i + $2################

#
####################
##pg_stat_statements##############################queryid##############
##pg_stat_statements################query############# ##################
###pg_stat_statements.max###############

###plans#calls############
####SQL##
##############################

F.30.2. pg_stat_statements_info###

pg_stat_statements###############pg_stat_statements_info##############
####### #######1######### ######## F.22########

#F.22 **pg_stat_statements_info##**

##
dealloc bigint pg_stat_statements.max########SQL##################SQL# #pg_stat_statements################
stats_reset timestamp with time zone pg_stat_statements#######################

F.30.3.

pg_stat_statements_reset(userid Oid, dbid Oid, queryid bigint, minmax_only boolean) returns timestamp with time zone

> ######pg_stat_statements_reset##userid#dbid#queryid####
> #pg_stat_statements################## ##############################
> #0################################ ################################0####
> ################## pg_stat_statements################
> ##pg_stat_statements_info################ minmax_only#true#########
> ######################
> #min_plan_time#max_plan_time#min_exec_time#max_exec_time######
> minmax_only###########false### #########/########
> ##pg_stat_statements####minmax_stats_since########### ########## ####
> ####################pg_stat_statements_info####stats_reset#######
> #pg_stat_statements####minmax_stats_since############ ##############
> ############## #####GRANT####################

pg_stat_statements(showtext boolean) returns setof record

> pg_stat_statements######pg_stat_statements################## ####
> ###pg_stat_statements#########showtext := false#####
> ################################query#####OUT###NULL###### #########
> ### ###############
> #pg_stat_statements##
> ################### ##############################pg_stat_statements###
> #############I/O########

F.30.4.

pg_stat_statements.max (integer)

> pg_stat_statements.max#################SQL########
> ##pg_stat_statements############## ########SQL##################SQL
> ########### ################pg_stat_statements_info########## ######
> 5000### ######################

pg_stat_statements.track (enum)

> pg_stat_statements.track####SQL##################### top############
> #################SQL####### all################################ none####
> ############### ######top### ####################

pg_stat_statements.track_utility (boolean)

> pg_stat_statements.track_utility##################################
> ############# SELECT#INSERT#UPDATE#DELETE####MERGE######## ######on###
> ####################

pg_stat_statements.track_planning(boolean)

pg_stat_statements.track_planning################################## ############################SQL#####pg_stat_statements############## ## #####off### ################## ##

pg_stat_statements.save(boolean)

pg_stat_statements.save################################ off######### ########################## #######on### #######postgresql.conf####### ##################

########pg_stat_statements.max################### pg_stat_statements.track#none## #####

#########postgresql.conf############### ###################

```
# postgresql.conf
shared_preload_libraries = 'pg_stat_statements'

compute_query_id = on
pg_stat_statements.max = 10000
pg_stat_statements.track = all
```

F.30.5.

```
bench=# SELECT pg_stat_statements_reset();

$ pgbench -i bench
$ pgbench -c10 -t300 bench

bench=# \x
bench=# SELECT query, calls, total_exec_time, rows, 100.0 *
 shared_blks_hit /
               nullif(shared_blks_hit + shared_blks_read, 0) AS
 hit_percent
          FROM pg_stat_statements ORDER BY total_exec_time DESC LIMIT 5;
-[ RECORD 1 ]---+--------------------------------------------------
------------------
query           | UPDATE pgbench_branches SET bbalance = bbalance + $1
 WHERE bid = $2
calls           | 3000
total_exec_time | 25565.855387
rows            | 3000
hit_percent     | 100.0000000000000000
-[ RECORD 2 ]---+--------------------------------------------------
------------------
query           | UPDATE pgbench_tellers SET tbalance = tbalance + $1
 WHERE tid = $2
calls           | 3000
total_exec_time | 20756.669379
rows            | 3000
hit_percent     | 100.0000000000000000
-[ RECORD 3 ]---+--------------------------------------------------
------------------
query           | copy pgbench_accounts from stdin
calls           | 1
total_exec_time | 291.865911
rows            | 100000
hit_percent     | 100.0000000000000000
```

```
-[ RECORD 4 ]---+-------------------------------------------------
------------------
query           | UPDATE pgbench_accounts SET abalance = abalance + $1
 WHERE aid = $2
calls           | 3000
total_exec_time | 271.232977
rows            | 3000
hit_percent     | 98.8454011741682975
-[ RECORD 5 ]---+-------------------------------------------------
------------------
query           | alter table pgbench_accounts add primary key (aid)
calls           | 1
total_exec_time | 160.588563
rows            | 0
hit_percent     | 100.0000000000000000


bench=# SELECT pg_stat_statements_reset(0,0,s.queryid) FROM
 pg_stat_statements AS s
            WHERE s.query = 'UPDATE pgbench_branches SET bbalance =
 bbalance + $1 WHERE bid = $2';

bench=# SELECT query, calls, total_exec_time, rows, 100.0 *
 shared_blks_hit /
               nullif(shared_blks_hit + shared_blks_read, 0) AS
 hit_percent
          FROM pg_stat_statements ORDER BY total_exec_time DESC LIMIT 5;
-[ RECORD 1 ]---+-------------------------------------------------
------------------
query           | UPDATE pgbench_tellers SET tbalance = tbalance + $1
 WHERE tid = $2
calls           | 3000
total_exec_time | 20756.669379
rows            | 3000
hit_percent     | 100.0000000000000000
-[ RECORD 2 ]---+-------------------------------------------------
------------------
query           | copy pgbench_accounts from stdin
calls           | 1
total_exec_time | 291.865911
rows            | 100000
hit_percent     | 100.0000000000000000
-[ RECORD 3 ]---+-------------------------------------------------
------------------
query           | UPDATE pgbench_accounts SET abalance = abalance + $1
 WHERE aid = $2
calls           | 3000
total_exec_time | 271.232977
rows            | 3000
hit_percent     | 98.8454011741682975
-[ RECORD 4 ]---+-------------------------------------------------
------------------
query           | alter table pgbench_accounts add primary key (aid)
calls           | 1
total_exec_time | 160.588563
rows            | 0
hit_percent     | 100.0000000000000000
-[ RECORD 5 ]---+-------------------------------------------------
------------------
query           | vacuum analyze pgbench_accounts
calls           | 1
total_exec_time | 136.448116
rows            | 0
hit_percent     | 99.9201915403032721
```

```
bench=# SELECT pg_stat_statements_reset(0,0,0);

bench=# SELECT query, calls, total_exec_time, rows, 100.0 *
 shared_blks_hit /
               nullif(shared_blks_hit + shared_blks_read, 0) AS
 hit_percent
          FROM pg_stat_statements ORDER BY total_exec_time DESC LIMIT 5;
-[ RECORD 1 ]---+--------------------------------------------------
---------------------------
query           | SELECT pg_stat_statements_reset(0,0,0)
calls           | 1
total_exec_time | 0.189497
rows            | 1
hit_percent     |
-[ RECORD 2 ]---+--------------------------------------------------
---------------------------
query           | SELECT query, calls, total_exec_time, rows, $1 *
 shared_blks_hit /          +
                |                nullif(shared_blks_hit +
 shared_blks_read, $2) AS hit_percent+
                |           FROM pg_stat_statements ORDER BY
 total_exec_time DESC LIMIT $3
calls           | 0
total_exec_time | 0
rows            | 0
hit_percent     |
```

F.30.6.

<itagaki.takahiro@oss.ntt.co.jp># Peter Geoghegan <peter@2ndquadrant.com>####################

F.31. pgstattuple —

pgstattuple##################################

#######pg_stat_scan_tables## ####EXECUTE######### ##################### #################GRANT######### ############################## #############pg_stat_scan_tables###########

F.31.1.

pgstattuple(regclass) returns record

pgstattuple## ################### ############# ############################OID### #########

```
test=> SELECT * FROM pgstattuple('pg_catalog.pg_proc');
-[ RECORD 1 ]------+-------
table_len          | 458752
tuple_count        | 1470
tuple_len          | 438896
tuple_percent      | 95.67
dead_tuple_count   | 11
dead_tuple_len     | 3157
dead_tuple_percent | 0.69
free_space         | 8932
free_percent       | 1.95
```

F.23#######

#F.23 pgstattuple####

#	#	##
table_len	bigint	###############
tuple_count	bigint	######
tuple_len	bigint	################
tuple_percent	float8	#######
dead_tuple_count	bigint	######
dead_tuple_len	bigint	############
dead_tuple_percent	float8	#######
free_space	bigint	##########
free_percent	float8	######

##

table_len##tuple_len#dead_tuple_len#free_space############# ## ##

pgstattuple ######################## ##################################### ###############

pgstattuple##HeapTupleSatisfiesDirty##########################

pgstattuple(text) returns record

TEXT#####################pgstattuple(regclass)###### ################################

pgstatindex(regclass) returns record

pgstatindex#B-tree######################## #######

```
test=> SELECT * FROM pgstatindex('pg_cast_oid_index');
-[ RECORD 1 ]------+------
version            | 2
tree_level         | 0
index_size         | 16384
root_block_no      | 1
internal_pages     | 0
leaf_pages         | 1
empty_pages        | 0
deleted_pages      | 0
avg_leaf_density   | 54.27
leaf_fragmentation | 0
```

############

#	#	##
version	integer	B-tree#######
tree_level	integer	#############
index_size	bigint	###############
root_block_no	bigint	####################
internal_pages	bigint	###############
leaf_pages	bigint	#######
empty_pages	bigint	#####
deleted_pages	bigint	######
avg_leaf_density	float8	###########
leaf_fragmentation	float8	##########

#####index_size#####internal_pages + leaf_pages + empty_pages + deleted_pages######1############### ####index_size####################

pgstattuple################# ######################################

pgstatindex(text) returns record

TEXT#####################pgstatindex(regclass)###### ################################

pgstatginindex(regclass) returns record

pgstatginindex##GIN######################## ##########

```
test=> SELECT * FROM pgstatginindex('test_gin_index');
-[ RECORD 1 ]--+--
version        | 1
pending_pages  | 0
pending_tuples | 0
```

############

#	#	##
version	integer	GIN#######
pending_pages	integer	###########
pending_tuples	bigint	###########

pgstathashindex(regclass) returns record

pgstathashindex##HASH######################## ##########

```
test=> select * from pgstathashindex('con_hash_index');
-[ RECORD 1 ]--+-----------------
version        | 4
bucket_pages   | 33081
overflow_pages | 0
bitmap_pages   | 1
unused_pages   | 32455
live_items     | 10204006
dead_items     | 0
free_percent   | 61.8005949100872
```

###########

#	#	##
version	integer	HASH#######
bucket_pages	bigint	#########
overflow_pages	bigint	###########
bitmap_pages	bigint	###########
unused_pages	bigint	###########
live_items	bigint	########
dead_tuples	bigint	########
free_percent	float	#######

pg_relpages(regclass) returns bigint

pg_relpages##################

pg_relpages(text) returns bigint

TEXT#####################pg_relpages(regclass)###### ################## ################################

pgstattuple_approx(regclass) returns record

pgstattuple_approx#pgstattuple########################## ########### ######OID### ##########

```
test=> SELECT * FROM
 pgstattuple_approx('pg_catalog.pg_proc'::regclass);
-[ RECORD 1 ]--------+-------
table_len            | 573440
scanned_percent      | 2
approx_tuple_count   | 2740
approx_tuple_len     | 561210
approx_tuple_percent | 97.87
dead_tuple_count     | 0
dead_tuple_len       | 0
```

```
dead_tuple_percent   | 0
approx_free_space    | 11996
approx_free_percent  | 2.09
```

######### F.24#######

pgstattuple##
##pgstattuple_approx###
##

################################VM#######################################
################## ##
#########

###########
###########################VACUUM#pg_class.reltuples###############

#F.24 pgstattuple_approx####

#	#	##
table_len	bigint	##################
scanned_percent	float8	############
approx_tuple_count	bigint	#########
approx_tuple_len	bigint	################
approx_tuple_percent	float8	########
dead_tuple_count	bigint	#########
dead_tuple_len	bigint	################
dead_tuple_percent	float8	########
approx_free_space	bigint	##############
approx_free_percent	float8	#######

###############pgstattuple#################### ##########################
#########################

F.31.2.

Tatsuo Ishii#Satoshi Nagayasu#Abhijit Menon-Sen

F.32. pg_surgery —

pg_surgery#################################### ##########################
################################### ##
######UNIQUE####FOREIGN KEY##
############ ##

F.32.1.

heap_force_kill(regclass, tid[]) returns void

heap_force_kill#######################(line pointer)######(dead)######### ####
#######

```
test=> select * from t1 where ctid = '(0, 1)';
ERROR:  could not access status of transaction 4007513275
DETAIL:  Could not open file "pg_xact/0EED": No such file or directory.

test=# select heap_force_kill('t1'::regclass, ARRAY['(0, 1)']::tid[]);
 heap_force_kill
-----------------

(1 row)

test=# select * from t1 where ctid = '(0, 1)';
(0 rows)
```

heap_force_freeze(regclass, tid[]) returns void

heap_force_freeze############################ ########################
###
########### #######

```
test=> vacuum t1;
ERROR:  found xmin 507 from before relfrozenxid 515
CONTEXT:  while scanning block 0 of relation "public.t1"

test=# select ctid from t1 where xmin = 507;
 ctid
-------
 (0,3)
(1 row)

test=# select heap_force_freeze('t1'::regclass, ARRAY['(0,
 3)']::tid[]);
 heap_force_freeze
-------------------

(1 row)

test=# select ctid from t1 where xmin = 2;
 ctid
-------
 (0,3)
(1 row)
```

F.32.2.

Ashutosh Sharma <ashu.coek88@gmail.com>

F.33. pg_trgm —

pg_trgm###
#########

########trusted#########################CREATE#######################

F.33.1.

###########3#################### ### #
##

##
pg_trgm####################################### ###################### #############1#################### #####cat################# c### ca###cat###at #### #foo|bar################# f### fo###foo###oo ### b### ba###bar###ar ####

F.33.2.

pg_trgm############## F.25####### F.26######

#F.25 pg_trgm Functions

##
similarity (text, text) → real ####################### ################################1############ ##############
show_trgm (text) → text[] ############################## ##############################
word_similarity (text, text) → real ### ##### ###############
strict_word_similarity (text, text) → real word_similarity####################### ########################## ##
show_limit () → real %##################### ##2 ################## (###########SHOW pg_trgm.similarity_threshold# ########)
set_limit (real) → real %##################### ###0##1####################0.3#### ########### #### (#######; ####SET pg_trgm.similarity_threshold#########)

#########

```
# SELECT word_similarity('word', 'two words');
 word_similarity
-----------------
             0.8
(1 row)
```

```
##################{" w"," wo","wor","ord","rd "}### ####################
##{" t"," tw",two,"wo "," w"," wo","wor","ord","rds", "ds "}### #####
##############################{" w"," wo","wor","ord"}######0.8######
```

```
##################################################### ####################
######## ##############################################
```

```
####strict_word_similarity###################                        ####
#strict_word_similarity###'words'##################{"             w","
wo","wor","ord","rds","ds "}#######
```

```
# SELECT strict_word_similarity('word', 'two words'), similarity('word',
 'words');
 strict_word_similarity | similarity
------------------------+------------
               0.571429 |   0.571429
(1 row)
```

```
#####strict_word_similarity##########################word_similarity###
########################
```

#F.26 `pg_trgm` Operators

###	##
text % text → boolean	######pg_trgm.similarity_threshold####################true#### ##
text <% text → boolean	## ##pg_trgm.word_similarity_threshold######################### #true######
text %> text → boolean	<%############
text <<% text → boolean	## #pg_trgm.strict_word_similarity_threshold#################### ##########true######
text %>> text → boolean	<<%############
text <-> text → real	#############1 - similarity()########
text <<-> text → real	#############1 - word_similarity()########
text <->> text → real	<<->############
text <<<-> text → real	

##
############1 - strict_word_similarity()#######
text <->>> text → real <<<->###########

F.33.3. GUC#####

pg_trgm.similarity_threshold(real)

%###################### ###0##1##################0.3###

pg_trgm.word_similarity_threshold(real)

<%#%>###################### ###0##1##################0.6###

pg_trgm.strict_word_similarity_threshold(real)

<<%#%>>######################### ###0##1##################0.5###

F.33.4.

pg_trgm##GiST###GIN############ ###### ############################LIKE#ILIKE#~#~*###=################### ############### pg_trgm################################ ################## ## ####################B-tree################################

##

```
CREATE TABLE test_trgm (t text);
CREATE INDEX trgm_idx ON test_trgm USING GIST (t gist_trgm_ops);
```

###

```
CREATE INDEX trgm_idx ON test_trgm USING GIN (t gin_trgm_ops);
```

gist_trgm_ops GiST################################ ############siglen##### ############## ###########12###### ##########1##2024####### ################## ##########(################################)#############

######32###############

```
CREATE INDEX trgm_idx ON test_trgm USING GIST (t
 gist_trgm_ops(siglen=32));
```

##########t####################### #################

```
SELECT t, similarity(t, 'word') AS sml
  FROM test_trgm
  WHERE t % 'word'
  ORDER BY sml DESC, t;
```

####word############################## ######################################

############

```
SELECT t, t <-> 'word' AS dist
```

```
  FROM test_trgm
  ORDER BY dist LIMIT 10;
```

##GIN#########GiST########################## ##############################1 ############

#########################t################## ###############

```
SELECT t, word_similarity('word', t) AS sml
  FROM test_trgm
  WHERE 'word' <% t
  ORDER BY sml DESC, t;
```

#

```
SELECT t, strict_word_similarity('word', t) AS sml
  FROM test_trgm
  WHERE 'word' <<% t
  ORDER BY sml DESC, t;
```

####word## ######### ########################## ##

##########

```
SELECT t, 'word' <<-> t AS dist
  FROM test_trgm
  ORDER BY dist LIMIT 10;
```

#

```
SELECT t, 'word' <<<-> t AS dist
  FROM test_trgm
  ORDER BY dist LIMIT 10;
```

##GIN#########GiST#######################

PostgreSQL 9.1###############LIKE###ILIKE################### #########

```
SELECT * FROM test_trgm WHERE t LIKE '%foo%bar';
```

B-tree#################################

PostgreSQL 9.3#######################~###~*######################## #########

```
SELECT * FROM test_trgm WHERE t ~ '(foo|bar)';
```

B-tree################################

LIKE###

GiST###GIN##########GiST#GIN############### ##################

F.33.5.

#########

```
#################################

CREATE TABLE words AS SELECT word FROM
        ts_stat('SELECT to_tsvector(''simple'', bodytext) FROM
 documents');
```

```
###documents#######bodytext##################             ################
##to_tsvector###simple######################################
```

```
##word####################
```

```
CREATE INDEX words_idx ON words USING GIN (word gin_trgm_ops);
```

```
##########SELECT############################################ ###################
####################################
```

##
words## ######### ##############

F.33.6.

GiST##### http://www.sai.msu.su/~megera/postgres/gist/

Tsearch2##### http://www.sai.msu.su/~megera/postgres/gist/tsearch/V2/

F.33.7.

Oleg Bartunov <oleg@sai.msu.su>, Moscow, Moscow University, Russia

Teodor Sigaev <teodor@sigaev.ru>, Moscow, Delta-Soft Ltd.,Russia

Alexander Korotkov <a.korotkov@postgrespro.ru>, Moscow, Postgres Professional, Russia

####: Christopher Kings-Lynne

###############Delta-Soft Ltd.########

F.34. pg_visibility —

pg_visibility###########(Visibility Map, VM)##

#########################3############### ## ## ####### ## ## #######PD_ALL_VISIBLE## ##### ###2## ############### ###pg_visibility## ######################## ##

PD_ALL_VISIBLE## ## ##############################

F.34.1.

pg_visibility_map(relation regclass, blkno bigint, all_visible OUT boolean, all_frozen OUT boolean) returns record

##

pg_visibility(relation regclass, blkno bigint, all_visible OUT boolean, all_frozen OUT boolean, pd_all_visible OUT boolean) returns record

###PD_ALL_VISIBLE######

pg_visibility_map(relation regclass, blkno OUT bigint, all_visible OUT boolean, all_frozen OUT boolean) returns setof record

#######################################

pg_visibility(relation regclass, blkno OUT bigint, all_visible OUT boolean, all_frozen OUT boolean, pd_all_visible OUT boolean) returns setof record

##PD_ALL_VISIBLE#########

pg_visibility_map_summary(relation regclass, all_visible OUT bigint, all_frozen OUT bigint) returns record

##

pg_check_frozen(relation regclass, t_ctid OUT tid) returns setof tid

#####################################TID###### #######TID################## ######

pg_check_visible(relation regclass, t_ctid OUT tid) returns setof tid

##TID###### #######TID################ ########

pg_truncate_visibility_map(relation regclass) returns void

######################## ## ### ###########################VACUUM####################################### ##

```
#####################pg_stat_scan_tables##################
##pg_truncate_visibility_map(relation regclass)####################
```

F.34.2.

Robert Haas <rhaas@postgresql.org>

F.35. pg_walinspect — #####WAL##

pg_walinspect##########PostgreSQL#####################################SQL ######## ####################### pg_waldump####################SQL######## ####

############################ID#####WAL########

> **##**
>
> pg_walinspect###############WAL#############LSN############### ## ##pg_logical_emit_message######################LSN######

> **###**
>
> ##LSN###################pg_walinspect#############LSN####*end_lsn*# ################ #####*end_lsn*###################
>
> FFFFFFFF/FFFFFFFF#####pg_lsn###*end_lsn*############## ##########LSN #####*end_lsn*##############

#########################pg_read_server_files################ ####### #GRANT####################

F.35.1.

pg_get_wal_record_info(in_lsn pg_lsn) returns record

in_lsn############WAL############ #######

```
postgres=# SELECT * FROM pg_get_wal_record_info('0/E419E28');
-[ RECORD 1 ]----+-------------------------------------------------
start_lsn        | 0/E419E28
end_lsn          | 0/E419E68
prev_lsn         | 0/E419D78
xid              | 0
resource_manager | Heap2
record_type      | VACUUM
record_length    | 58
main_data_length | 2
fpi_length       | 0
description      | nunused: 5, unused: [1, 2, 3, 4, 5]
block_ref        | blkref #0: rel 1663/16385/1249 fork main blk 364
```

in_lsn#WAL################WAL##################### #####WAL####### ##############

pg_get_wal_records_info(start_lsn pg_lsn, end_lsn pg_lsn) returns setof record

start_lsn#*end_lsn*#########WAL############## WAL######1######### ##### ##

```
postgres=# SELECT * FROM pg_get_wal_records_info('0/1E913618',
 '0/1E913740') LIMIT 1;
-[ RECORD 1 ]----
+-------------------------------------------------------------
start_lsn        | 0/1E913618
end_lsn          | 0/1E913650
prev_lsn         | 0/1E9135A0
xid              | 0
resource_manager | Standby
record_type      | RUNNING_XACTS
record_length    | 50
main_data_length | 24
fpi_length       | 0
description      | nextXid 33775 latestCompletedXid 33774
 oldestRunningXid 33775
block_ref        |
```

start_lsn#######################

pg_get_wal_block_info(start_lsn pg_lsn, end_lsn pg_lsn, show_data boolean DEFAULT true) returns setof record pg_get_wal_records_info_till_end_of_wal(start_lsn pg_lsn) returns setof record

start_lsn#*end_lsn*#########WAL######1#################################
WAL#############1######

```
postgres=# SELECT * FROM pg_get_wal_block_info('0/1230278',
 '0/12302B8');
-[ RECORD 1 ]-----+-----------------------------------
start_lsn         | 0/1230278
end_lsn           | 0/12302B8
prev_lsn          | 0/122FD40
block_id          | 0
reltablespace     | 1663
reldatabase       | 1
relfilenode       | 2658
relforknumber     | 0
relblocknumber    | 11
xid               | 341
resource_manager  | Btree
record_type       | INSERT_LEAF
record_length     | 64
main_data_length  | 2
block_data_length | 16
block_fpi_length  | 0
block_fpi_info    |
description       | off: 46
block_data        | \x00002a000700104026300000706964 00
block_fpi_data    |
```

#####1############WAL###############WAL#######################
pg_get_wal_block_info##############*start_lsn*#*block_id*##############
#######

###########pg_get_wal_records_info####################### ##
##pg_get_wal_block_info###WAL#######################################
#################### ### ####
###############COMMITWAL#############pg_get_wal_block_info####
#pg_get_wal_records_info################

reltablespace#reldatabase#relfilenode#####
##pg_tablespace.oid#pg_database.oid#pg_class.oid#pg_class.relfilenode#

relforknumber######################## ###common/relpath.h######
####

> ###
>
> pg_filenode_relation#### 9.101#################################
> ##

######################
show_data#false##########block_data#block_fpi_data################
#########block_data#block_fpi_data#OUT###NULL## ######################
#################

start_lsn########################

pg_get_wal_stats(start_lsn pg_lsn, end_lsn pg_lsn, per_record boolean DEFAULT false) returns setof record

start_lsn#*end_lsn*#########WAL############## ######
##*resource_manager*####1######## *per_record*#true########
##*record_type*###1######## #######

```
postgres=# SELECT * FROM pg_get_wal_stats('0/1E847D00', '0/1E84F500')
           WHERE count > 0 AND
                 "resource_manager/record_type" = 'Transaction'
           LIMIT 1;
-[ RECORD 1 ]----------------+-------------------
resource_manager/record_type | Transaction
count                        | 2
count_percentage             | 8
record_size                  | 875
record_size_percentage       | 41.23468426013195
fpi_size                     | 0
fpi_size_percentage          | 0
combined_size                | 875
combined_size_percentage     | 2.8634072910530795
```

start_lsn########################

F.35.2.

Bharath Rupireddy <bharath.rupireddyforpostgres@gmail.com>

F.36. postgres_fdw — ###PostgreSQL######

postgres_fdw#########PostgreSQL##########################postgres_fdw##### ##########

####################dblink####################postgres_fdw################## ##

postgres_fdw###################:

1. CREATE EXTENSION#####postgres_fdw############
2. CREATE SERVER## user## #password##################################
3. CREATE USER MAPPING## ##### ####user###password##################################
4. CREATE FOREIGN TABLE####IMPORT FOREIGN SCHEMA###################### ############ ## ##

##SELECT####### # ##INSERT#UPDATE#DELETE#COPY#TRUNCATE####################### ############ ##

#####################SELECT#UPDATE#DELETE#TRUNCATE####ONLY############## ######

postgres_fdw######ON CONFLICT DO UPDATE####INSERT############## ###### ###########################ON CONFLICT DO NOTHING######## ## postgres_fdw#################UPDATE################################## ################################UPDATE#################################### ##

###
postgres_fdw###
##

##################

F.36.1. postgres_fdw################

F.36.1.1.

postgres_fdw##32.1.2######## libpq##################################

- #####[90.131579]#user#password###sslpassword############################ ################
- client_encoding################################
- application_name-######postgres_fdw.application_name#################### ### ######postgres_fdw.application_name############ libpq#### ##postgres_fdw#application_name######################## ## #postgres_fdw.application_name#########
- fallback_application_name#####postgres_fdw########
- sslkey#sslcert - ################################ ######################### #######

#########sslcert#sslkey###########################

##################GSSAPI###
###password###########

############password_required 'false'#####################################
#############

```
ALTER USER MAPPING FOR some_non_superuser SERVER loopback_nopw
OPTIONS (ADD password_required 'false');
```

########postgres#########unix###
#################

CVE-2007-3278#CVE-2007-6601##
############# public#####password_required=false########### postgres#######
##########unix#########################.pgpass#.pg_service.conf########
############################ peer#ident##############################

F.36.1.2.

################PostgreSQL########SQL#################### #################
##############################

schema_name (string)

> ## ######################
> #####

table_name (string)

> ## ##########################

column_name (string)

> ## #########################

F.36.1.3.

postgres_fdw###
################################## ##
###############
##########postgres_fdw#################

use_remote_estimate (boolean)

> #######################################postgres_fdw######EXPLAIN########
> ############ ## #
> ######false###

fdw_startup_cost (floating point)

> ##
> ### ## #######100###

fdw_tuple_cost (floating point)

> #####[88.051948]###
> ################ ##
> ################### #######0.01### ######################################

##################################### ############################## ########
######################### #######0.2###

use_remote_estimate#true###postgres_fdw#######################
#fdw_startup_cost#fdw_tuple_cost###### ###use_remote_estimate#false#
##postgres_fdw##################fdw_startup_cost#fdw_tuple_cost#######
############
#############ANALYZE##
########## ## #####
##

##############ANALYZE###########

analyze_sampling(string)

##ANALYZE######################
############################## ##########
##off#random#system#bernoulli#auto### off#########################
################### random#random()################################
##system#bernoulli##########TABLESAMPLE########## random###########
#########TABLESAMPLE#9.5############# auto(#####)######################
#####################bernoulli###random###########

F.36.1.4.

#########################WHERE###################### ##########################
######### ##WHERE######
################################## #########################

extensions(string)

#########PostgreSQL##
IMMUTABLE## ######################
#################

extensions###
##

fetch_size(integer)

#########postgres_fdw#1########################### ######################
####### ## ######100###

batch_size(integer)

#########postgres_fdw####INSERT############### ##########################
######1###

postgres_fdw#######################batch_size################# ########
##############libpq######postgres_fdw#################################
#####65535################## ## * batch_size###################
#batch_size#######

############################### #####postgres_fdw#######################
#############COPY#############1000#########

F.36.1.5.

postgres_fdw############## ################Append######################## #
###################

`async_capable` (boolean)

#########postgres_fdw################################ ################### ####### ################################ ######false###

###########################postgres_fdw######################### ######## ## ############## ##

Append## ##### ################postgres_fdw## ## ## ###########################

F.36.1.6.

#########################postgres_fdw## ### ################################### ###########postgres_fdw### ######### ##postgres_fdw#### ## ################ ###########

`parallel_commit` (boolean)

###
#postgres_fdw##################### ##################################### ##### ## ######false###

`parallel_abort` (boolean)

###
#postgres_fdw################### ### ### ## ######false###

##

###########

F.36.1.7.

#######postgres_fdw###

`updatable` (boolean)

#########postgres_fdw#INSERT#UPDATE####DELETE######################### ########### ## #######true###

###information_schema################ #postgres_fdw## ################

F.36.1.8.

#######postgres_fdw############################ ###################

truncatable (boolean)

#######postgres_fdw#######TRUNCATE######################## ######### ############### ### ######true###

############################## ### ##################

F.36.1.9.

postgres_fdw#IMPORT FOREIGN SCHEMA########################### ######## ## ######################### ######################################

###########################IMPORT FOREIGN SCHEMA###########

import_collate (boolean)

###########COLLATE## ##### #true### ## ##########OS ####################### ## #######################

#######true## ######COLLATE "default"##

import_default (boolean)

###########DEFAULT## ######false# ## ## nextval()###### ###### ##IMPORT######

import_generated (boolean)

################################GENERATED################# ######true### ######################################IMPORT######

import_not_null (boolean)

###########NOT NULL## ##### #true###

NOT NULL#################################### PostgreSQL#######check ################################### ## ## check## #####check##### ## #######check############## #CREATE FOREIGN TABLE########

################################LIMIT TO####################### ######### ##IMPORT FOREIGN SCHEMA########### ##################################### ##

F.36.1.10.

#######postgres_fdw##

keep_connections (boolean)

#################postgres_fdw################################## ######## ######### ######on### off##

F.36.2.

postgres_fdw_get_connections(OUT server_name text, OUT valid boolean) returns setof record

######postgres_fdw## ##### ############# ## #false#########################NULL#################################### ####### #######true###### ################### ###########

```
postgres=# SELECT * FROM postgres_fdw_get_connections() ORDER BY 1;
 server_name | valid
-------------+-------
 loopback1   | t
 loopback2   | f
```

postgres_fdw_disconnect(server_name text) returns boolean

######postgres_fdw## ######## ## ######################## ########################## ######################true###########false## #### ############################### #############

```
postgres=# SELECT postgres_fdw_disconnect('loopback1');
 postgres_fdw_disconnect
-------------------------
 t
```

postgres_fdw_disconnect_all() returns boolean

######postgres_fdw### ########### ## ######################true###### # #####false###### #############

```
postgres=# SELECT postgres_fdw_disconnect_all();
 postgres_fdw_disconnect_all
----------------------------
 t
```

F.36.3.

postgres_fdw### ######### ################################### ############keep_connections############ #### ##

##

####keep_connections############postgres_fdw_disconnect##### #postgres_fdw_disconnect_all######## ###################################

F.36.4.

##postgres_fdw##################### ### ## ########### ################################

############SERIALIZABLE#######################SERIALIZABLE###########
########REPEATABLE READ########### ################################
#################################### ####################################
############SERIALIZABLE###
#REPEATABLE READ############################ ################READ
COMMITTED############################## ###PostgreSQL######################
#####

postgres_fdw##

F.36.5.

####################postgres_fdw#################### ########WHERE#####
################################
#extensions##WHERE#######
#######WHERE############IMMUTABLE########## UPDATE####DELETE#########
###############WHERE###BEFORE####AFTER#
######################CHECK OPTION##########postgres_fdw##############
########################## UPDATE##
##IMMUTABLE####IMMUTABLE###############

##########################postgres_fdw################### ################
JOIN########
##WHERE#######################

############################EXPLAIN VERBOSE#############

F.36.6.

postgres_fdw###############search_path######pg_catalog####################
################### postgres_fdw##
######################
#################################### ####################################
############SET search_path#####(CREATE FUNCTION##)#############

postgres_fdw##############################

- TimeZone#UTC#######
- DateStyle#ISO#######
- IntervalStyle#postgres#######
- extra_float_digits#########9.0####3##################2########

###search_path##########################SET##############

postgres_fdw################
##

F.36.7.

postgres_fdw#########PostgreSQL 8.3################### ##########8.1########
###postgres_fdw#IMMUTABLE#######################WHERE#############
##
#################### ############################## #######################
#OFFSET 0####sub-SELECT################sub-SELECT################

F.36.8. Wait Events

######postgres_fdw#########Extension##################

PostgresFdwCleanupResult

#########################

PostgresFdwConnect

#########################

PostgresFdwGetResult

##########################

F.36.9.

postgres_fdw.application_name (string)

postgres_fdw#####################application_name############### ######application_name############# ##################################

postgres_fdw.application_name############ASCII############# ######application_name##############NAMEDATALEN###################### ####ASCII#######C#####16################## ###application_name#########

%## ################## ################ ##############%##############/########################### ###########

#####	##
%a	##################
%c	##############ID####log_line_prefix####
%C	##################cluster_name####
%u	#############
%d	################
%p	####################ID
%%	## %

#########local_user#######local_db##foreign_db#########foreign_user## ######### ##'db=%d,user=%u'#'db=local_db,user=local_user'#########

F.36.10.

###postgres_fdw############### ###############

```
CREATE EXTENSION postgres_fdw;
```

###CREATE SERVER############### #########192.83.123.89####5432####### PostgreSQL######### ####################foreign_db########

```
CREATE SERVER foreign_server
        FOREIGN DATA WRAPPER postgres_fdw
        OPTIONS (host '192.83.123.89', port '5432', dbname 'foreign_db');
```

##CREATE USER MAPPING########

```
CREATE USER MAPPING FOR local_user
```

```
        SERVER foreign_server
        OPTIONS (user 'foreign_user', password 'password');
```

###CREATE FOREIGN TABLE#######################################
#some_schema.some_table#################################foreign_table###

```
CREATE FOREIGN TABLE foreign_table (
        id integer NOT NULL,
        data text
)
        SERVER foreign_server
        OPTIONS (schema_name 'some_schema', table_name 'some_table');
```

CREATE FOREIGN TABLE###
#################column_name###
##########IMPORT FOREIGN SCHEMA#############

F.36.11.

<shigeru.hanada@gmail.com>

F.37. seg —

####################seg########## seg#################################

########trusted#######################CREATE######################

F.37.1.

########################## ########################## ######################
###

############################
######

#
####6.50########################## ################# ########

```
test=> select 6.50 :: float8 as "pH";
 pH
---
6.5
(1 row)
```

#########6.50#6.5########## ######################## ###################### 6.50###
###6.5################################ 2########################### ############
########################

############################

##########

```
test=> select '6.25 .. 6.50'::seg as "pH";
          pH
------------
6.25 .. 6.50
(1 row)
```

F.37.2.

#########1####2###############..###...############## ##########################
###########<#>#~############## ################################ # F.27#######
######### F.28###########

F.27###x#y#delta############# delta###x#y####################

#F.27 seg####

x	#####0####
x .. y	x##y#####
x (+-) delta	x - delta##x + delta#####
x ..	###x##########
.. x	###x##########

#F.28 ###SEG####

5.0	#0########################

~5.0	#0###############~####### ~#seg################### ########
<5.0	5.0########## <#####################
>5.0	5.0########## >#####################
5(+-)0.3	4.7 .. 5.3########### (+-)#################### ##
50 ..	50######
.. 0	0######
1.5e-2 .. 2E-2	0.015 .. 0.02##########
1 ... 2	1...2#1 .. 2#1..2########################

#######...#############..#################### ######################## 0...23####23###0.23############# ####seg############1#################### #########

########seg##5 .. 2#########################

F.37.3.

####seg###32#################### ###7############################

#######7#####7################## ##########0.00###########0#################### ##### ################## ####0############## 0.0067#2##################

F.37.4.

seg#######seg###GiST################## GiST################## F.29######

#F.29 Seg GiST###

##
seg << seg → boolean 1###seg####2########### b < c###[a, b] << [c, d]#####
seg >> seg → boolean 1###seg####2########### a > d###[a, b] >> [c, d]#####
seg &< seg → boolean 1###seg#2############## b <= d###[a, b] &< [c, d]#####
seg &> seg → boolean 1###seg#2############## a >= c###[a, b] &> [c, d]#####
seg = seg → boolean 2##seg######
seg && seg → boolean 2##seg######
seg @> seg → boolean 1###seg#2########
seg <@ seg → boolean 1###seg#2#########

##########seg#### 9.1################## ##########(a)#(c)########(b)#(d)###### ### ###################### ######ORDER BY##############

F.37.5.

#################sql/seg.sql#########

(+-)### ################10########### ##############

```
postgres=> select '10(+-)1'::seg as seg;
      seg
---------

9.0 .. 11             -- ###########: 9 .. 11
```

R-tree################################## seg############################ ### #sort-segments.pl##############

F.37.6.

####Gene Selkov, Jr. <selkovjr@mcs.anl.gov>, Mathematics and Computer Science Division, Argonne National Laboratory.

GiST (http://gist.cs.berkeley.edu/)####gist###########Joe Hellerstein###https://dsf.berkeley.edu/jmh/######## ##Postgres########## #######################Argonne Lab#U.S. Department of Energy########

F.38. sepgsql — SELinux################### ###MAC#############

sepgsql##SELinux################################MAC; Mandatory Access Control### ############

##
F.38.7 #####

F.38.1.

#########PostgreSQL################SELinux###################### SELinux####### ########PostgreSQL################################ #####DML################## ## ########PostgreSQL######SQL########### ##############

SELinux################system_u:object_r:sepgsql_table_t:s0############ ######################## ############################## ###################### ############################## ## ## ##

SECURITY LABEL####################################

F.38.2.

sepgsql#SELinux####Linux#### 2.6.28 ######### ########################## ####### ##libselinux 2.1.10###selinux-policy 3.9.13########

sestatus########SELinux###############################

```
$ sestatus
SELinux status:                 enabled
SELinuxfs mount:                /selinux
Current mode:                   enforcing
Mode from config file:          enforcing
Policy version:                 24
Policy from config file:        targeted
```

SELinux##SELinux##################

############################--with-selinuxmake#autoconf##########-Dselinux={ auto | enabled | disabled }meson############# #### #libselinux-develRPM######################

##############postgresql.conf#shared_preload_libraries###### sepgsql########### ################################## #######################sepgsql.sql######## ##

###sepgsql## ########################## ######

```
$ export PGDATA=/path/to/data/directory
$ initdb
$ vi $PGDATA/postgresql.conf

  ####
    #shared_preload_libraries = ''                # (change requires
 restart)

  #####
    shared_preload_libraries = 'sepgsql'          # (change requires
 restart)
$ for DBNAME in template0 template1 postgres; do
    postgres --single -F -c exit_on_error=true $DBNAME \
      </usr/local/pgsql/share/contrib/sepgsql.sql >/dev/null
  done
```

libselinux#selinux-policy##################################

```
/etc/selinux/targeted/contexts/sepgsql_contexts:  line 33 has invalid
 object type db_blobs
/etc/selinux/targeted/contexts/sepgsql_contexts:  line 36 has invalid
 object type db_language
/etc/selinux/targeted/contexts/sepgsql_contexts:  line 37 has invalid
 object type db_language
/etc/selinux/targeted/contexts/sepgsql_contexts:  line 38 has invalid
 object type db_language
/etc/selinux/targeted/contexts/sepgsql_contexts:  line 39 has invalid
 object type db_language
/etc/selinux/targeted/contexts/sepgsql_contexts:  line 40 has invalid
 object type db_language
```

#########################

#####################################

F.38.3.

SELinux#####sepgsql##root########## ## #############make check#make installcheck################ ########### ##################### ######PostgreSQL######contrib/sepgsql################ ## ## ####make check#####make installcheck########

####F.38.2####sepgsql################# ####OS################################ #############

sepgsql-regtest################ ############################## ##############sepgsql-regtest.te###SELinux #####Makefile####make################ ######################Makefile###### ############################ (##Makefile###selinux-policy-devel#selinux-policy##############) #########semodule################################## ## ##############semodule -l######## ########sepgsql-regtest##########

```
$ cd .../contrib/sepgsql
$ make -f /usr/share/selinux/devel/Makefile
$ sudo semodule -u sepgsql-regtest.pp
$ sudo semodule -l | grep sepgsql
sepgsql-regtest 1.07
```

###sepgsql_regression_test_mode########## ############sepgsql-regtest# ########################## sepgsql_regression_test_mode################# ######################### setsebool###################

```
$ sudo setsebool sepgsql_regression_test_mode on
$ getsebool sepgsql_regression_test_mode
sepgsql_regression_test_mode --> on
```

#######unconfined_t####################

```
$ id -Z
unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

################################F.38.8########

##########################

```
$ ./test_sepgsql
```

#####################################sepgsql#########################

#######sepgsql_regression_test_mode###################

```
$ sudo setsebool sepgsql_regression_test_mode off
```

sepgsql-regtest##############################

```
$ sudo semodule -r sepgsql-regtest
```

F.38.4. GUC#####

sepgsql.permissive (boolean)

> ##############################sepgsql####################### ##########off ### postgresql.conf##
>
> ########on#######SELinux#######################sepgsql################### # #######################

sepgsql.debug_audit (boolean)

> ## #######off################### ###
>
> SELinux## ################### #####################
>
> ##

F.38.5.

F.38.5.1.

SELinux####################################; ############################## ############################## ##
#unlabeled_t######################

###sepgsql## sepgsql#######
############################
###

###
#################### ##
#########################

F.38.5.2. DML#######

##################db_table:select#db_table:insert#db_table:update###
#db_table:delete######################### ####WHERE##RETURNING###########
#UPDATE############################db_table:select###########

################################ SELECT######################DML#########
#######db_column:select########### ###UPDATE#INSERT####################
##db_column:update#db_column:insert###############

###########

```
UPDATE t1 SET x = 2, y = func1(y) WHERE z = 100;
```

#########t1.x####db_column:update##############t1.y#####db_column:
{select update}##############t1.z##db_column:select########### ########
###db_table:{select update}###########

SELECT##################db_sequence:get_value######### ########
#lastval()#####################################

########db_view:expand###

###db_procedure:{execute}#
########### ##
#db_procedure:{entrypoint}###########

#######################################db_schema:search######## #############
###USAGE##############
####### ##

###
###

#####################################DML################TOAST##############
######sepgsql####################

F.38.5.3. DDL#######

##############################SELinux## #
###

####################create######## SELinux#################################
############################## #########################

- CREATE DATABASE#######################getattr########
- ###########################add_name########
- ###
- LEAKPROOF###########install####################LEAKPROOF#############
  ###

DROP####################drop########### CASCADE##########################
##remove_name######
########

ALTER###
##setattr######### #########################

- ################################remove_name##############add_name######
  ##

- ######LEAKPROOF######install########

- SECURITY LABEL################################setattr###relabelfrom####
  ##############relabelto########### ############################SELinux####
  #################setattr##

F.38.5.4.

#############SECURITY DEFINER###Set-UID########### ##########################
###(###########################)#########SELinux##############################
################ ##
############ ####

```
postgres=# CREATE TABLE customer (
               cid     int primary key,
               cname   text,
               credit  text
           );
CREATE TABLE
postgres=# SECURITY LABEL ON COLUMN customer.credit
               IS 'system_u:object_r:sepgsql_secret_table_t:s0';
SECURITY LABEL
postgres=# CREATE FUNCTION show_credit(int) RETURNS text
             AS 'SELECT regexp_replace(credit, ''-[0-9]+$'', ''-xxxx'',
 ''g'')
                       FROM customer WHERE cid = $1'
           LANGUAGE sql;
CREATE FUNCTION
postgres=# SECURITY LABEL ON FUNCTION show_credit(int)
               IS 'system_u:object_r:sepgsql_trusted_proc_exec_t:s0';
SECURITY LABEL
```

#########################

```
postgres=# SELECT * FROM customer;
ERROR:  SELinux: security policy violation
postgres=# SELECT cid, cname, show_credit(cid) FROM customer;
 cid | cname  |     show_credit
-----+--------+---------------------
   1 | taro   | 1111-2222-3333-xxxx
   2 | hanako | 5555-6666-7777-xxxx
(2 rows)
```

###########customer.credit################################show_credit#####
##

F.38.5.5.

#######################SELinux### #######
##setcurrent############################dyntransition###############

```
##########################################################################
######## dyntransition################################################## ##
##


regression=# select sepgsql_getcon();
                    sepgsql_getcon
-----------------------------------------------------
 unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
(1 row)

regression=# SELECT
 sepgsql_setcon('unconfined_u:unconfined_r:unconfined_t:s0-s0:c1.c4');
 sepgsql_setcon
----------------
 t
(1 row)

regression=# SELECT
 sepgsql_setcon('unconfined_u:unconfined_r:unconfined_t:s0-s0:c1.c1023');
ERROR:  SELinux: security policy violation

########MCS###c1.c1023####MCS###c1.c4###########################

################################################################### ###########
#############SQL################################sepgsql_setcon()#########
#################################################################  #######
########################################### ###sepgsql_setcon()#NULL######
################################################################### ##########
########################################################################### ####
##########################################################
```

F.38.5.6.

```
#######################################LOAD####################
```

F.38.6. sepgsql##

F.30##############

#F.30 sepgsql##

##	##
sepgsql_getcon () → text	##############################
sepgsql_setcon (text) → boolean	### NULL############ #########################
sepgsql_mcstrans_in (text) → text	mcstrans###################MLS/MCS##############################
sepgsql_mcstrans_out (text) → text	mcstrans###################MLS/MCS##############################
sepgsql_restorecon (text) → boolean	### ###NULL################### ########

F.38.7.

Data Definition Language (DDL) #######

###############DDL##################

Data Control Language (DCL) #######

##########DCL##################

#########

PostgreSQL#####################`sepgsql`###########

######

#############################`sepgsql`################## ################## ## "####"######### ################################

F.38.8.

SE-PostgreSQL Introduction[6]

##wiki##

SELinux User's and Administrator's Guide[7]

#########SELinux####################### ####Red Hat########################## #############

Fedora SELinux FAQ[8]

#########SELinux#############(FAQ)### ####Fedora##########################

F.38.9.

KaiGai Kohei `<kaigai@ak.jp.nec.com>`

[6] https://wiki.postgresql.org/wiki/SEPostgreSQL
[7] https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/7/html/selinux_users_and_administrators_guide/index
[8] https://fedoraproject.org/wiki/SELinux_FAQ

F.39. spi — ####################/#

spi#####################(SPI)######################### ##################### ######################### ##

##

F.39.1. refint —

check_primary_key()###check_foreign_key()##################### ######## ##

check_primary_key()############ ##############BEFORE INSERT OR UPDATE# ########################### ######################################/########## ########## ################################

check_foreign_key()############## ##############BEFORE DELETE OR UPDATE############################ ################################ ################cascade — ######restrict — ##################### ##setnull—#########NULL#########/## ##################### #####/######NOT NULL################################## ##########

refint.example#######

F.39.2. autoinc —

autoinc()############################ ################################## autoinc()###

##############BEFORE INSERT#### BEFORE INSERT OR UPDATE############# 2# ## ################################ ###############

autoinc.example#######

F.39.3. insert_username —

insert_username()############################ ########################### #######

##############BEFORE INSERT#UPDATE#################### 1################## ############

insert_username.example#######

F.39.4. moddatetime —

moddatetime()######timestamp################ ############################ ###

##############BEFORE UPDATE############ 1####################### ### #timestamp####timestamp with time zone###########

moddatetime.example#######

F.40. sslinfo — #######SSL#######

##########PostgreSQL########## SSL ##########`sslinfo`########### ###### SSL ### ###################### NULL #######

################################`pg_stat_ssl`################

#######`--with-ssl=openssl`############################

F.40.1.

`ssl_is_used() returns boolean`

############## SSL ####### true######## false ######

`ssl_version() returns text`

SSL####################TLSv1.0#TLSv1.1#TLSv1.2###TLSv1.3#######

`ssl_cipher() returns text`

SSL##################DHE-RSA-AES256-SHA#######

`ssl_client_cert_present() returns boolean`

#################### SSL ############### true######## false ###### ############# ################################

`ssl_client_serial() returns numeric`

######################### ## ###################### ######################################

##########

`ssl_client_dn() returns text`

########################## ############################## ######## ASCII # ################################ ####### SQL_ASCII ######################## ASCII ### UTF-8 ###############

#####`/CN=Somebody /C=Some country /O=Some organization`########

`ssl_issuer_dn() returns text`

######################### ############################## ############ `ssl_client_dn` ######

######################################

########

`ssl_client_dn_field(fieldname text) returns text`

########################### ############### NULL ###### ###### OpenSSL ###### ########### ASN1 ###################### #############

`commonName (alias CN)`

```
surname (alias SN)
name
givenName (alias GN)
countryName (alias C)
localityName (alias L)
stateOrProvinceName (alias ST)
organizationName (alias O)
organizationalUnitName (alias OU)
title
description
initials
postalCode
streetAddress
generationQualifier
description
dnQualifier
x500UniqueIdentifier
pseudonym
role
emailAddress
```

commonName ############## ################################### ####X.500 ## X.509 ######################## ##########################

ssl_issuer_field(fieldname text) returns text

##ssl_client_dn_field ########

ssl_extension_info() returns setof record

##

F.40.2.

Victor Wagner <vitus@cryptocom.ru>, Cryptocom LTD

Dmitry Voronin <carriingfate92@yandex.ru>

Cryptocom # OpenSSL ############# <openssl@cryptocom.ru>

F.41. tablefunc — #########(crosstab#)

tablefunc############################ ###########################C#######
############

#########trusted#########################CREATE#######################

F.41.1.

tablefunc################ F.31######

#F.31 tablefunc###

##
normal_rand (*numvals* integer, *mean* float8, *stddev* float8) → setof float8   ################
crosstab (*sql* text) → setof record   #####N#################### ###N#########################
crosstab*N* (*sql* text) → setof table_crosstab_*N*   #####N#################### crosstab2#crosstab3#crosstab4#######   ############crosstab*N*##############
crosstab (*source_sql* text, *category_sql* text) → setof record   2##############################
crosstab (*sql* text, *N* integer) → setof record   #####crosstab(text)### ##############################*N*########
connectby (*relname* text, *keyid_fld* text, *parent_keyid_fld* text [, *orderby_fld* text], *start_with* text, *max_depth* integer [, *branch_delim* text]) → setof record   ###############

F.41.1.1. normal_rand

```
normal_rand(int numvals, float8 mean, float8 stddev) returns setof float8
```

normal_rand#####################

###*numvals*############# *mean*#########*stddev*#############

#############5#####3#1000#########

```
test=# SELECT * FROM normal_rand(1000, 5, 3);
     normal_rand
----------------------
     1.56556322244898
     9.10040991424657
     5.36957140345079
   -0.369151492880995
    0.283600703686639
       .
       .
```

```
       .
     4.82992125404908
     9.71308014517282
     2.49639286969028
(1000 rows)
```

F.41.1.2. crosstab(text)

```
crosstab(text sql)
crosstab(text sql, int N)
```

crosstab######################### ################################ ##########
###########

```
row1    val11
row1    val12
row1    val13
...
row2    val21
row2    val22
row2    val23
...
```

#################

```
row1    val11   val12   val13   ...
row2    val21   val22   val23   ...
...
```

crosstab######################SQL###################2###################
##

sql###################SQL#### ####row_name##1##category##1##value##1######
####### *N*########################### ##
#################

######SQL####################

```
 row_name    cat    value
----------+-------+-------
  row1      cat1    val1
  row1      cat2    val2
  row1      cat3    val3
  row1      cat4    val4
  row2      cat1    val5
  row2      cat2    val6
  row2      cat3    val7
  row2      cat4    val8
```

crosstab###setof record################ ######################SELECT#
#FROM############## ##########

```
SELECT * FROM crosstab('...') AS ct(row_name text, category_1 text,
 category_2 text);
```

###################

```
           <== value  columns  ==>
 row_name   category_1   category_2
----------+------------+------------
  row1        val1         val2
  row2        val5         val6
```

FROM#####1##row_name##SQL#####################N##value##SQL######3#######
############################## ############################## ################
##

crosstab######row_name####################1########### ###########value###
#####value######### ###############value###############NULL###### ########
#############

###########################row_name#########################SQL######
#ORDER BY 1,2############# crosstab##########2###################### #####
######3##################################

############

```
CREATE TABLE ct(id SERIAL, rowid TEXT, attribute TEXT, value TEXT);
INSERT INTO ct(rowid, attribute, value) VALUES('test1','att1','val1');
INSERT INTO ct(rowid, attribute, value) VALUES('test1','att2','val2');
INSERT INTO ct(rowid, attribute, value) VALUES('test1','att3','val3');
INSERT INTO ct(rowid, attribute, value) VALUES('test1','att4','val4');
INSERT INTO ct(rowid, attribute, value) VALUES('test2','att1','val5');
INSERT INTO ct(rowid, attribute, value) VALUES('test2','att2','val6');
INSERT INTO ct(rowid, attribute, value) VALUES('test2','att3','val7');
INSERT INTO ct(rowid, attribute, value) VALUES('test2','att4','val8');

SELECT *
FROM crosstab(
  'select rowid, attribute, value
   from ct
   where attribute = ''att2'' or attribute = ''att3''
   order by 1,2')
AS ct(row_name text, category_1 text, category_2 text, category_3 text);

 row_name | category_1 | category_2 | category_3
----------+------------+------------+------------
 test1    | val2       | val3       |
 test2    | val6       | val7       |
(2 rows)
```

###################crosstab#######################FROM################## #####
####### ######FROM######################

##
psql#\crosstabview#############crosstab()############

F.41.1.3. crosstab*N*(text)

crosstab*N*(text sql)

crosstab*N*#########SELECT##################################crosstab######
################# tablefunc####################
#crosstab2#crosstab3#crosstab4########

```
CREATE TYPE tablefunc_crosstab_N AS (
    row_name TEXT,
    category_1 TEXT,
    category_2 TEXT,
        .
        .
        .
    category_N TEXT
);
```

#############text##row_name##value########2#3####4#######################
####### ################crosstab################

#####################

```
SELECT *
FROM crosstab3(
  'select rowid, attribute, value
   from ct
   where attribute = ''att2'' or attribute = ''att3''
   order by 1,2');
```

########################### ###crosstab()########################### ###crosstab
##########2######

- contrib/tablefunc/tablefunc--1.0.sql############################ ##
  ##text#######1####setof your_type_name################crosstab C#####
  ######### #############text######float8#####5#####################

  ```
  CREATE TYPE my_crosstab_float8_5_cols AS (
      my_row_name text,
      my_category_1 float8,
      my_category_2 float8,
      my_category_3 float8,
      my_category_4 float8,
      my_category_5 float8
  );

  CREATE OR REPLACE FUNCTION crosstab_float8_5_cols(text)
      RETURNS setof my_crosstab_float8_5_cols
      AS '$libdir/tablefunc','crosstab' LANGUAGE C STABLE STRICT;
  ```

- ################OUT############## ###################

  ```
  CREATE OR REPLACE FUNCTION crosstab_float8_5_cols(
      IN text,
      OUT my_row_name text,
      OUT my_category_1 float8,
      OUT my_category_2 float8,
      OUT my_category_3 float8,
      OUT my_category_4 float8,
      OUT my_category_5 float8)
    RETURNS setof record
    AS '$libdir/tablefunc','crosstab' LANGUAGE C STABLE STRICT;
  ```

F.41.1.4. crosstab(text, text)

```
crosstab(text source_sql, text category_sql)
```

#######crosstab## ############## ## 2#######crosstab############ ##############################

source_sql##############SQL#### ##SQL##row_name##1#category##1##value##1 ############ ##1################### row_name############## category#value# #########2########### row_name#category##################### ####### #row_name#######################

####*source_sql*#####################

```
SELECT row_name, extra_col, cat, value FROM foo ORDER BY 1;

 row_name    extra_col   cat    value
----------+------------+-----+---------
  row1         extra1    cat1    val1
  row1         extra1    cat2    val2
  row1         extra1    cat4    val4
  row2         extra2    cat1    val5
  row2         extra2    cat2    val6
  row2         extra2    cat3    val7
  row2         extra2    cat4    val8
```

category_sql############SQL########### ##SQL##1################ #### ####1############################## ###################### ############## *category_sql*##############

```
SELECT DISTINCT cat FROM foo ORDER BY 1;
    cat
  -------
    cat1
    cat2
    cat3
    cat4
```

crosstab###setof record### #SELECT#FROM##############

```
SELECT * FROM crosstab('...', '...')
    AS ct(row_name text, extra text, cat1 text, cat2 text, cat3 text, cat4
 text);
```

#################

```
                  <==  value  columns   ==>
row_name   extra   cat1   cat2   cat3   cat4
---------+-------+------+------+------+------
  row1     extra1  val1   val2          val4
  row2     extra2  val5   val6   val7   val8
```

FROM##################################### *source_sql*#######*N*###########*N*-2 ####*N*-2################ ######*source_sql*######################## ##*category_sql*#######################

crosstab#####row_name################1########### row_name############## ################# value#####category#######value####### # #category#*category_sql*#######################value####### ############## ###################NULL########

######row_name##############source_sql########ORDER BY 1######## ####### ################### ###category_sql######################################
##

########2######

```
create table sales(year int, month int, qty int);
insert into sales values(2007, 1, 1000);
insert into sales values(2007, 2, 1500);
insert into sales values(2007, 7, 500);
insert into sales values(2007, 11, 1500);
insert into sales values(2007, 12, 2000);
insert into sales values(2008, 1, 1000);

select * from crosstab(
  'select year, month, qty from sales order by 1',
  'select m from generate_series(1,12) m'
) as (
  year int,
  "Jan" int,
  "Feb" int,
  "Mar" int,
  "Apr" int,
  "May" int,
  "Jun" int,
  "Jul" int,
  "Aug" int,
  "Sep" int,
  "Oct" int,
  "Nov" int,
  "Dec" int
);
 year | Jan  | Feb  | Mar | Apr | May | Jun | Jul | Aug | Sep | Oct | Nov
 | Dec
------+------+------+-----+-----+-----+-----+-----+-----+-----+-----
+------+------
 2007 | 1000 | 1500 |     |     |     |     | 500 |     |     |     | 1500
 | 2000
 2008 | 1000 |      |     |     |     |     |     |     |     |     |
 |
(2 rows)


CREATE TABLE cth(rowid text, rowdt timestamp, attribute text, val text);
INSERT INTO cth VALUES('test1','01 March 2003','temperature','42');
INSERT INTO cth VALUES('test1','01 March 2003','test_result','PASS');
INSERT INTO cth VALUES('test1','01 March 2003','volts','2.6987');
INSERT INTO cth VALUES('test2','02 March 2003','temperature','53');
INSERT INTO cth VALUES('test2','02 March 2003','test_result','FAIL');
INSERT INTO cth VALUES('test2','02 March 2003','test_startdate','01 March
 2003');
INSERT INTO cth VALUES('test2','02 March 2003','volts','3.1234');

SELECT * FROM crosstab
(
  'SELECT rowid, rowdt, attribute, val FROM cth ORDER BY 1',
  'SELECT DISTINCT attribute FROM cth ORDER BY 1'
)
AS
(
       rowid text,
       rowdt timestamp,
       temperature int4,
       test_result text,
```

```
       test_startdate timestamp,
       volts float8
);
 rowid |          rowdt           | temperature | test_result |
 test_startdate      | volts
-------+--------------------------+-------------+-------------
+--------------------------+--------
 test1 | Sat Mar 01 00:00:00 2003 |          42 | PASS        |
               | 2.6987
 test2 | Sun Mar 02 00:00:00 2003 |          53 | FAIL        | Sat Mar 01
 00:00:00 2003 | 3.1234
(2 rows)
```

##crosstab## #####C###crosstab_hash#######

F.41.1.5. connectby

```
connectby(text relname, text keyid_fld, text parent_keyid_fld
          [, text orderby_fld ], text start_with, int max_depth
          [, text branch_delim ])
```

connectby########################## ## ################ connectby##########################

F.32##############

#F.32 connectby#####

#####	##
relname	##########
keyid_fld	#########
parent_keyid_fld	###########
orderby_fld	###################
start_with	#######
max_depth	###################
branch_delim	###################

##*start_with*################################

connectby###setof record### #SELECT##FROM##############

```
SELECT * FROM connectby('connectby_tree', 'keyid', 'parent_keyid', 'pos',
 'row2', 0, '~')
    AS t(keyid text, parent_keyid text, level int, branch text, pos int);
```

####2############################## ################################ 3####### #########integer########### *branch_delim*###########################text## ######### ####*orderby_fld*##########################integer###########

################################## ########*branch_delim*########### ######### ##*branch_delim*##########################

###*orderby_fld*############## ########### ################### *orderby_fld*##

#########################connectby########SQL############# ###############
################################ ############################

#######################################

branch_delim######################### ######connectby#####################
branch_delim###########################~########

##########

```
CREATE TABLE connectby_tree(keyid text, parent_keyid text, pos int);

INSERT INTO connectby_tree VALUES('row1',NULL, 0);
INSERT INTO connectby_tree VALUES('row2','row1', 0);
INSERT INTO connectby_tree VALUES('row3','row1', 0);
INSERT INTO connectby_tree VALUES('row4','row2', 1);
INSERT INTO connectby_tree VALUES('row5','row2', 0);
INSERT INTO connectby_tree VALUES('row6','row4', 0);
INSERT INTO connectby_tree VALUES('row7','row3', 0);
INSERT INTO connectby_tree VALUES('row8','row6', 0);
INSERT INTO connectby_tree VALUES('row9','row5', 0);


-- #####orderby_fld##(############)
SELECT * FROM connectby('connectby_tree', 'keyid', 'parent_keyid', 'row2',
 0, '~')
 AS t(keyid text, parent_keyid text, level int, branch text);
 keyid | parent_keyid | level |       branch
-------+--------------+-------+---------------------
 row2  |              |     0 | row2
 row4  | row2         |     1 | row2~row4
 row6  | row4         |     2 | row2~row4~row6
 row8  | row6         |     3 | row2~row4~row6~row8
 row5  | row2         |     1 | row2~row5
 row9  | row5         |     2 | row2~row5~row9
(6 rows)


-- #####orderby_fld##(############)
SELECT * FROM connectby('connectby_tree', 'keyid', 'parent_keyid', 'row2',
 0)
 AS t(keyid text, parent_keyid text, level int);
 keyid | parent_keyid | level
-------+--------------+-------
 row2  |              |     0
 row4  | row2         |     1
 row6  | row4         |     2
 row8  | row6         |     3
 row5  | row2         |     1
 row9  | row5         |     2
(6 rows)


-- #####orderby_fld##(row5#row4############)
SELECT * FROM connectby('connectby_tree', 'keyid', 'parent_keyid', 'pos',
 'row2', 0, '~')
 AS t(keyid text, parent_keyid text, level int, branch text, pos int);
 keyid | parent_keyid | level |       branch        | pos
-------+--------------+-------+---------------------+-----
 row2  |              |     0 | row2                |   1
 row5  | row2         |     1 | row2~row5           |   2
 row9  | row5         |     2 | row2~row5~row9      |   3
 row4  | row2         |     1 | row2~row4           |   4
```

```
 row6  | row4         |     2 | row2~row4~row6      |   5
 row8  | row6         |     3 | row2~row4~row6~row8 |   6
(6 rows)


-- #####orderby_fld##(row5#row4############)
SELECT * FROM connectby('connectby_tree', 'keyid', 'parent_keyid', 'pos',
 'row2', 0)
 AS t(keyid text, parent_keyid text, level int, pos int);
 keyid | parent_keyid | level | pos
-------+--------------+-------+-----
 row2  |              |     0 |   1
 row5  | row2         |     1 |   2
 row9  | row5         |     2 |   3
 row4  | row2         |     1 |   4
 row6  | row4         |     2 |   5
 row8  | row6         |     3 |   6
(6 rows)
```

F.41.2.

Joe Conway

F.42. tcn —

tcn### ###FOR EACH ROW#AFTER#########
########

#########trusted#########################CREATE#######################

CREATE TRIGGER## ####################
######## ################tcn########

#################
############
#######

#####################


```
test=# create table tcndata
test-#   (
test(#     a int not null,
test(#     b date not null,
test(#     c text,
test(#     primary key (a, b)
test(#   );
CREATE TABLE
test=# create trigger tcndata_tcn_trigger
test-#   after insert or update or delete on tcndata
test-#   for each row execute function triggered_change_notification();
CREATE TRIGGER
test=# listen tcn;
LISTEN
test=# insert into tcndata values (1, date '2012-12-22', 'one'),
test-#                            (1, date '2012-12-23', 'another'),
test-#                            (2, date '2012-12-23', 'two');
INSERT 0 3
Asynchronous notification "tcn" with payload
 ""tcndata",I,"a"='1","b"='2012-12-22'" received from server process with
 PID 22770.
Asynchronous notification "tcn" with payload
 ""tcndata",I,"a"='1","b"='2012-12-23'" received from server process with
 PID 22770.
Asynchronous notification "tcn" with payload
 ""tcndata",I,"a"='2","b"='2012-12-23'" received from server process with
 PID 22770.
test=# update tcndata set c = 'uno' where a = 1;
UPDATE 2
Asynchronous notification "tcn" with payload
 ""tcndata",U,"a"='1","b"='2012-12-22'" received from server process with
 PID 22770.
Asynchronous notification "tcn" with payload
 ""tcndata",U,"a"='1","b"='2012-12-23'" received from server process with
 PID 22770.
test=# delete from tcndata where a = 1 and b = date '2012-12-22';
DELETE 1
Asynchronous notification "tcn" with payload
 ""tcndata",D,"a"='1","b"='2012-12-22'" received from server process with
 PID 22770.
```

F.43. test_decoding — SQL####WAL#######################/######

test_decoding################## ##

test_decoding############WAL###########################

########SQL##

```
postgres=# SELECT * FROM pg_logical_slot_get_changes('test_slot', NULL,
 NULL, 'include-xids', '0');
   lsn     | xid |                       data
-----------+-----+--------------------------------------------------
 0/16D30F8 | 691 | BEGIN
 0/16D32A0 | 691 | table public.data: INSERT: id[int4]:2 data[text]:'arg'
 0/16D32A0 | 691 | table public.data: INSERT: id[int4]:3 data[text]:'demo'
 0/16D32A0 | 691 | COMMIT
 0/16D32D8 | 692 | BEGIN
 0/16D3398 | 692 | table public.data: DELETE: id[int4]:2
 0/16D3398 | 692 | table public.data: DELETE: id[int4]:3
 0/16D3398 | 692 | COMMIT
(8 rows)
```

##

```
postgres[33712]=#* SELECT * FROM pg_logical_slot_get_changes('test_slot',
 NULL, NULL, 'stream-changes', '1');
    lsn    | xid |                       data
-----------+-----+--------------------------------------------------
 0/16B21F8 | 503 | opening a streamed block for transaction TXN 503
 0/16B21F8 | 503 | streaming change for TXN 503
 0/16B2300 | 503 | streaming change for TXN 503
 0/16B2408 | 503 | streaming change for TXN 503
 0/16BEBA0 | 503 | closing a streamed block for transaction TXN 503
 0/16B21F8 | 503 | opening a streamed block for transaction TXN 503
 0/16BECA8 | 503 | streaming change for TXN 503
 0/16BEDB0 | 503 | streaming change for TXN 503
 0/16BEEB8 | 503 | streaming change for TXN 503
 0/16BEBA0 | 503 | closing a streamed block for transaction TXN 503
(10 rows)
```

F.44. tsm_system_rows — TABLESAMPLE### #SYSTEM_ROWS##########

tsm_system_rows#####SYSTEM_ROWS#################### ###SELECT#### #TABLESAMPLE########

##################################1###### ################################ ### ####################################

#####SYSTEM#############SYSTEM_ROWS## ##############################

SYSTEM_ROWS#REPEATABLE##########

#########trusted#########################CREATE#######################

F.44.1.

###SYSTEM_ROWS#############SELECT######## ###############

```
CREATE EXTENSION tsm_system_rows;
```

############SELECT##############

```
SELECT * FROM my_table TABLESAMPLE SYSTEM_ROWS(100);
```

##########my_table######100####### ##############100##################

F.45. tsm_system_time — TABLESAMPLE### #SYSTEM_TIME##########

tsm_system_time#####SYSTEM_TIME#################### ###SELECT#### #TABLESAMPLE########

##1###### ################################### ################################### ################################### ###### ##################

#####SYSTEM#############SYSTEM_TIME#######################################
#######SELECT####################

SYSTEM_TIME#REPEATABLE##########

#########trusted#########################CREATE#######################

F.45.1.

###SYSTEM_TIME#############SELECT######## ###############

```
CREATE EXTENSION tsm_system_time;
```

############SELECT##############

```
SELECT * FROM my_table TABLESAMPLE SYSTEM_TIME(1000);
```

#######1##1000#######my_table################### ############1################ ##########

F.46. unaccent —

unaccent##########(######)############### ################################## ################### ##################################

###unaccent#####thesaurus#########################

#########trusted#########################CREATE#######################

F.46.1.

unaccent##################

- RULES######################## #######$SHAREDIR/tsearch_data/########### ######$SHAREDIR#PostgreSQL########################### #####.rules######## ######.rules#RULES##############

rules###############

- ######################################1######### ############## ##########

À	A
Á	A
Â	A
Ã	A
Ä	A
Å	A
Æ	AE

  2######################################

- ##

- ##################################unaccent################################## ####

- ##################################### ################################# #### ###################2################## ####:#

¼	" 1/4"
½	" 1/2"
¾	" 3/4"
″	"""
″	"""

- ##PostgreSQL##################rules#####UTF-8##################### ########### ########################### rules## #############

unaccent.rules############################## ###unaccent############## #$SHAREDIR/tsearch_data/########### ##rules################################ ######################(####Æ#AE#)######

F.46.2.

unaccent##############unaccent##############################unaccent##### ##### unaccent###RULES='unaccent'######################## #unaccent.rules############### ########################

```
mydb=# ALTER TEXT SEARCH DICTIONARY unaccent (RULES='my_rules');
```

```
##################################
```

```
##########################
```

```
mydb=# select ts_lexize('unaccent','Hôtel');
 ts_lexize
-----------
 {Hotel}
(1 row)
```

```
#######unaccent####################
```

```
mydb=# CREATE TEXT SEARCH CONFIGURATION fr ( COPY = french );
mydb=# ALTER TEXT SEARCH CONFIGURATION fr
        ALTER MAPPING FOR hword, hword_part, word
        WITH unaccent, french_stem;
mydb=# select to_tsvector('fr','Hôtels de la Mer');
    to_tsvector
-------------------
 'hotel':1 'mer':4
(1 row)

mydb=# select to_tsvector('fr','Hôtel de la Mer') @@
 to_tsquery('fr','Hotels');
 ?column?
----------
 t
(1 row)

mydb=# select ts_headline('fr','Hôtel de la
 Mer',to_tsquery('fr','Hotels'));
      ts_headline
------------------------
 <b>Hôtel</b> de la Mer
(1 row)
```

F.46.3.

unaccent##################################### #######unaccent#################### ####################

unaccent([*dictionary* regdictionary,] *string* text) returns text

##*dictionary*#########unaccent######unaccent()###########################

#########

```
SELECT unaccent('unaccent', 'Hôtel');
SELECT unaccent('Hôtel');
```

F.47. uuid-ossp — UUID###

uuid-ossp##################1#############UUID############### ######UUID### ############ ###########PostgreSQL########################## UUID########### #9.14#########

########trusted######## ################CREATE#######################

F.47.1. uuid-ossp##

UUID################ F.33##### ######ITU-T Rec. X.667#ISO/IEC 9834-8:2005#RFC 4122[9]#UUID###############1#3#4#5######4############### ######2############### ### ################################

#F.33 UUID#####

##
uuid_generate_v1 () → uuid   #####1 UUID####### ##########MAC################## ####UUID########   ##
uuid_generate_v1mc () → uuid   ########MAC###################MAC###############1 UUID#######
uuid_generate_v3 (*namespace* uuid, *name* text) → uuid   ########name###################3 UUID####### ####### F.34####   #uuid_ns_*()#############1########### ###########UUID####### name##   ################   ############   SELECT uuid_generate_v3(uuid_ns_url(), 'http://www.postgresql.org');   name######MD5########### ##########UUID################# #######UUID#   #####################################
uuid_generate_v4 () → uuid   #####4 UUID####### #####################
uuid_generate_v5 (*namespace* uuid, *name* text) → uuid   #####5 UUID####### #####3 UUID#################SHA-1##############   SHA-1#MD5################5######3#############

#F.34 UUID#######

##
uuid_nil () → uuid   ##nil##UUID#############UUID###########
uuid_ns_dns () → uuid   DNS#####UUID############
uuid_ns_url () → uuid

[9] https://datatracker.ietf.org/doc/html/rfc4122

URL#####UUID#############
`uuid_ns_oid` () → `uuid` ISO##########OID##UUID############# ####ASN.1#OID########PostgreSQL## ###OID###########
`uuid_ns_x500` () → `uuid` X.500####DN##UID#############

F.47.2. `uuid-ossp`###

##########################OSSP UUID############## OSSP UUID########http://www.ossp.org/pkg/lib/uuid/###`uuid-ossp`#################OSSP############## FreeBSD#####BSD##############################UUID#######`libc`############# Linux#macOS#####################`libuuid`#################(###Linux##`util-linux-ng`#############)########`e2fsprogs`############# `configure`########BSD########`--with-uuid=bsd`##`e2fsprogs`#`libuuid`#####`#--with-uuid=e2fs`##OSSP UUID##########`--with-uuid=ossp`######### ##########################`configure`##############

F.47.3.

Peter Eisentraut `<peter_e@gmx.net>`

F.48. xml2 — XPath######XSLT##

xml2######XPath######XSLT########

F.48.1.

PostgreSQL 8.3###SQL/XML######XML################## ######XML#####XPath#######################################API########### #####API#################### PostgreSQL################################## #############API#################### ###################<pgsql-hackers@lists.postgresql.org>##############

F.48.2.

F.35################### #########XML###XPath############

#F.35 xml2##

##
xml_valid (*document* text) → boolean ##################XML########## ###:######PostgreSQL# #xml_is_well_formed()###### XML##################xml_valid()### ##################
xpath_string (*document* text, *query* text) → text ##########XPath############text########
xpath_number (*document* text, *query* text) → real ##########XPath############real########
xpath_bool (*document* text, *query* text) → boolean ##########XPath############boolean########
xpath_nodeset (*document* text, *query* text, *toptag* text, *itemtag* text) → text ###############XML########## ######################## <toptag> <itemtag>Value 1 which could be an XML fragment</itemtag> <itemtag>Value 2....</itemtag> </toptag> *toptag*###*itemtag*#######################
xpath_nodeset (*document* text, *query* text, *itemtag* text) → text xpath_nodeset(document, query, toptag, itemtag)######## #*toptag*######
xpath_nodeset (*document* text, *query* text) → text xpath_nodeset(document, query, toptag, itemtag)############## ######
xpath_list (*document* text, *query* text, *separator* text) → text #################################### ####*separator*#,###Value 1,Value 2,Value 3######
xpath_list (*document* text, *query* text) → text ####,########################

F.48.3. xpath_table

```
xpath_table(text key, text document, text relation, text xpaths, text
 criteria) returns setof record
```

xpath_table#########XPath################################ ################## ################################### ########### F.36#######

#F.36 xpath_table######

#####	##
key	#key########### ############################### ## ######################## ########################
document	XML###############
relation	###################
xpaths	|#######1####XPath####
criteria	WHERE###### ############### ################# #true###1=1#########

#XPath###################SQL SELECT ########### ################

```
SELECT <key>, <document> FROM <relation> WHERE <criteria>
```

##SELECT#######2############# ##################################### ############SQL##########################

################AS#####FROM################ #########

```
SELECT * FROM
xpath_table('article_id',
            'article_xml',
            'articles',
            '/article/author|/article/pages|/article/title',
            'date_entered > ''2003-01-01'' ')
AS t(article_id integer, author text, page_count integer, title text);
```

##AS####################### ####key#########XPath########### #######XPath##### ######################## XPath##########################NULL######

```
####page_count########################### ################# ################
#######XPath####################AS##################PostgreSQL########### #####
###################### ################################text################
```

SELECT##############SELECT *########### ################################ ### ## ########################

```
SELECT t.title, p.fullname, p.email
FROM xpath_table('article_id', 'article_xml', 'articles',
                 '/article/title|/article/author/@id',
                 'xpath_string(article_xml,''/article/@date'') >
 ''2003-03-20'' ')
       AS t(article_id integer, title text, author_id integer),
     tblPeopleInfo AS p
WHERE t.author_id = p.person_id;
```

####################################

F.48.3.1.

xpath_table####XPath################################ ######################### ############# ######################2############2########### ######1#######
############NULL########

####XPath## ############XPath## ############################## ###########XPath################################ ## ##########

```
CREATE TABLE test (
    id int PRIMARY KEY,
    xml text
);

INSERT INTO test VALUES (1, '<doc num="C1">
<line num="L1"><a>1</a><b>2</b><c>3</c></line>
<line num="L2"><a>11</a><b>22</b><c>33</c></line>
</doc>');

INSERT INTO test VALUES (2, '<doc num="C2">
<line num="L1"><a>111</a><b>222</b><c>333</c></line>
<line num="L2"><a>111</a><b>222</b><c>333</c></line>
</doc>');

SELECT * FROM
  xpath_table('id','xml','test',
              '/doc/@num|/doc/line/@num|/doc/line/a|/doc/line/b|/doc/line/
c',
              'true')
  AS t(id int, doc_num varchar(10), line_num varchar(10), val1 int, val2
 int, val3 int)
WHERE id = 1 ORDER BY doc_num, line_num

 id | doc_num | line_num | val1 | val2 | val3
----+---------+----------+------+------+------
  1 | C1      | L1       |    1 |    2 |    3
  1 |         | L2       |   11 |   22 |   33
```

###doc_num########2##xpath_table##################

```
SELECT t.*,i.doc_num FROM
  xpath_table('id', 'xml', 'test',
              '/doc/line/@num|/doc/line/a|/doc/line/b|/doc/line/c',
              'true')
    AS t(id int, line_num varchar(10), val1 int, val2 int, val3 int),
  xpath_table('id', 'xml', 'test', '/doc/@num', 'true')
    AS i(id int, doc_num varchar(10))
WHERE i.id=t.id AND i.id=1
ORDER BY doc_num, line_num;

 id | line_num | val1 | val2 | val3 | doc_num
----+----------+------+------+------+---------
  1 | L1       |    1 |    2 |    3 | C1
  1 | L2       |   11 |   22 |   33 | C1
(2 rows)
```

F.48.4. XSLT##

libxslt################################

F.48.4.1. `xslt_process`

```
xslt_process(text document, text stylesheet, text paramlist) returns text
```

#####XSL########################## `paramlist##a=1,b=2`#######################
########### ##

#################2################`xslt_process`#######

F.48.5.

John Gray `<jgray@azuli.co.uk>`

##########Torchbox Ltd. (www.torchbox.com)######## PostgreSQL###BSD########

##G

###########PostgreSQL####contrib######################### contrib##########
#contrib############################F##########

######contrib#################### ####################################
PostgreSQL############bin################################

G.1.

##########contrib###PostgreSQL##
#################### PostgreSQL################################PostgreSQL#####
##################

oid2name

oid2name — OID#PostgreSQL#######################

##

`oid2name` [*option*...]

##

oid2name######PostgreSQL####################################### ################65##############################

> ##
>
> #oid2name## #####OID###########################

oid2name##############OID####################### #########OID#########OID#######################

#####

oid2name####################

`-f` *filenode*
`--filenode=`*filenode*

filenode##########################

`-i`
`--indexes`

######################

`-o` *oid*
`--oid=`*oid*

oid###OID################

`-q`
`--quiet`

#########################

`-s`
`--tablespaces`

######OID#######

`-S`
`--system-objects`

###########information_schema#pg_toast#pg_catalog##################

`-t` *tablename_pattern*
`--table=`*tablename_pattern*

tablename_pattern######################

-V
--version

oid2name################

-x
--extended

##OID#

-?
--help

oid2name#######################

##oid2name################################

-d *database*
--dbname=*database*

########

-h *host*
--host=*host*

##############

-H *host*

########################PostgreSQL 12##########

-p *port*
--port=*port*

##############

-U *username*
--username=*username*

#######

################-o#-f#-t###################### -o#OID##-f##########-t#######
##LIKE###########foo%#################### ########################## #########
########################## ##############-d#################################
#######

-o#-f#-t#########-d########-d########################### ########-S###-i####
##############

-d#################OID######### ###-s########################

##

PGHOST
PGPORT
PGUSER

##############

###############PostgreSQL###########libpq###########(32.15##)######

####PG_COLOR####################### #####always#auto#never###

##

oid2name################################ ####################################
#########

# #

```
$ # ############################
$ oid2name
All databases:
    Oid  Database Name  Tablespace
----------------------------------
  17228       alvherre  pg_default
  17255     regression  pg_default
  17227      template0  pg_default
      1      template1  pg_default

$ oid2name -s
All tablespaces:
     Oid  Tablespace Name
-------------------------
    1663       pg_default
    1664        pg_global
  155151         fastdisk
  155152          bigdisk


$ # #########alvherre########
$ cd $PGDATA/base/17228


$ # ##################################10###
$ ls -lS * | head -10
-rw-------  1 alvherre alvherre 136536064 sep 14 09:51 155173
-rw-------  1 alvherre alvherre  17965056 sep 14 09:51 1155291
-rw-------  1 alvherre alvherre   1204224 sep 14 09:51 16717
-rw-------  1 alvherre alvherre    581632 sep  6 17:51 1255
-rw-------  1 alvherre alvherre    237568 sep 14 09:50 16674
-rw-------  1 alvherre alvherre    212992 sep 14 09:51 1249
-rw-------  1 alvherre alvherre    204800 sep 14 09:51 16684
-rw-------  1 alvherre alvherre    196608 sep 14 09:50 16700
-rw-------  1 alvherre alvherre    163840 sep 14 09:50 16699
-rw-------  1 alvherre alvherre    122880 sep  6 17:51 16751

$ # What file is 155173?
$ oid2name -d alvherre -f 155173
From database "alvherre":
  Filenode  Table Name
----------------------
    155173    accounts


$ # 2##########################
$ oid2name -d alvherre -f 155173 -f 1155291
From database "alvherre":
  Filenode     Table Name
-------------------------
    155173       accounts
   1155291  accounts_pkey


$ # ###################-x###############
```

```
$ oid2name -d alvherre -t accounts -f 1155291 -x
From database "alvherre":
  Filenode     Table Name      Oid  Schema  Tablespace
----------------------------------------------------
    155173       accounts   155173  public  pg_default
   1155291  accounts_pkey  1155291  public  pg_default


$ # #######################
$ du [0-9]* |
> while read SIZE FILENODE
> do
>   echo "$SIZE       `oid2name -q -d alvherre -i -f $FILENODE`"
> done
16            1155287  branches_pkey
16            1155289  tellers_pkey
17561            1155291  accounts_pkey
...


$ # ###########
$ du [0-9]* | sort -rn | while read SIZE FN
> do
>   echo "$SIZE   `oid2name -q -d alvherre -f $FN`"
> done
133466             155173    accounts
17561            1155291  accounts_pkey
1177              16717  pg_proc_proname_args_nsp_index
...


$ # ###################pg_tblspc#########
$ cd $PGDATA/pg_tblspc
$ oid2name -s
All tablespaces:
     Oid  Tablespace Name
-------------------------
    1663       pg_default
    1664        pg_global
  155151         fastdisk
  155152          bigdisk


$ # ######"fastdisk"#########################
$ ls -d 155151/*
155151/17228/  155151/PG_VERSION


$ # #########17228#######
$ oid2name
All databases:
    Oid  Database Name  Tablespace
----------------------------------
  17228       alvherre  pg_default
  17255     regression  pg_default
  17227      template0  pg_default
      1      template1  pg_default


$ # ########################################
$ cd 155151/17228
$ ls -l
total 0
-rw-------  1 postgres postgres 0 sep 13 23:20 155156
```

```
$ # ##################################
$ oid2name -d alvherre -f 155156
From database "alvherre":
  Filenode  Table Name
----------------------
    155156         foo
```

##

B. Palmer <bpalmer@crimelabs.net>

vacuumlo

vacuumlo — PostgreSQL##########################

##

`vacuumlo` [*`option`*...] *`dbname`*...

##

vacuumlo#PostgreSQL## ########`oid`## #`lo`##############OID###############(LO)########LO#######

##############lo#######`lo_manage`################ `lo_manage`###########LO#### ##########

##################################

#####

vacuumlo####################

`-l` *`limit`*
`--limit=`*`limit`*

1###########*`limit`*########1000######################### #########LO########### ########LO####1##############max_locks_per_transaction########### ##1########## ###############limit##0##########

`-n`
`--dry-run`

##################################

`-v`
`--verbose`

################

`-V`
`--version`

vacuumlo###############

`-?`
`--help`

vacuumlo##########################

vacuumlo##############################

`-h` *`host`*
`--host=`*`host`*

################

`-p` *`port`*
`--port=`*`port`*

#################

-U *username*
--username=*username*

########

-w
--no-password

################### #####################.pgpass#######################
########################### ##
##########

-W
--password

vacuumlo##############################

#################vacuumlo#################################### ####vacuumlo
#################################### ################-W################

##

PGHOST
PGPORT
PGUSER

##############

###############PostgreSQL###########libpq##########(32.15##)######

####PG_COLOR###################### #####always#auto#never###

##

vacuumlo############ ##vacuumlo#######################OID################## ##
########oid####lo################################### (##:##################
######################################) ###############LO####### ###########

##

Peter Mount <peter@retep.org.uk>

G.2.

############PostgreSQL############# ################contrib############### #
#PostgreSQL################################PostgreSQL####################

##H

PostgreSQL################################## PostgreSQL#######################
##################

H.1.

PostgreSQL###########2#######################

- libpq###C##
- ECPG#######SQL######PostgreSQL##################

###########[1]#PostgreSQL wiki#######
########PostgreSQL##################################
##Web###############

https://wiki.postgresql.org/wiki/List_of_drivers

H.2.

PostgreSQL################## #########pgAdmin[2]###############

H.3.

PostgreSQL###################### PL/pgSQL#PL/Tcl#PL/Perl###PL/Python###

####PostgreSQL############################## #####[3]####PostgreSQL wiki########## ##
############PostgreSQL############################## #######################
#Web###############

https://wiki.postgresql.org/wiki/PL_Matrix

H.4.

PostgreSQL################### ### ##
###########`contrib`#################### ##F####### ###PostGIS[4]############
PostgreSQL################################## ##############/##################
Slony-I[5]######################

[1] https://wiki.postgresql.org/wiki/List_of_drivers
[2] https://www.pgadmin.org/
[3] https://wiki.postgresql.org/wiki/PL_Matrix
[4] https://postgis.net/
[5] https://www.slony.info

##I

PostgreSQL########Git######################## ################################
#######################

Wikihttps://wiki.postgresql.org/wiki/Working_with_Git#Git##################

I.1. Git#############

Git################################## ################################## ######
############################

Git

1. Git#########################https://git-scm.com######### ##############Git#######
   ####################################

2. Git####################################

   ```
   git clone https://git.postgresql.org/git/postgresql.git
   ```

   ############################ ############################### ##############
   ###postgresql###################

   Git#########Git############## URL#########git#################

   ```
   git clone git://git.postgresql.org/git/postgresql.git
   ```

3. ##################cd######################################

   ```
   git fetch
   ```

Git############################### #########Git#man#############https://git-scm.com#
###############

##J

PostgreSQL#####4#################

- ############
- HTML############
- PDF####
- ###############

####PostgreSQL################################`README`############

HTML#################################### PDF#########################

J.1. DocBook

#######XML#################*DocBook*########## ######DocBook#XML############## ################

DocBook## ################### ############################## DocBook#OASIS####[1]############ ##DocBook###[2]##### ###############O'Reilly######################## NewbieDoc Docbook Guide[3]########### ###### FreeBSD############[4]#####DocBook######################################

J.2.

######################## ##############################

DocBook DTD[5]

DocBook########## ########4.5################################ DocBook DTD# SGML#######XML#########

DocBook XSL Stylesheets[6]

DocBook#####HTML#############################

#############1.77.0####################################

`xmllint`####Libxml2[7]

###############`xmllint`####XML############ PostgreSQL#################### ########Libxml2############### #######`xmllint`############################## #############

`xsltproc`####Libxslt[8]

`xsltproc`#XSLT###########XSLT###########XML#####################

[1] https://www.oasis-open.org
[2] https://www.oasis-open.org/docbook/
[3] http://newbiedoc.sourceforge.net/metadoc/docbook-guide.html
[4] https://www.freebsd.org/docproj/
[5] https://www.oasis-open.org/docbook/
[6] https://github.com/docbook/wiki/wiki/DocBookXslStylesheets
[7] http://xmlsoft.org/
[8] http://xmlsoft.org/XSLT/

FOP[9]

##########XML##PDF############### PDF#######################

##################doc##################

J.2.1. Fedora#RHEL##################

################################

```
yum install docbook-dtds docbook-style-xsl libxslt fop
```

J.2.2. FreeBSD########

pkg##########################

```
pkg install docbook-xml docbook-xsl libxslt fop
```

#######Makefile#FreeBSD#make##########doc################gmake############

J.2.3. Debian#####

Debian GNU/Linux##################### ###############

```
apt-get install docbook-xml docbook-xsl libxml2-utils xsltproc fop
```

J.2.4. macOS

MacPorts###########################

```
sudo port install docbook-xml docbook-xsl-nons libxslt fop
```

Homebrew#######################

```
brew install docbook docbook-xsl libxslt fop
```

Homebrew################################ Intel####################

```
export XML_CATALOG_FILES=/usr/local/etc/xml/catalog
```

Apple Silicon####################

```
export XML_CATALOG_FILES=/opt/homebrew/etc/xml/catalog
```

##########xsltproc################

```
I/O error : Attempt to load network entity http://www.oasis-open.org/
docbook/xml/4.5/docbookx.dtd
postgres.sgml:21: warning: failed to load external entity "http://
www.oasis-open.org/docbook/xml/4.5/docbookx.dtd"
...
```

[9] https://xmlgraphics.apache.org/fop/

Apple##########xmllint#xsltproc#MacPorts#Homebrew#################DocBook#DTD##

J.2.5. configure#####

PostgreSQL##########################configure################ ##

```
checking for xmllint... xmllint
checking for xsltproc... xsltproc
checking for fop... fop
checking for dbtoepub... dbtoepub
```

xmllint#xsltproc#################### fop#PDF################## dbtoepub#EPUB##################

####configure##################################

```
./configure ... XMLLINT=/opt/local/bin/xmllint ...
```

Meson####PostgreSQL#########17.4##########meson setup#####J.4#########

J.3. Make###########

###########doc/src/sgml##################1######### #GNU make##########

J.3.1. HTML

HTML####################

```
doc/src/sgml$ make html
```

############## ###html###############

################postgresql.org[10]###############HTML###################

```
doc/src/sgml$ make STYLE=website html
```

STYLE=website################HTML#####postgresql.org[11]#######################################

J.3.2.

DocBook refentry################*roff##########DocBook XSL############## ##########################

```
doc/src/sgml$ make man
```

J.3.3. PDF

FOP####PDF##################################1#########

[10] https://www.postgresql.org/docs/current/
[11] https://www.postgresql.org/docs/current/

- A4###:

```
doc/src/sgml$ make postgres-A4.pdf
```

- U.S.######:

```
doc/src/sgml$ make postgres-US.pdf
```

PostgreSQL###########FOP#################### ################################ ################# #########~/.foprc#Java#######################

```
# FOP binary distribution
FOP_OPTS='-Xmx1500m'
# Debian
JAVA_ARGS='-Xmx1500m'
# Red Hat
ADDITIONAL_FLAGS='-Xmx1500m'
```

#################################### #########(1GB##)########################## #############

##########FOP#######INFO########### ######~/.foprc#######

```
LOGCHOICE=-Dorg.apache.commons.logging.Log=
org.apache.commons.logging.impl.SimpleLog
LOGLEVEL=-Dorg.apache.commons.logging.simplelog.defaultlog=WARN
```

##XSL-FO############################FOP##############

J.3.4.

################## ############################ ########### ##############

```
doc/src/sgml$ make check
```

J.4. Meson############

#####[75.706215]#Meson##############2################ ###############build###### ###########-C build######### ######[###]################################### ###################-C##########

HTML####################

```
build$ ninja html
```

For a list of other documentation targets see 17.4.4.3. ##########build/doc/src/sgml#### ####

J.5.

XML## #####XML################## ##DocBook####################

###########XML####################.sgml############################## ##### ##################################

J.5.1. Emacs

Emacs########nXML Mode##Emacs#XML########################## #######Emacs ################################DocBook######## ##########nXML manual[12]###### ####

`src/tools/editors/emacs.samples#####################`

J.6.

J.6.1.

########################## ## ##################### #####PostgreSQL## ######################## ######################## ############################## ##############################

##############################

##

    ##

##

    ################### ## ######## ##################################

##

    ########################

#####

    ########################### ###########################

#######

    ### ######################## ########

####

    ################################ ################################ ######### ########################### ##################

##

    ############################## ################### SHELL################################## ##########

####

    ##################################### ## ##########

[12] https://www.gnu.org/software/emacs/manual/html_mono/nxml-mode.html

##

    ##################################### ########################### ###########
    ################# ##

##

    ###

#

    ########

####

    ################################## ################

##

    ###contrib######

####

    ###################### ##PostgreSQL###############PostgreSQL#SQL#############
    ##PostgreSQL###
    ###################

SQL#################################### ##
####################
###################### ######################SQL##############################
############ SQL##################################SQL################

##K PostgreSQL###

K.1#PostgreSQL##############

#K.1 PostgreSQL###

##	##	####
##########	###	
########	4,294,950,911	
################	1,431,650,303	
##########	32 TB	`BLCKSZ`######8192#######
#########	4,294,967,295############# ########	
#########	1,600	1##
#######	1,664	
#########	1 GB	
##############	###	##################################
###########	32	PostgreSQL###################
#########	32	PostgreSQL###################
######	63###	PostgreSQL###################
function arguments	100	PostgreSQL###################
##########	65,535	

#########8192####1## ##############1,600#`int`###6400################1#########1,600#`bigint`###12800##############1########### `text`#`varchar`#`char`###TOAST########### 18############################## ##################4#### #1#################################

########################## ####################################NULL#######NULL ###########NULL###############

##L

###PostgreSQL########PostgreSQL####################

AM

#############

ANSI

American National Standards Institute#######[1]

API

Application Programming Interface##########[2]

ASCII

American Standard Code for Information Interchange##############[3]

BKI

Backend Interface###############

CA

Certificate Authority#####[4]

CIDR

Classless Inter-Domain Routing##################[5]

CPAN

Comprehensive Perl Archive Network######[6]

CRL

Certificate Revocation List##########[7]

CSV

Comma Separated Values#########[8]

CTE

Common Table Expression#########

CVE

Common Vulnerabilities and Exposures##########[9]

[1] https://en.wikipedia.org/wiki/American_National_Standards_Institute
[2] https://en.wikipedia.org/wiki/API
[3] https://en.wikipedia.org/wiki/Ascii
[4] https://en.wikipedia.org/wiki/Certificate_authority
[5] https://en.wikipedia.org/wiki/Classless_Inter-Domain_Routing
[6] https://www.cpan.org/
[7] https://en.wikipedia.org/wiki/Certificate_revocation_list
[8] https://en.wikipedia.org/wiki/Comma-separated_values
[9] https://cve.mitre.org/

DBA

Database Administrator##########[10]

DBI

Database Interface (Perl)#[Perl#]##########[11]

DBMS

Database Management System##########[12]

DDL

Data Definition Language########[13]#`CREATE TABLE`#`ALTER USER`###SQL####

DML

Data Manipulation Language########[14]#`INSERT`# `UPDATE`#`DELETE`###SQL####

DST

Daylight Saving Time#####[15]

ECPG

Embedded C for PostgreSQL#PostgreSQL######C#

ESQL

Embedded SQL#####SQL#[16]

FAQ

Frequently Asked Questions#######[17]

FSM

Free Space Map########

GEQO

Genetic Query Optimizer################

GIN

Generalized Inverted Index###########

GiST

Generalized Search Tree########

Git

Git[18]

[10] https://en.wikipedia.org/wiki/Database_administrator
[11] https://dbi.perl.org/
[12] https://en.wikipedia.org/wiki/Dbms
[13] https://en.wikipedia.org/wiki/Data_Definition_Language
[14] https://en.wikipedia.org/wiki/Data_Manipulation_Language
[15] https://en.wikipedia.org/wiki/Daylight_saving_time
[16] https://en.wikipedia.org/wiki/Embedded_SQL
[17] https://en.wikipedia.org/wiki/FAQ
[18] https://en.wikipedia.org/wiki/Git_(software)

GMT

Greenwich Mean Time##########[19]

GSSAPI

Generic Security Services Application Programming Interface##########################
#########[20]

GUC

Grand Unified Configuration# ##########PostgreSQL#######

HBA

Host-Based Authentication##########

HOT

Heap-Only Tuples##########

IEC

International Electrotechnical Commission##########[21]

IEEE

Institute of Electrical and Electronics Engineers##########[22]

IPC

Inter-Process Communication#########[23]

ISO

International Organization for Standardization#########[24]

ISSN

International Standard Serial Number#############[25]

JDBC

Java Database Connectivity[26]

JIT

Just-in-Time compilation[27]

JSON

JavaScript Object Notation[28]

[19] https://en.wikipedia.org/wiki/GMT
[20] https://en.wikipedia.org/wiki/Generic_Security_Services_Application_Program_Interface
[21] https://en.wikipedia.org/wiki/International_Electrotechnical_Commission
[22] https://standards.ieee.org/
[23] https://en.wikipedia.org/wiki/Inter-process_communication
[24] https://www.iso.org/home.html
[25] https://en.wikipedia.org/wiki/Issn
[26] https://en.wikipedia.org/wiki/Java_Database_Connectivity
[27] https://en.wikipedia.org/wiki/Just-in-time_compilation
[28] https://www.json.org

LDAP

Lightweight Directory Access Protocol[29]

LSN

Log Sequence Number##########

MCF

Most Common Frequency##################

MCV

Most Common Value#######################1#

MITM

Man-in-the-middle attack#######[30]

MSVC

Microsoft Visual C[31]

MVCC

Multi-Version Concurrency Control##########

NLS

National Language Support[32]

ODBC

Open Database Connectivity[33]

OID

Object Identifier##########

OLAP

Online Analytical Processing[34]

OLTP

Online Transaction Processing################[35]

ORDBMS

Object-Relational Database Management System###########################[36]

PAM

Pluggable Authentication Modules############[37]

[29] https://en.wikipedia.org/wiki/Lightweight_Directory_Access_Protocol
[30] https://en.wikipedia.org/wiki/Man-in-the-middle_attack
[31] https://en.wikipedia.org/wiki/Visual_C++
[32] https://en.wikipedia.org/wiki/Internationalization_and_localization
[33] https://en.wikipedia.org/wiki/Open_Database_Connectivity
[34] https://en.wikipedia.org/wiki/Olap
[35] https://en.wikipedia.org/wiki/OLTP
[36] https://en.wikipedia.org/wiki/ORDBMS
[37] https://en.wikipedia.org/wiki/Pluggable_Authentication_Modules

PGSQL

PostgreSQL

PGXS

PostgreSQL Extension System#PostgreSQL#######

PID

Process Identifier#########[38]

PITR

Point-In-Time Recovery###############Continuous Archiving - #########

PL

Procedural Languages####### ######

POSIX

Portable Operating System Interface[39]

RDBMS

Relational Database Management System####################[40]

RFC

Request For Comments[41]

SGML

Standard Generalized Markup Language##############[42]

SNI

Server Name Indication########[43], RFC 6066[44]

SPI

Server Programming Interface##################

SP-GiST

Space-Partitioned Generalized Search Tree#############

SQL

Structured Query Language############[45]

SRF

Set-Returning Function#########

[38] https://en.wikipedia.org/wiki/Process_identifier
[39] https://en.wikipedia.org/wiki/POSIX
[40] https://en.wikipedia.org/wiki/Relational_database_management_system
[41] https://en.wikipedia.org/wiki/Request_for_Comments
[42] https://en.wikipedia.org/wiki/SGML
[43] https://en.wikipedia.org/wiki/Server_Name_Indication
[44] https://datatracker.ietf.org/doc/html/rfc6066#section-3
[45] https://en.wikipedia.org/wiki/SQL

SSH

Secure Shell[46]

SSL

Secure Sockets Layer[47]

SSPI

Security Support Provider Interface[48]

SYSV

Unix System V[49]

TCP/IP

Transmission Control Protocol (TCP) / Internet Protocol (IP)[50]

TID

Tuple Identifier########

TLS

Transport Layer Security[51]

TOAST

The Oversized-Attribute Storage Technique##########

TPC

Transaction Processing Performance Council##################[52]

URL

Uniform Resource Locator###########[53]

UTC

Coordinated Universal Time#######[54]

UTF

Unicode Transformation Format###########[55]

UTF8

Eight-Bit Unicode Transformation Format###############[56]

[46] https://en.wikipedia.org/wiki/Secure_Shell
[47] https://en.wikipedia.org/wiki/Secure_Sockets_Layer
[48] https://msdn.microsoft.com/en-us/library/aa380493%28VS.85%29.aspx
[49] https://en.wikipedia.org/wiki/System_V
[50] https://en.wikipedia.org/wiki/Transmission_Control_Protocol
[51] https://en.wikipedia.org/wiki/Transport_Layer_Security
[52] http://www.tpc.org/
[53] https://en.wikipedia.org/wiki/URL
[54] https://en.wikipedia.org/wiki/Coordinated_Universal_Time
[55] https://www.unicode.org/
[56] https://en.wikipedia.org/wiki/Utf8

UUID

Universally Unique Identifier#########

WAL

Write-Ahead Log##########

XID

Transaction Identifier############

XML

Extensible Markup Language##############[57]

[57] https://en.wikipedia.org/wiki/XML

##M

###PostgreSQL##

ACID

Atomicity######Consistency######Isolation### ###Durability###### ################################## ##############################

Aggregate function######### ###

#######9.21##########

Window function################

Access Method

PostgreSQL

#####[55.952381]########13.2########## ############### 61###62###########

Analytic function######

Window function################

Analyze###########

################

#######EXPLAIN#####ANALYZE##################

#######ANALYZE##########

Atomic#####

datum################################

##################################

Atomicity#####

###ACID########

Attribute####

######################

Autovacuum############## ##

#########################

#######24.1.6##########

Auxiliary process########

the *autovacuum launcher* (but not the autovacuum workers), the *background writer*, the *checkpointer*, the *logger*, the *startup process*, the *WAL archiver*, the *WAL receiver* (but not the *WAL senders*), the *WAL summarizer*, and the *WAL writer*.

Backend##############

##

###

Background worker######### ###########

##################################### ############## ########PostgreSQL############# ################### #####################

########46##########

Background writer#########
#########

##########
#########I/O#######################I/O############
######################

#######19.4.5#########

Base Backup##########

database cluster################### pg_basebackup######
####### WAL#######################################
################

Bloat######

##
####

Bootstrap superuser#########
########

#######################

################################### ###############
############ ######################

#####[66.037736]############################## ####
##
##

Buffer Access Strategy######
#########

####################### ########################
###########################

####
########
##################*WAL*##########################
###

################################
##VACUUM#COPY#CREATE TABLE AS SELECT#ALTER TABLE#CREATE DATABASE#CREATE INDEX#CLUSTER#
################

Cast######

datum#######################

#######CREATE CAST#########

Catalog######

##SQL##########PostgreSQL#*database*############

#######*system catalog*############

#######22.1#########

Check constraint########

#################################### #############
##

#######5.5#########

Checkpoint#########

###*WAL*###
########################## ######################
##WAL##########

##
############## ##################################
################################## ###
##CHECKPOINT###################

	#######28.5##########
Checkpointer##############	######################
Class##########	Relation##########
Client##############	## ############
Cluster owner#########	##############`postgres`########################## #####################################
	`root`##
Column###	##################
Commit######	## ##############
	#######COMMIT##########
Concurrency#####	############################## PostgreSQL############ ##########*(multiversion concurrency control*#############
Connection####	## ###### ######## ##############################
	#######19.3##########
Consistency#####	##################*(integrity constraints)*###### ######## ## ############################## ###ACID########
Constraint####	##################################
	#######5.5##########
Cumulative Statistics System## ########	##################################
	#######27.2##########
Data area#######	Data directory#############
Database########	*#####SQL*##############
	#######22.1##########
Database cluster############	#############SQL################################ ## *#####(cluster)*########## ############initdb############ ####
	PostgreSQL##################*(cluster)*################ ### #######SQL#####`CLUSTER`############
	#######(cluster owner)################################ *########(bootstrap superuser)*######PostgreSQL########## ###

Database server########## Instance#########

Database superuser######### ###### ################21.2#####

################

Data directory########## ## ##*WAL*############################## ####PGDATA### ########################

###

#######65.1#########

Data page######## ########################## ############### ######## ## ######################*(dirty)*###### ################### ###(clean)###### ################################

Datum SQL###############

Delete #########################SQL#####

#######DELETE#########

Domain###### ############################ ######################### ####################

#######8.18#########

Durability##### ## ## ###ACID########

Epoch###### Transaction ID#########ID###

Extension#### ##

#######36.17#########

File segment########## ########################## ####################1### ########## ######################################

#######65.1#########

################*WAL segment*#############

Foreign data wrapper######## ### ## ## ####################################

#######CREATE FOREIGN DATA WRAPPER#########

Foreign key###### ############################ #####################0# ###1###########################

Foreign server####### ##

#######CREATE SERVER#########

Foreign table############## ## ## ##############################

#######CREATE FOREIGN TABLE#########

Fork######

############################ ################### ## ###########2############### ################ *unlogged*###############*(init fork)*######

Free space map############ ##

####################################

#######65.3#########

Function#########

#######`SELECT`############### ###### ###########

#######################

#######CREATE FUNCTION#########

Grant

##SQL## ###

#######GRANT#########

Heap#####

######################### ######################### #####################

Host#####

############################## ################### # #####################################

Index################

#################################### ############# ######################

#######CREATE INDEX#########

Incremental backup

#pg_basebackup#########

#pg_combinebackup######## *#########WAL*############

#####[64.285714]########28.5######### ########### ##25.3.3#########

Insert

#######################SQL#####

#######INSERT#########

Instance########

################################## *###postmaster*##### ############# ##################################### ######### TCP################################### ####

#####################################ACID######## ##DBMS##### ###########

Isolation#####

###ACID#

#######13.2#########

Join####	######################################SQL######
Key####	##
Lock#####	##
Log file########	################################ ##
	#######24.3#########
Logged#######	*WAL*############################ ## ########`ALTER TABLE`#####*unlogged*##
Logger###########	## ## *syslogger*##
	#######19.8###########
Log record########	*WAL*#########
Log sequence number################################## (LSN)	*WAL*################*WAL*################
	#######`pg_lsn`#28.6###########
LSN	Log sequence number#############
Master###########	Primary##############
Materialized###########	##
	##
	##
Materialized view##################################	`SELECT`## `INSERT`#`UPDATE`#`DELETE`#`MERGE`#############
	#######CREATE MATERIALIZED VIEW###########
Merge	## SQL#####
	#######MERGE###########
Multi-version concurrency control###########(MVCC)	## PostgreSQL## MVCC########### ##
Null	## ##
Optimizer#########	Query planner#############
Parallel query#########	##CPU##

Partition#########

##########################

####################(parent)#########################
########## ############################## #######
##

##
#########################

Partitioned table########### ##########

##

Postmaster######

################ ##################################
#####

#######18.3##########

Primary key#####

#########null#####################################
############# ##################################
######## ####NULL############################

Primary###########

2##
################ #################

Procedure##############

######## ############COMMIT#ROLLBACK###########
#################### CALL###############

#######CREATE PROCEDURE##########

Query#######

##
###

Query planner###########

#####################################PostgreSQL#####
##

Record

Tuple#######

Recycling

WAL file#WAL#######

Referential integrity#######

##
############

Relation########

###
##
###

######################## ####################

PostgreSQL#######################

Replica###########

##
########## ######################################
####################

Replication##########

######################## ############################
##
####################

Restartpoint###########

######################

#######28.5##########

Result set######

#####[86.585366]#SQL################################ ####### SQL######SELECT####RETURNING#######
#INSERT#UPDATE#DELETE###### ###################
#########################*SQL*# #########
##RETURNING############INSERT################
########

###
#######

Revoke

###

#######REVOKE##########

Role#####

#################### ##########################
##
####

#######CREATE ROLE##########

Rollback########

####################################

#######ROLLBACK##########

Routine######

##########
############### ##################################
############

########PostgreSQL#################################
####

Row###

Tuple#######

Savepoint#########

######################## ###########################
#############

#######SAVEPOINT##########

Schema######

###################*SQL*################ #SQL#######
######################

###########SQL#######pg_catalog###########

###
############################

#######5.10##########

Segment#######

File segment#############

Select

####################SQL##### ##SELECT###########
###
###

#######SELECT##########

Sequence###############

######################## ################## ########
####################

Server#####

PostgreSQL################## ######################
##################

	################################
Session#######	##
Shared memory#######	##########################RAM# *database*########### ##*WAL*################################ ############ ###################################
	## ######### ##################################(dirty page)#######
	#######19.4.1##########
SQL object#SQL#######	`CREATE`################### ######################### ###########################
	######################## ######################### ########################## ######################### ########################
	########################### ######################### ########## ## #######
	## ######################################SQL######## ###### ##################### ######################### ###############################
	#######22.1##########
SQL standard###SQL#	SQL############
Standby###########	Replica############
Startup process#############	####################WAL###########
	## ### ################################
Superuser#########	##
System catalog##########	###########*SQL*######################## ######## #`pg_catalog`########## ########################## ###################### `pg_catalog`################ ## *##SQL*### ############################## `#information_schema`######35################### ##
	#######5.10##########
Table######	## ## ####PostgreSQL####################
	#######CREATE TABLE##########
Tablespace########	################### ####*SQL*######################### ################################ ##################

	################### ###pg_default#######SQL#### ################ #######22.6#########
Temporary table########	## # ################################## ################ ###################### #######CREATE TABLE#########
TOAST	############################ *TOAST*########### ##### ###############TOAST######### #######65.2#########
Transaction#########	################################ ################## ## ######### #######13.2#########
Transaction ID########ID#	## ######## ####*xid*####### #######xid#32###########40## ###########ID######### ########################## ####32################## #####xid##########xid#3### ##############################1##### ########xid### ##########64################ #######66.1######### #######8.19#########
Transactions per second (TPS)	######################## ########################### #### ################################
Trigger#####	#####(`INSERT`#`UPDATE`#`DELETE`#`TRUNCATE`)########## ################### ############################## ###################### #######CREATE TRIGGER#########
Tuple#####	############### ################################## ######################### ######################## ###########################
Unique constraint#######	## ################################### — ############ ###################### *NULL*################################NULL####### ############
Unlogged########	######################*WAL*######## ############## ###################### ## ## ##################
Update	###########################SQL##### ############## ####

	#######UPDATE##########
User#####	*LOGIN*######## #21.2#####
User mapping##########	## #############
	#######CREATE USER MAPPING##########
Vacuum#######	## PostgreSQL#*MVCC*############################## ## #VACUUM## #####
	#######24.1##########
View#####	SELECT################################ ########### ## ##
	#######CREATE VIEW##########
Visibility map############ ##	## ########## #######2########### 1######(all-visible)######## ################################### 2######(all-frozen)##################################
WAL	Write-ahead log############
WAL archiver#WAL######## ####	##################################*WAL*############ ########
	#######25.3##########
WAL file#WAL#####	*WAL*#####*WAL*#################### *WAL*########### ####### ### ######################## ########################## ##################################
	#WAL## #### ## #*recycling*############
	#######28.6##########
WAL record#WAL#####	###################### ############################ ######################### WAL#################### #######
	#######28.6##########
WAL receiver#WAL####### ##	##########WAL################################### ######
	#######26.2##########
WAL segment#WAL######	WAL file#WAL######
WAL sender#WAL########	########WAL######################## ########*WAL*# ##pg_receivewal##############################

WAL summarizer (process)	#####[64.361702]####################### ############ ############WAL###############
	#####[70.129870]########19.3########## ############ ##19.5.7##########
WAL writer#WAL######## ##	*WAL############WAL*################
	#######19.5##########
Window function######### ######	## ##### ################################
	### ############### ############
	#######3.5##########
Write-ahead log##########	## ######## ##*WAL*#########*WAL*##############

##N

PostgreSQL################################### ############################

N.1.

#################PG_COLOR############

1. ##always###########
2. ##auto##
3. #################

N.2.

############PG_COLORS######################## ##########*key=value*####### ## #key################# ##value##SGR#Select Graphic Rendition####################

################

error

    ##############error############

warning

    ############warning############

note

    ###########detail###hint############

locus

    ##

#######error=01;31:warning=01;35:note=01;36:locus=01###01;31 = ### ##01;35 = #######01;36 = ######01 = ##########

###
#########GCC#GNU coreutils#####GNU grep#########################

##O

PostgreSQL## ##

O.1. recovery.conf#### #postgresql.conf###

PostgreSQL 11#####recovery.conf ##################################### ######## ####PostgreSQL 12######## ############PostgreSQL 12#################

PostgreSQL 12################################PITR########################### # ###postgresql.conf#################ALTER SYSTEM###########

recovery.conf#################

PostgreSQL 15####promote_trigger_file#12#####trigger_file######### ### #pg_ctl promote###pg_promote()####################

standby_mode# ########## #####data#######standby.signal########### ###### #######################

O.2.

PostgreSQL 13############################## ##################################### ##################################### ########################21.5########### ##PostgreSQL 14################

O.3. pg_xlogdump####pg_waldump###

PostgreSQL 9.6#####pg_xlogdump########## write-ahead-log(WAL)################### ### #######pg_waldump########### pg_waldump###########pg_waldump########## ###PostgreSQL 10################

O.4. pg_resetxlog####pg_resetwal###

PostgreSQL 9.6#####pg_resetxlog########## write-ahead-log(WAL)################# ###### #######pg_resetwal########### pg_resetwal############pg_resetwal###### ########PostgreSQL 10#################

O.5. pg_receivexlog####pg_receivewal##

PostgreSQL 9.6#####pg_receivexlog########## write-ahead-log(WAL)######## ############## #######pg_receivewal########### pg_receivewal########## #pg_receivewal#############PostgreSQL 10#################

##P

########################## ############(PostgreSQL9.6#########)

PostgreSQL16

- ###

  ## #

  ## #

  SRA OSS LLC ## ## ##

  SRA OSS LLC ## ##

  ######## ## ###

  ####### ## ##

- ######

  SRA OSS LLC ## #

  ######## ####

- ##

  #################NICT################TexTra®#########################

PostgreSQL15

- ###

  ## #

  ## #

  SRA OSS LLC ## ## ##

  SRA OSS LLC ## ##

  ######## ## ###

- ######

  ## ##

  ## ##

  ## ##

  Amazon Web Services ## ##

  NTT######### ## ##

  ######## ## ##

  SRA OSS LLC ## #

  ####### ## ##

- ##

  ################NICT###############TexTra®#######################

PostgreSQL14

- ###

  ## ##

  ## #

  ## ##

  ## #

  ## ###

  SRA OSS LLC ## ## ##

  SRA OSS LLC ## ##

  SRA OSS LLC ## #

  NTT##### ## ##

  ######## ## ##

  ######## ## ###

- ######

  tkuramoto33

  Masatoshi Fukunaga

  ##@##

  ## ##

  ## ##

  SRA OSS LLC ## ##

  NTT OSS### ## ##

  NTT OSS### ## ##

  NTT##### ## ##

  ########## ## ##

  ####### ## ##

PostgreSQL13

- ###

  ## #

  ## #

  ## #

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## #

NTT##### ## ##

######## ## ###

##########HR######## ## ##

- ######

  ####### ## ##

PostgreSQL12

- ###

  ## #

  ## #

  ## ##

  ## ##

  SRA OSS, Inc. #### ## ##

  SRA OSS, Inc. #### ## #

  ######## ## ###

- ######

  ## ##

  ## ##

  SRA OSS, Inc. #### ## ##

  SRA OSS, Inc. #### ## ##

  NTT OSS### ## ##

  NTT OSS### ## ##

  NTT##### ####

  ####NTT### ## ##

  ############ ## ##

  ####### ## ##

PostgreSQL11

- ###

  ## #

  ## #

##

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## #

NTT OSS### ####

######## ## ###

- ######

azarakko

##

##

##

####NTT### ## ##

##

########## ## ##

PostgreSQL10

- ###

#

#

##

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## #

######## ## ###

########## ## ##

- ######

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### # #

PostgreSQL9.6

- ###

#

#

#

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## ##

SRA OSS, Inc. #### ## #

######## ## ###

########## ## ##

- ######

  ## ##

  ## ##

  ## ##

  ## ##

  #####

  ## ##

  SRA OSS, Inc. #### ## ##

  NTT OSS### ## ##

  NEC############(#) ## ##

  NEC############(#) Dang Minh Huong

  ####NTT### ## ##

####

#####SQL#PostgreSQL##############

#####POSTGRES##Web###[1]######

SQL####

[bowman01] *The Practical SQL Handbook*. Using SQL Variants. Fourth Edition. Judith Bowman, Sandra Emerson, Marcy Darnovsky. ISBN 0-201-70309-2. Addison-Wesley Professional. 2001.

[date97] *A Guide to the SQL Standard [#########SQL#####4-7561-2047-4]*. A user's guide to the standard database language SQL. Fourth Edition. C. J. Date, Hugh Darwen. ISBN 0-201-96426-0. Addison-Wesley. 1997.

[date04] *An Introduction to Database Systems*. Eighth Edition. C. J. Date. ISBN 0-321-19784-4. Addison-Wesley. 2003.

[elma04] *Fundamentals of Database Systems*. Fourth Edition. Ramez Elmasri, Shamkant Navathe. ISBN 0-321-12226-7. Addison-Wesley. 2003.

[melt93] *Understanding the New SQL [###########SQL:1999##############4-8947-1531-7]*. A complete guide. Jim Melton, Alan R. Simon. ISBN 1-55860-245-3. Morgan Kaufmann. 1993.

[ull88] *Principles of Database and Knowledge-Base Systems*. Classical Database Systems. Jeffrey D. Ullman. Volume 1. Computer Science Press. 1988.

[sqltr-19075-6] *SQL Technical Report*. Part 6: SQL support for JavaScript Object Notation (JSON). First Edition. 2017.

PostgreSQL#######

[sim98] *Enhancement of the ANSI SQL Implementation of PostgreSQL*. Stefan Simkovics. Department of Information Systems, Vienna University of Technology. Vienna, Austria. November 29, 1998.

[yu95] *The Postgres95. User Manual*. A. Yu, J. Chen. University of California. Berkeley, California. Sept. 5, 1995.

[fong] *The design and implementation of the POSTGRES query optimizer*[2]. Zelaine Fong. University of California, Berkeley, Computer Science Department.

#########

[ports12] #Serializable Snapshot Isolation in PostgreSQL[3]#. D. Ports, K. Grittner. VLDB Conference, August 2012.

[berenson95] #A Critique of ANSI SQL Isolation Levels[4]#. H. Berenson, P. Bernstein, J. Gray, J. Melton, E. O'Neil, P. O'Neil. ACM-SIGMOD Conference on Management of Data, June 1995.

[olson93] *Partial indexing in POSTGRES: research project*. Nels Olson. UCB Engin T7.49.1993 O676. University of California. Berkeley, California. 1993.

[ong90] #A Unified Framework for Version Modeling Using Production Rules in a Database System#. L. Ong, J. Goh. *ERL Technical Memorandum M90/33*. University of California. Berkeley, California. April, 1990.

[1] https://dsf.berkeley.edu/papers/
[2] https://dsf.berkeley.edu/papers/UCB-MS-zfong.pdf
[3] https://arxiv.org/pdf/1208.4179
[4] https://www.microsoft.com/en-us/research/wp-content/uploads/2016/02/tr-95-51.pdf

[rowe87] #The POSTGRES data model[5]#. L. Rowe, M. Stonebraker. VLDB Conference, Sept. 1987.

[seshadri95] #Generalized Partial Indexes[6]#. P. Seshadri, A. Swami. Eleventh International Conference on Data Engineering, 6–10 March 1995. Cat. No.95CH35724. IEEE Computer Society Press. Los Alamitos, California. 1995. 420–7.

[ston86] #The design of POSTGRES[7]#. M. Stonebraker, L. Rowe. ACM-SIGMOD Conference on Management of Data, May 1986.

[ston87a] #The design of the POSTGRES. rules system#. M. Stonebraker, E. Hanson, C. H. Hong. IEEE Conference on Data Engineering, Feb. 1987.

[ston87b] #The design of the POSTGRES storage system[8]#. M. Stonebraker. VLDB Conference, Sept. 1987.

[ston89] #A commentary on the POSTGRES rules system[9]#. M. Stonebraker, M. Hearst, S. Potamianos. *SIGMOD Record 18(3)*. Sept. 1989.

[ston89b] #The case for partial indexes[10]#. M. Stonebraker. *SIGMOD Record 18(4)*. Dec. 1989. 4–11.

[ston90a] #The implementation of POSTGRES[11]#. M. Stonebraker, L. A. Rowe, M. Hirohama. *Transactions on Knowledge and Data Engineering 2(1)*. IEEE. March 1990.

[ston90b] #On Rules, Procedures, Caching and Views in Database Systems[12]#. M. Stonebraker, A. Jhingran, J. Goh, S. Potamianos. ACM-SIGMOD Conference on Management of Data, June 1990.

[5] https://dsf.berkeley.edu/papers/ERL-M87-13.pdf
[6] https://citeseer.ist.psu.edu/viewdoc/summary?doi=10.1.1.40.5740
[7] https://dsf.berkeley.edu/papers/ERL-M85-95.pdf
[8] https://dsf.berkeley.edu/papers/ERL-M87-06.pdf
[9] https://dsf.berkeley.edu/papers/ERL-M89-82.pdf
[10] https://dsf.berkeley.edu/papers/ERL-M89-17.pdf
[11] https://dsf.berkeley.edu/papers/ERL-M90-34.pdf
[12] https://dsf.berkeley.edu/papers/ERL-M90-36.pdf

##

####

$, 38
$libdir, 1060
$libdir/plugins, 591, 1717
*, 113
.pgpass, 867
.pg_service.conf, 867
10## (## numeric)
::, 44
_PG_archive_module_init, 1340
_PG_init, 1060
_PG_output_plugin_init, 1329
####
    PL/pgSQL####, 1194
########, 522
#########, 1340
######, 1124
    C#####, 1128
    PL/Tcl####, 1223
######, 479
    ####, 478
#######
    #####, 2559
######, 385
    B-Tree, 386, 2203
    BRIN, 387
    #####, 390
    GIN, 387, 2233
        ####, 428
    GiST, 386, 2208
        ####, 428
    SP-GiST, 387, 2223
    #ORDER BY, 388
    ########, 396
    ##############, 393
    #####, 393
    ####, 386, 2250
    ##########, 1093
    ###, 444
    ##, 389
    ######, 1740
    ####, 1525
    ###############, 389
    ###, 387
    ##, 390
##############, 2186
##############, 393
########, 663
########, 563
#######, 17
    ###, 113
    ####, 329
    ##, 42
########, 679
#########, 32
######
    ###, 2286
#####
    PL/pgSQL####, 1193
##############, 7
########
    ####, 184
########, 519
#######
    ###, 1090
    ##, 1058
############, 679
############
    #####, 2177
##########, 393
####
    CLOSE, 1461
    DECLARE, 1631
    FETCH, 1698
    MOVE, 1727
    PL/pgSQL####, 1187
    ##########, 1692
####
    #####, 1487
#####
    ###, 2301
    ###, 30
########, 603
####
    ####### (## #########)
#####, 679
########, 488
#####, 101
#####, 109
######, 2299
########
    PL/Python####, 1247
    PL/Tcl####, 1219
####
    ################, 350
    SQL##, 36
#####, 1041
#####
    libpq ########, 872
#########
    PL/Tcl####, 1224
######
    in_range, 2204
########, 524
#####, 568
    ########, 663
####
    #########, 356
########
    ####, 79
#######, 521
########, 436

##################, 433
#########, 2001
########, 564
#####
  ###, 131
(## #)
####, 76, 629
  public, 77
  ###, 76
  ###, 77
  ###, 77
#######, 679
###############, 679
(## TOAST)
####
  libpq####, 872
######, 5, 623
####, 515
#######
  ###, 1754
  ###, 1743
  ######, 1752
######
  ######, 592
######, 666
#######, 433
######, 398
  ####, 152
  ######, 152
###, 783
####, 7, 52
  ###, 52
  ###, 53
  #####, 66
  ###, 64
  ########, 82
  ##, 80
############, 2185
##############, 2174
#####, 100
######, 632
  #####, 584
  ###, 584
######, 105
  XMLTABLE, 275
#######, 758
######, 656
######, 441
  #######, 592
#######, 2588
######, 53
  ###, 66
######, 679
######, 629
  #######, 623
  ##, 3
#########, 7, 510
#########
  ##, 696
####, 126
  ####, 375
  ####, 1041
  ####, 183
  ######, 1086
  ####, 1061
  ###enum#, 145
  #####, 44
  ##, 1041
  ##, 375
  ##, 1042
  ##, 35
  ##, 127
  ##, 1041
(## #########)
#######, 16
#######ID
  ##, 658
##########, 433
######### (## WAL)
############, 433
  ###, 1787
  #######, 436
  #######, 585
  ########, 435
  ########, 434
###, 1115
  C###, 1117
  PL/pgSQL####, 1195
  PL/Python####, 1247
  PL/Tcl####, 1221
  #######, 1154
  tsvector###########, 412
  #######, 1117
  #####, 1610
####, 30
####, 183
#######, 33
######
  ####, 149
(## ######)
#######, 134
  ##, 209
#######
  #######, 212
  ##, 209
#####, 656
######, 358, 666
############, 2278
############, 1320
############, 32
######
  libpq####, 838
#####
  ###, 522
######
  libpq####, 838

#######
    #######, 2096
##
    ######, 583
#####, 623
    ########, 511
    ##, 612
#########, 867
####
    psql###pg_dump####, 1932
#######, 214
#####
    ###, 38
#######, 466
#########, 82
###########, 82
#########, 92
##########, 389
########, 563
####
    ####, 151
######
    ##, 34
    ##, 212
###, 15
    ########, 1140
    ##########, 1134
    ##, 1147
################, 511
########, 433
#####
    #####, 177
#######, 39
########, 679
######
    ###, 1731
    ###, 1630
    ###########, 1692
    ###, 1691
######
    #####, 1044
    #######, 1050
#####
    #######/######, 2089
####, 805
########, 679
#############, 666
####, 70
####
    DECLARE, 1631
    PL/pgSQL##, 1187
######
    #######, 512
########, 1060
###########, 2071
    ########, 1140
###########, 519
#########
    SPI###, 1303
#####, 2299
###, 622
########, 94
#######, 610
############, 1060
(## ##)
#########, 884
##########, 482, 493, 783
#########, 435
#############, 7
######, 7
#########, 434
####, 1044
###
    PL/pgSQL####, 1179
###, 1133
    DELETE#, 1143
    INSERT#, 1143
    SELECT#, 1134
    ########, 1154
    UPDATE#, 1143
    ####, 1134
    ############, 1140
########, 679
############
    ################, 684
    ############, 1326
##########, 1339
#############, 1339
######, 568
######, 623
#######, 679
####, 511, 634
###########, 1323, 1326
###, 438
    ###, 442
    ##, 734
###, 622, 626
    ########, 624
    ##################, 623
    ######, 623
    #######, 624
    #########, 624
    ###, 1006
    ######, 623
    ####, 991
######
    psql, 1935
#####, 58
####, 116
###(####), 42
    (## ####)
###, 59
######
    ####, 327
######, 128
##, 251

##
PL/pgSQL####, 1183
##, 656
##
PL/Perl, 1236
###, 76
###, 130
#, 144
####, 1086
########
######, 152
####, 398
######, 428
####, 152
###
##, 326
###, 326
#####, 515
#######, 502, 1067
####### (## WITH)
########, 444
####, 1086
##
##, 327
###, 327
#, 7, 52
###, 65
#####, 66
###, 64
#####, 63
######
###, 66
####, 38
###, 145
#######, 2256
##
FROM###, 104
##############, 11
#######, 114
######, 1105
##, 56
###, 65
###, 56
###, 65
###, 58
###, 59
####, 60
##, 63
##, 56
#NULL, 58
#######, 93
####, 567
##, 98
######, 12, 45, 105
#####, 592, 1060
######, 442
#####, 15, 60
########, 433
######, 2256
###, 102
######, 433
######, 754
##########, 679
##
###, 30
#####, 76
####, 77
##, 188
##
########ID#, 658
##########ID#, 661
#####, 9, 100
########
PL/pgSQL####, 1204
PL/Python####, 1249
PL/Tcl####, 1220
########, 1133
#######, 446
####, 326
####, 326
(## ####)
#####, 35, 44
####SQL
C#####, 894
###, 1041
###
##, 1058
####, 15, 60
######, 94
#####, 94
#########
#####, 2156
####, 101
###, 1042
####, 1042
#########, 433
#####, 178
#########
SQL#####, 30
######, 656
##, 31
######## (## JIT)
###, 102
#
###, 38
####, 48
###
######, 31
#####, 31
##, 446
######, 990
###, 66
#####, 1156
#####, 2154
#####, 2559
##, 1104

    ########, 2559
#######, 38
##, 97
####, 63
##########, 867
##
    ##, 34
#####
    ####, 132
#####, 590, 596, 646
(## #####)
    ##########, 593
    ##########, 212
    ############, 593
    ##, 31
    ######, 202
    #####, 594
    ##, 200
    ##, 201
########
    libpq####, 831
##
    ####, 398
##
    ##, 250
##
    ##, 250
###, 141, 589
    POSIX######, 2297
    #####, 2296
    ##, 248
######, 487
####, 589
####, 136, 142
    ####, 143
        (## ####)
###, 525
    #####, 2468
##, 98
#######, 1627
####, 230
#####
    #######, 1090
    ######, 1080
###(###)
    ##, 327
######, 1006
####, 589
###, 311
#######, 1587
####, 56
####, 77
    ##########, 344
##
    SQL, 30
####, 326
    ##, 327
    ###, 326
(## ##)
##, 66
    #####, 1152
    ######, 78
    #####, 1152
    #####, 341
####, 216, 217
    (## #######)
    #####, 635
##
    PostgreSQL#, xxxii
##
    ########, 330
    ###, 188
    ########, 333
    ###, 333
#####, 130
    ##, 589
##, 39
###, 188
    ############, 376
    #####, 1090
    ####, 37
    ####, 40
    ##, 36
    ##, 188
######, 395, 1093
####, 395, 1099
########, 1638
##
    numeric (####), 129
    #####, 130
##
    PL/pgSQL####, 1165
####
    SQL######, 1057
####, 865
###
    #####, 1116
####
    ##, 327
    ##, 327
##
    ##########, 696
#####, 1133
#####, 444
#####, 433, 436
##, 147
##, 326
####
    ##########, 459
#, 144
##, 148
############
    SPI########, 1293
    SPI#####, 1292, 1294
#######, 1082
#######, 2255

#####
    ########, 2559
######, 1133
###, 178
    #######, 182
    ##, 182
##, 10, 101
    ###, 101
    #, 102
    ##, 11, 101
    #, 102
    ##, 11
    ##, 102
    #######, 461
####, 697
    #####, 457, 459, 657
##, 19, 80
##########
    #########, 688
########, 666
##, 147
######
    #########, 463
(## serial)
####, 102
#, 2587
#, 7, 52
#######, 333
#########, 70
##, 173
    ########, 47
####
    ####, 2269
    ###, 2273
##
    ##, 49
###, 173, 1041
    ########, 47
    ###, 174
    ##, 333
##########, 177
##
    ####
        ##, 356
    ####, 531
    #####
        #########, 554
        ####, 554
###, 619
###, 38
##############
    ###, 1787
    #######, 585
###
    ### (## ###, ##)
###, 188
###, 188
###
    ###, 30
    ##, 30
####
    libpq##, 858
####, 679
##, 848
#####, 436
###, 327
#######, 858
######, 858
##########, 1469, 1750
    PL/pgSQL##, 1192
##, 251
##########
    ###, 1787
    #######, 585
########, 991
######, 1609
    (## #############)
    C########, 1117
    PL####, 1294
###########, 566
##, 165
    I/O, 172
    ####, 167
    #######, 46
    #######, 1089
    ##, 169
    ##, 166
    ##, 165
    ##, 171
##, 10, 114
##
    ######## (## #######, ##)
    #### (## ###, ##)
##, 188
    FROM###, 105
    RETURNS TABLE, 1055
    SETOF ##, 1052
    variadic, 1050
    ############, 379
    #####, 1043
        C######, 1060
        SQL#####, 1044
    ####, 49
    ##, 1060
    #######, 1049
    ######, 1045
    ######, 50
    ####, 40
    ##, 1042
    #########, 1051
    ####, 50
    ##, 373
####, 110
#########, 7
#######
    ##, 335

###, 115
###, 115
####, 115
###, 115
####, 12
    ######, 1086
    #####, 1081
    #####, 1083
    ####, 40
    ####, 1083
    ####, 1082
    ####, 322
    ####, 1085
    ####, 1085
#NULL##, 58
#########, 800, 833
#######, 798
####, 77
#######, 754
##
    ########, 129
    #####, 130
####, 465
#######, 1102
###, 338
######, 40
    ####, 327
####, 679

A

abbrev, 261
ABORT, 1351
abs, 194
ACL, 66
aclcontains, 343
acldefault, 343
aclexplode, 343
aclitem, 70
aclitemeq, 343
acos, 198
acosd, 198
acosh, 200
administration tools
    externally maintained, 2559
advisory lock, 442
age, 239, 352
aggregate function, 12
    built-in, 322
    invocation, 40
    moving aggregate, 1082
    ordered set, 1085
    partial aggregation, 1085
    polymorphic, 1083
    support functions for, 1086
    user-defined, 1081
    variadic, 1083
akeys, 2429
alias
    for table name in query, 11
    in the FROM clause, 104
    in the select list, 114
ALL, 331, 333
    GROUP BY ALL, 110
    SELECT ALL, 114
allow_alter_system configuration parameter, 594
allow_in_place_tablespaces configuration parameter, 597
allow_in_place_tablespaces#######, 597
allow_system_table_mods configuration parameter, 597
allow_system_table_mods#######, 597
ALTER AGGREGATE, 1352
ALTER COLLATION, 1354
ALTER CONVERSION, 1356
ALTER DATABASE, 1357
ALTER DEFAULT PRIVILEGES, 1359
ALTER DOMAIN, 1362
ALTER EVENT TRIGGER, 1365
ALTER EXTENSION, 1366
ALTER FOREIGN DATA WRAPPER, 1369
ALTER FOREIGN TABLE, 1371
ALTER FUNCTION, 1376
ALTER GROUP, 1379
ALTER INDEX, 1381
ALTER LANGUAGE, 1384
ALTER LARGE OBJECT, 1385
ALTER MATERIALIZED VIEW, 1386
ALTER OPERATOR, 1388
ALTER OPERATOR CLASS, 1390
ALTER OPERATOR FAMILY, 1391
ALTER POLICY, 1394
ALTER PROCEDURE, 1395
ALTER PUBLICATION, 1398
ALTER ROLE, 624, 1401
ALTER ROUTINE, 1405
ALTER RULE, 1406
ALTER SCHEMA, 1407
ALTER SEQUENCE, 1408
ALTER SERVER, 1411
ALTER STATISTICS, 1413
ALTER SUBSCRIPTION, 1414
ALTER SYSTEM, 1417
ALTER TABLE, 1419
ALTER TABLESPACE, 1436
ALTER TEXT SEARCH CONFIGURATION, 1438
ALTER TEXT SEARCH DICTIONARY, 1440
ALTER TEXT SEARCH PARSER, 1442
ALTER TEXT SEARCH TEMPLATE, 1443
ALTER TRIGGER, 1444
ALTER TYPE, 1446
ALTER USER, 1450
ALTER USER MAPPING, 1451
ALTER VIEW, 1452
ALTER_REPLICATION_SLOT, 2104
amcheck, 2360
ANALYZE, 657, 1454

AND (operator), 188
AND#####, 188
anonymous code blocks, 1638
any, 186
ANY, 325, 331, 333
anyarray, 186
anycompatible, 186
anycompatiblearray, 186
anycompatiblemultirange, 186
anycompatiblenonarray, 186
anycompatiblerange, 186
anyelement, 186
anyenum, 186
anymultirange, 186
anynonarray, 186
anyrange, 186
any_value, 322
application_name configuration parameter, 573
application_name#######, 573
arbitrary precision numbers, 128
Archive Modules, 1340
archive_cleanup_command configuration parameter, 554
archive_cleanup_command#######, 554
archive_command configuration parameter, 553
archive_command#######, 553
archive_library configuration parameter, 553
archive_mode configuration parameter, 553
archive_mode#######, 553
archive_timeout configuration parameter, 553
archive_timeout#######, 553
area, 256
armor, 2472
array, 165
    accessing, 167
    constant, 166
    constructor, 46
    declaration, 165
    I/O, 172
    modifying, 169
    of user-defined type, 1089
    searching, 171
ARRAY, 46
    determination of result type, 382
    ######, 382
array_agg, 322, 2435
array_append, 314
array_cat, 314
array_dims, 314
array_fill, 314
array_length, 314
array_lower, 314
array_ndims, 315
array_nulls configuration parameter, 593
array_nulls#######, 593
array_position, 315
array_positions, 315
array_prepend, 315
array_remove, 315
array_replace, 315
array_sample, 315
array_shuffle, 315
array_to_json, 284
array_to_string, 315
array_to_tsvector, 263
array_upper, 316
ascii, 203
asin, 198
asind, 198
asinh, 200
ASSERT
    in PL/pgSQL, 1194
    PL/pgSQL####, 1194
assertions
    in PL/pgSQL, 1194
asynchronous commit, 754
AT LOCAL, 248
AT TIME ZONE, 248
atan, 198
atan2, 198
atan2d, 198
atand, 198
atanh, 200
authentication_timeout configuration parameter, 537
authentication_timeout#######, 537
auth_delay, 2365
auth_delay.milliseconds configuration parameter, 2365
auth_delay.milliseconds#######, 2365
auto-increment (## serial)
autocommit
    bulk-loading data, 463
    psql, 1933
autosummarize storage parameter, 1525
autosummarize#######, 1525
autovacuum
    configuration parameters, 581
    general information, 662
    ####, 662
    #######, 582
autovacuum configuration parameter, 582
autovacuum_analyze_scale_factor
    configuration parameter, 582
    storage parameter, 1588
    #######, 1588
    #######, 582
autovacuum_analyze_threshold
    configuration parameter, 582
    storage parameter, 1588
    #######, 1588
    #######, 582
autovacuum_enabled storage parameter, 1587
autovacuum_enabled#######, 1587
autovacuum_freeze_max_age
    configuration parameter, 583
    storage parameter, 1588
    #######, 1588

#######, 583
autovacuum_freeze_min_age storage parameter, 1588
autovacuum_freeze_min_age#######, 1588
autovacuum_freeze_table_age storage parameter, 1589
autovacuum_freeze_table_age#######, 1589
autovacuum_max_workers configuration parameter, 582
autovacuum_max_workers#######, 582
autovacuum_multixact_freeze_max_age
configuration parameter, 583
storage parameter, 1589
#######, 1589
#######, 583
autovacuum_multixact_freeze_min_age storage parameter, 1589
autovacuum_multixact_freeze_min_age#######, 1589
autovacuum_multixact_freeze_table_age storage parameter, 1589
autovacuum_multixact_freeze_table_age#######, 1589
autovacuum_naptime configuration parameter, 582
autovacuum_naptime#######, 582
autovacuum_vacuum_cost_delay
configuration parameter, 583
storage parameter, 1588
#######, 1588
#######, 583
autovacuum_vacuum_cost_limit
configuration parameter, 583
storage parameter, 1588
#######, 1588
#######, 583
autovacuum_vacuum_insert_scale_factor
configuration parameter, 582
storage parameter, 1588
#######, 1588
#######, 582
autovacuum_vacuum_insert_threshold
configuration parameter, 582
storage parameter, 1588
#######, 1588
#######, 582
autovacuum_vacuum_scale_factor
configuration parameter, 582
storage parameter, 1588
#######, 1588
#######, 582
autovacuum_vacuum_threshold
configuration parameter, 582
storage parameter, 1588
#######, 1588
#######, 582
autovacuum_work_mem configuration parameter, 542
autovacuum_work_mem#######, 542
autovacuum#######, 582
auto_explain, 2366
auto_explain.log_analyze configuration parameter, 2366
auto_explain.log_analyze#######, 2366
auto_explain.log_buffers configuration parameter, 2366
auto_explain.log_buffers#######, 2366
auto_explain.log_format configuration parameter, 2367
auto_explain.log_format#######, 2367
auto_explain.log_level configuration parameter, 2367
auto_explain.log_level#######, 2367
auto_explain.log_min_duration configuration parameter, 2366
auto_explain.log_min_duration#######, 2366
auto_explain.log_nested_statements configuration parameter, 2367
auto_explain.log_nested_statements#######, 2367
auto_explain.log_parameter_max_length configuration parameter, 2366
auto_explain.log_parameter_max_length#######, 2366
auto_explain.log_settings configuration parameter, 2367
auto_explain.log_settings#######, 2367
auto_explain.log_timing configuration parameter, 2366
auto_explain.log_timing#######, 2366
auto_explain.log_triggers configuration parameter, 2367
auto_explain.log_triggers#######, 2367
auto_explain.log_verbose configuration parameter, 2367
auto_explain.log_verbose#######, 2367
auto_explain.log_wal configuration parameter, 2366
auto_explain.log_wal#######, 2366
auto_explain.sample_rate configuration parameter, 2367
auto_explain.sample_rate#######, 2367
avals, 2429
average, 322
avg, 322

B

B-Tree (## index) (## ######)
backend_flush_after configuration parameter, 546
backend_flush_after#######, 546
Background workers, 1320
backslash escapes, 32
backslash_quote configuration parameter, 593
backslash_quote#######, 593
backtrace_functions configuration parameter, 597
backtrace_functions#######, 597
backup, 358, 666
Backup Manifest, 2278
base type, 1041
base64 format, 212
basebackup_to_shell, 2369
basebackup_to_shell.command configuration parameter, 2369

basebackup_to_shell.command#######, 2369
basebackup_to_shell.required_role configuration parameter, 2369
basebackup_to_shell.required_role#######, 2369
BASE_BACKUP, 2108
basic_archive, 2370
basic_archive.archive_directory configuration parameter, 2370
basic_archive.archive_directory#######, 2370
batch mode
  in libpq, 838
BEGIN, 1457
BETWEEN, 191
BETWEEN SYMMETRIC, 191
BGWORKER_BACKEND_DATABASE_CONNECTION, 1320
BGWORKER_SHMEM_ACCESS, 1320
bgwriter_delay configuration parameter, 545
bgwriter_delay#######, 545
bgwriter_flush_after configuration parameter, 546
bgwriter_flush_after#######, 546
bgwriter_lru_maxpages configuration parameter, 546
bgwriter_lru_maxpages#######, 546
bgwriter_lru_multiplier configuration parameter, 546
bgwriter_lru_multiplier#######, 546
bigint, 35, 128
bigserial, 131
binary data, 134
  functions, 209
binary string
  concatenation, 209
  converting to character string, 212
  length, 210
bison, 479
bit string
  constant, 34
  data type, 151
  length, 213
bit strings
  functions, 212
bitmap scan, 389, 563
bit_and, 322
bit_count, 210, 213
bit_length, 201, 209, 213
bit_or, 323
bit_xor, 323
BLOB (## large object) (## #########)
block_size configuration parameter, 595
block_size#######, 595
bloom, 2371
bonjour configuration parameter, 536
bonjour_name configuration parameter, 537
bonjour_name#######, 537
bonjour#######, 536
Boolean
  data type, 144
  operators (## operators, logical)
  ####, 144
bool_and, 323
bool_or, 323
bound_box, 258
box, 257
box (data type), 148
box######, 148
bpchar, 132
BRIN (## index) (## ######)
brin_desummarize_range, 367
brin_metapage_info, 2458
brin_page_items, 2458
brin_page_type, 2457
brin_revmap_data, 2458
brin_summarize_new_values, 367
brin_summarize_range, 367
broadcast, 261
BSD Authentication, 620
btree_gin, 2375
btree_gist, 2376
btrim, 200, 209
bt_index_check, 2360
bt_index_parent_check, 2361
bt_metap, 2455
bt_multi_page_stats, 2455
bt_page_items, 2456, 2457
bt_page_stats, 2455
buffering storage parameter, 1525
buffering#######, 1525
bytea, 134
bytea_output configuration parameter, 588
bytea_output#######, 588

C

C, 796, 894
C++, 1080
CALL, 1459
canceling SQL queries, 843
cardinality, 316
CASCADE
  DROP#, 94
  with DROP, 94
  foreign key action, 61
  ######, 61
Cascading Replication, 679
CASE, 311
  determination of result type, 382
  ######, 382
case sensitivity
  of SQL commands, 30
cast
  I/O conversion, 1487
cbrt, 194
ceil, 194
ceiling, 194
center, 256
Certificate, 619
chained transactions, 1469, 1750
  in PL/pgSQL, 1192

char, 132
character, 132
character set, 590, 596, 646
character string
  concatenation, 200
  constant, 31
  converting to binary string, 212
  data types, 132
  length, 201
  prefix test, 202
character varying, 132
character_length, 201
char_length, 201
check constraint, 56
CHECK OPTION, 1625
checkpoint, 755
CHECKPOINT, 1460
checkpoint_completion_target configuration parameter, 552
checkpoint_completion_target#######, 552
checkpoint_flush_after configuration parameter, 552
checkpoint_flush_after#######, 552
checkpoint_timeout configuration parameter, 552
checkpoint_timeout#######, 552
checkpoint_warning configuration parameter, 552
checkpoint_warning#######, 552
checksums, 753
check_function_bodies configuration parameter, 585
check_function_bodies#######, 585
chr, 203
cid, 184
cidr, 149
circle, 149, 258
citext, 2378
client authentication, 603
  timeout during, 537
  ########, 537
client_connection_check_interval configuration parameter, 537
client_connection_check_interval#######, 537
client_encoding configuration parameter, 590
client_encoding#######, 590
client_min_messages configuration parameter, 583
client_min_messages#######, 583
clock_timestamp, 239
CLOSE, 1461
cluster
  of databases (## database cluster)
CLUSTER, 1462
clusterdb, 1809
clustering, 679
cluster_name configuration parameter, 580
cluster_name#######, 580
cmax, 63
cmin, 63
COALESCE, 312
COLLATE, 45
collation, 638
  in PL/pgSQL, 1165
  in SQL functions, 1057
COLLATION FOR, 347
color, 2587
column, 7, 52
  adding, 64
  removing, 65
  renaming, 66
  system column, 63
column data type
  changing, 66
column reference, 38
col_description, 350
comment
  about database objects, 350
  in SQL, 36
COMMENT, 1465
COMMIT, 1469
COMMIT PREPARED, 1470
commit_delay configuration parameter, 552
commit_delay#######, 552
commit_siblings configuration parameter, 552
commit_siblings#######, 552
commit_timestamp_buffers configuration parameter, 543
common table expression (## WITH)
comparison
  composite type, 333
  operators, 188
  row constructor, 333
  subquery result row, 330
compiling
  libpq applications, 872
composite type, 173, 1041
  comparison, 333
  constant, 174
  constructor, 47
computed field, 177
compute_query_id configuration parameter, 581
compute_query_id#######, 581
concat, 203
concat_ws, 203
concurrency, 433
conditional expression, 311
configuration
  of recovery
    general settings, 554
    of a standby server, 554
  of the server, 531
  of the server
    functions, 356
configure, 481
configure environment variables, 490
configure options, 483
configure#####, 483
configure####, 490
config_file configuration parameter, 534
config_file#######, 534

conjunction, 188
connectby, 2530, 2536
connection service file, 867
conninfo, 803
constant, 31
constraint, 56
    adding, 65
    check, 56
    exclusion, 63
    foreign key, 60
    name, 56
    NOT NULL, 58
    primary key, 59
    removing, 65
    unique, 58
constraint exclusion, 93, 567
constraint_exclusion configuration parameter, 567
constraint_exclusion#######, 567
container type, 1041
CONTINUE
    in PL/pgSQL, 1180
    PL/pgSQL####, 1180
continuous archiving, 666
    in standby, 688
control file, 1104
convert, 212
convert_from, 212
convert_to, 212
COPY, 8, 1471
    libpq#####, 849
    with libpq, 849
corr, 326
correlation, 326
    in the query planner, 459
cos, 199
cosd, 199
cosh, 199
cot, 199
cotd, 199
count, 323
covariance
    population, 326
    sample, 326
covar_pop, 326
covar_samp, 326
covering index, 393
cpu_index_tuple_cost configuration parameter, 565
cpu_index_tuple_cost#######, 565
cpu_operator_cost configuration parameter, 565
cpu_operator_cost#######, 565
cpu_tuple_cost configuration parameter, 565
cpu_tuple_cost#######, 565
CREATE ACCESS METHOD, 1480
CREATE AGGREGATE, 1481
CREATE CAST, 1487
CREATE COLLATION, 1491
CREATE CONVERSION, 1494
CREATE DATABASE, 629, 1496
CREATE DOMAIN, 1500
CREATE EVENT TRIGGER, 1503
CREATE EXTENSION, 1505
CREATE FOREIGN DATA WRAPPER, 1507
CREATE FOREIGN TABLE, 1509
CREATE FUNCTION, 1513
CREATE GROUP, 1521
CREATE INDEX, 1522
CREATE LANGUAGE, 1530
CREATE MATERIALIZED VIEW, 1532
CREATE OPERATOR, 1534
CREATE OPERATOR CLASS, 1537
CREATE OPERATOR FAMILY, 1540
CREATE POLICY, 1541
CREATE PROCEDURE, 1546
CREATE PUBLICATION, 1549
CREATE ROLE, 622, 1553
CREATE RULE, 1557
CREATE SCHEMA, 1560
CREATE SEQUENCE, 1562
CREATE SERVER, 1566
CREATE STATISTICS, 1568
CREATE SUBSCRIPTION, 1572
CREATE TABLE, 7, 1577
CREATE TABLE AS, 1596
CREATE TABLESPACE, 632, 1599
CREATE TEXT SEARCH CONFIGURATION, 1601
CREATE TEXT SEARCH DICTIONARY, 1602
CREATE TEXT SEARCH PARSER, 1604
CREATE TEXT SEARCH TEMPLATE, 1606
CREATE TRANSFORM, 1607
CREATE TRIGGER, 1609
CREATE TYPE, 1615
CREATE USER, 1623
CREATE USER MAPPING, 1624
CREATE VIEW, 1625
createdb, 3, 630, 1812
createrole_self_grant
    configuration parameter, 588
    #######, 588
createrole_self_grant configuration parameter
    use in securing functions, 1519
createrole_self_grant#######
    ############, 1519
createuser, 622, 1815
CREATE_REPLICATION_SLOT, 2103
cross compilation, 488, 498
cross join, 101
crosstab, 2531, 2532, 2533
crypt, 2469
cstring, 186
CSV (Comma-Separated Values) format
    in psql, 1927
CSV(#######)##
    psql##, 1927
ctid, 64
CTID, 1139
CUBE, 110

cube (extension), 2381
cube (##), 2381
cume_dist, 329
    hypothetical, 328
    ###, 328
current_catalog, 339
current_database, 339
current_date, 239
current_logfiles
    and the log_destination configuration parameter, 568
    and the pg_current_logfile function, 340
    ###log_destination#######, 568
    #pg_current_logfile##, 340
current_query, 339
current_role, 339
current_schema, 339
current_schemas, 339
current_setting, 356
current_time, 239
current_timestamp, 239
current_user, 339
currval, 310
cursor
    CLOSE, 1461
    DECLARE, 1631
    FETCH, 1698
    in PL/pgSQL, 1187
    MOVE, 1727
    showing the query plan, 1692
cursor_tuple_fraction configuration parameter, 568
cursor_tuple_fraction#######, 568
custom scan provider
    handler for, 2177
Cygwin
    installation on, 504
    #########, 504

D

daitch_mokotoff, 2423
data area (## database cluster)
data partitioning, 679
data type, 126
    base, 1041
    category, 375
    composite, 1041
    constant, 35
    container, 1041
    conversion, 375
    domain, 183
    enumerated (enum), 145
    internal organization, 1061
    numeric, 127
    polymorphic, 1042
    type cast, 44
    user-defined, 1086
database, 629
    creating, 3
database activity
    monitoring, 696
database cluster, 7, 510
data_checksums configuration parameter, 595
data_checksums#######, 595
data_directory configuration parameter, 534
data_directory_mode configuration parameter, 596
data_directory_mode#######, 596
data_directory#######, 534
data_sync_retry configuration parameter, 595
data_sync_retry#######, 595
date, 136, 137
    constants, 139
    current, 250
    output format, 140
        (## formatting)
    ####, 140
        (## ####)
    ##, 139
DateStyle configuration parameter, 589
DateStyle#######, 589
date_add, 239
date_bin, 248
date_part, 239, 243
date_subtract, 240
date_trunc, 240, 247
dblink, 2385, 2390
dblink_build_sql_delete, 2412
dblink_build_sql_insert, 2410
dblink_build_sql_update, 2413
dblink_cancel_query, 2408
dblink_close, 2399
dblink_connect, 2386
dblink_connect_u, 2388
dblink_disconnect, 2389
dblink_error_message, 2402
dblink_exec, 2393
dblink_fetch, 2397
dblink_get_connections, 2401
dblink_get_notify, 2405
dblink_get_pkey, 2409
dblink_get_result, 2406
dblink_is_busy, 2404
dblink_open, 2395
dblink_send_query, 2403
deadlock, 441
    timeout during, 592
deadlock_timeout configuration parameter, 592
deadlock_timeout#######, 592
DEALLOCATE, 1630
dearmor, 2472
debug_assertions configuration parameter, 596
debug_assertions#######, 596
debug_deadlocks configuration parameter, 599
debug_deadlocks#######, 599
debug_discard_caches configuration parameter, 598
debug_discard_caches#######, 598
debug_io_direct configuration parameter, 598

debug_io_direct#######, 598
debug_logical_replication_streaming configuration parameter, 601
debug_logical_replication_streaming#######, 601
debug_parallel_query configuration parameter, 598
debug_parallel_query#######, 598
debug_pretty_print configuration parameter, 574
debug_pretty_print#######, 574
debug_print_parse configuration parameter, 573
debug_print_parse#######, 573
debug_print_plan configuration parameter, 573
debug_print_plan#######, 573
debug_print_rewritten configuration parameter, 573
debug_print_rewritten#######, 573
decimal (## numeric)
DECLARE, 1631
decode, 212
decode_bytea
    in PL/Perl, 1235
    PL/Perl####, 1235
decrypt, 2475
decrypt_iv, 2475
deduplicate_items storage parameter, 1525
deduplicate_items#######, 1525
default value, 53
    changing, 65
default-roles, 2588
default_statistics_target configuration parameter, 567
default_statistics_target#######, 567
default_tablespace configuration parameter, 584
default_tablespace#######, 584
default_table_access_method configuration parameter, 584
default_table_access_method#######, 584
default_text_search_config configuration parameter, 590
default_text_search_config#######, 590
default_toast_compression configuration parameter, 584
default_toast_compression#######, 584
default_transaction_deferrable configuration parameter, 585
default_transaction_deferrable #######, 585
default_transaction_isolation configuration parameter, 585
default_transaction_isolation#######, 585
default_transaction_read_only configuration parameter, 585
default_transaction_read_only#######, 585
deferrable transaction, 585
    setting, 1787
    setting default, 585
defined, 2430
degrees, 194
delay, 251
DELETE, 14, 98, 1634
    RETURNING, 99
delete, 2430
deleting, 98
dense_rank, 329
    hypothetical, 328
    ###, 328
diagonal, 256
diameter, 256
dict_int, 2415
dict_xsyn, 2416
difference, 2423
digest, 2468
DISCARD, 1637
disjunction, 188
disk space, 656
disk usage, 749
DISTINCT, 10
    GROUP BY DISTINCT, 110
    SELECT DISTINCT, 114
div, 195
dmetaphone, 2426
dmetaphone_alt, 2426
DO, 1638
document
    text search, 398
dollar quoting, 33
domain, 183
double precision, 130
DROP ACCESS METHOD, 1640
DROP AGGREGATE, 1641
DROP CAST, 1643
DROP COLLATION, 1644
DROP CONVERSION, 1645
DROP DATABASE, 631, 1646
DROP DOMAIN, 1647
DROP EVENT TRIGGER, 1648
DROP EXTENSION, 1649
DROP FOREIGN DATA WRAPPER, 1650
DROP FOREIGN TABLE, 1651
DROP FUNCTION, 1652
DROP GROUP, 1654
DROP INDEX, 1655
DROP LANGUAGE, 1657
DROP MATERIALIZED VIEW, 1658
DROP OPERATOR, 1659
DROP OPERATOR CLASS, 1661
DROP OPERATOR FAMILY, 1663
DROP OWNED, 1664
DROP POLICY, 1665
DROP PROCEDURE, 1666
DROP PUBLICATION, 1668
DROP ROLE, 622, 1669
DROP ROUTINE, 1670
DROP RULE, 1671
DROP SCHEMA, 1672
DROP SEQUENCE, 1673
DROP SERVER, 1674
DROP STATISTICS, 1675
DROP SUBSCRIPTION, 1676
DROP TABLE, 8, 1678

DROP TABLESPACE, 1679
DROP TEXT SEARCH CONFIGURATION, 1680
DROP TEXT SEARCH DICTIONARY, 1681
DROP TEXT SEARCH PARSER, 1682
DROP TEXT SEARCH TEMPLATE, 1683
DROP TRANSFORM, 1684
DROP TRIGGER, 1685
DROP TYPE, 1686
DROP USER, 1687
DROP USER MAPPING, 1688
DROP VIEW, 1689
dropdb, 632, 1819
dropuser, 622, 1822
DROP_REPLICATION_SLOT, 2107
DTD, 155
DTrace, 489, 500, 742
duplicate, 10
duplicates, 114
dynamic loading, 592, 1060
dynamic_library_path, 1060
dynamic_library_path configuration parameter, 592
dynamic_library_path#######, 592
dynamic_shared_memory_type configuration parameter, 544
dynamic_shared_memory_type#######, 544

E

each, 2430
earth, 2418
earthdistance, 2418
earth_box, 2419
earth_distance, 2419
ECPG, 894
ecpg, 1824
effective_cache_size configuration parameter, 566
effective_cache_size#######, 566
effective_io_concurrency configuration parameter, 546
effective_io_concurrency#######, 546
elog, 2140
  in PL/Perl, 1235
  in PL/Python, 1252
  in PL/Tcl, 1221
  PL/Perl####, 1235
  PL/Python####, 1252
  PL/Tcl####, 1221
embedded SQL
  in C, 894
enable_async_append configuration parameter, 563
enable_async_append#######, 563
enable_bitmapscan configuration parameter, 563
enable_bitmapscan#######, 563
enable_gathermerge configuration parameter, 563
enable_gathermerge#######, 563
enable_group_by_reordering configuration parameter, 563
enable_hashagg configuration parameter, 563
enable_hashagg#######, 563
enable_hashjoin configuration parameter, 563
enable_hashjoin#######, 563
enable_incremental_sort configuration parameter, 563
enable_incremental_sort#######, 563
enable_indexonlyscan configuration parameter, 563
enable_indexonlyscan#######, 563
enable_indexscan configuration parameter, 563
enable_indexscan#######, 563
enable_material configuration parameter, 563
enable_material#######, 563
enable_memoize configuration parameter, 563
enable_memoize#######, 563
enable_mergejoin configuration parameter, 563
enable_mergejoin#######, 563
enable_nestloop configuration parameter, 564
enable_nestloop#######, 564
enable_parallel_append configuration parameter, 564
enable_parallel_append#######, 564
enable_parallel_hash configuration parameter, 564
enable_parallel_hash#######, 564
enable_partitionwise_aggregate configuration parameter, 564
enable_partitionwise_aggregate#######, 564
enable_partitionwise_join configuration parameter, 564
enable_partitionwise_join#######, 564
enable_partition_pruning configuration parameter, 564
enable_partition_pruning#######, 564
enable_presorted_aggregate configuration parameter, 564
enable_presorted_aggregate#######, 564
enable_seqscan configuration parameter, 564
enable_seqscan#######, 564
enable_sort configuration parameter, 564
enable_sort#######, 564
enable_tidscan configuration parameter, 564
enable_tidscan#######, 564
encode, 212
encode_array_constructor
  in PL/Perl, 1235
  PL/Perl####, 1235
encode_array_literal
  in PL/Perl, 1235
  PL/Perl####, 1235
encode_bytea
  in PL/Perl, 1235
  PL/Perl####, 1235
encode_typed_literal
  in PL/Perl, 1235
  PL/Perl####, 1235
encrypt, 2475
encryption, 525
  for specific columns, 2468
encrypt_iv, 2475
END, 1690
enumerated types, 145
enum_first, 252
enum_last, 252
enum_range, 252

environment variable, 865
ephemeral named relation
    registering with SPI, 1292, 1294
    unregistering from SPI, 1293
ereport, 2140
erf, 195
erfc, 195
error codes
    libpq, 825
    list of, 2286
error message
    in PGcancelConn, 846
    in PGconn, 816
escape format, 212
escape string syntax, 32
escape_string_warning configuration parameter, 593
escape_string_warning#######, 593
escaping strings
    in libpq, 831
event log
    event log, 530
event trigger, 1124
    in C, 1128
    in PL/Tcl, 1223
event_source configuration parameter, 571
event_source#######, 571
event_trigger, 186
event_triggers
    configuration parameter, 589
every, 323
EXCEPT, 115
exceptions
    in PL/pgSQL, 1183
    in PL/Tcl, 1223
exclusion constraint, 63
EXECUTE, 1691
exist, 2430
EXISTS, 330
EXIT
    in PL/pgSQL, 1179
    PL/pgSQL####, 1179
exit_on_error configuration parameter, 595
exit_on_error #######, 595
exp, 195
EXPLAIN, 446, 1692
expression
    order of evaluation, 48
    syntax, 38
extending SQL, 1041
extension, 1104
    externally maintained, 2559
external_pid_file configuration parameter, 535
external_pid_file#######, 535
extract, 240, 243
extra_float_digits configuration parameter, 589
extra_float_digits#######, 589

F

factorial, 195
failover, 679
false, 144
family, 261
fast path, 848
fastupdate storage parameter, 1525
fastupdate#######, 1525
fdw_handler, 186
FETCH, 1698
field
    computed, 177
field selection, 39
file system mount points, 511
file_fdw, 2420
fillfactor storage parameter, 1524, 1587
fillfactor#######, 1524, 1587
FILTER, 40
first_value, 329
flex, 479
float4 (## real)
float8 (## double precision)
floating point, 130
floating-point
    display, 589
floor, 195
foreign data, 94
foreign data wrapper
    handler for, 2156
foreign key, 15, 60
    self-referential, 61
foreign table, 94
format, 203, 207
    PL/pgSQL####, 1171
    use in PL/pgSQL, 1171
formatting, 230
format_type, 345
Free Space Map, 2255
FreeBSD
    IPC configuration, 516
    IPC##, 516
    shared library, 1067
    ########, 513
    #######, 1067
from_collapse_limit configuration parameter, 568
from_collapse_limit#######, 568
FSM (## Free Space Map) (## #######)
fsm_page_contents, 2453
fsync configuration parameter, 548
fsync#######, 548
full text search, 398
    data types, 152
    functions and operators, 152
full_page_writes configuration parameter, 550
full_page_writes#######, 550
function, 188
    default values for arguments, 1051

in the FROM clause, 105
internal, 1060
invocation, 40
mixed notation, 50
named argument, 1045
named notation, 50
output parameter, 1049
polymorphic, 1042
positional notation, 49
RETURNS TABLE, 1055
statistics, 373
type resolution in an invocation, 379
user-defined, 1043
in C, 1060
in SQL, 1044
variadic, 1050
with SETOF, 1052
functional dependency, 110
fuzzystrmatch, 2423

G

gcd, 195
gc_to_sec, 2418
generated column, 54, 1511, 1584
generate_series, 335
generate_subscripts, 337
genetic query optimization, 566
gen_random_bytes, 2476
gen_random_uuid, 268, 2476
gen_salt, 2469
GEQO (## genetic query optimization) (## ######## ###)
geqo configuration parameter, 566
geqo_effort configuration parameter, 567
geqo_effort#######, 567
geqo_generations configuration parameter, 567
geqo_generations#######, 567
geqo_pool_size configuration parameter, 567
geqo_pool_size#######, 567
geqo_seed configuration parameter, 567
geqo_seed#######, 567
geqo_selection_bias configuration parameter, 567
geqo_selection_bias#######, 567
geqo_threshold configuration parameter, 566
geqo_threshold#######, 566
geqo#######, 566
get_bit, 210, 214
get_byte, 210
get_current_ts_config, 263
get_raw_page, 2453
GIN (## index) (## ######)
gin_clean_pending_list, 367
gin_fuzzy_search_limit configuration parameter, 592
gin_fuzzy_search_limit#######, 592
gin_leafpage_items, 2459
gin_metapage_info, 2458
gin_page_opaque_info, 2459
gin_pending_list_limit
configuration parameter, 588
storage parameter, 1525
#######, 1525
#######, 588
GiST (## index) (## ######)
gist_page_items, 2460
gist_page_items_bytea, 2460
gist_page_opaque_info, 2459
global data
in PL/Tcl, 1219
GRANT, 66, 1702
GREATEST, 313
determination of result type, 382
######, 382
Gregorian calendar, 2299
GROUP BY, 12, 109
grouping, 109
GROUPING, 328
GROUPING SETS, 110
gssapi, 529
GSSAPI, 613
libpq##, 808
gss_accept_delegation configuration parameter, 538
gss_accept_delegation#######, 538
GUID, 154

H

hash (## index)
hash_bitmap_info, 2461
hash_mem_multiplier configuration parameter, 542
hash_mem_multiplier#######, 542
hash_metapage_info, 2461
hash_page_items, 2461
hash_page_stats, 2461
hash_page_type, 2460
has_any_column_privilege, 341
has_column_privilege, 341
has_database_privilege, 341
has_foreign_data_wrapper_privilege, 341
has_function_privilege, 341
has_language_privilege, 342
has_parameter_privilege, 342
has_schema_privilege, 342
has_sequence_privilege, 342
has_server_privilege, 342
has_tablespace_privilege, 342
has_table_privilege, 342
has_type_privilege, 342
HAVING, 12, 110
hba_file configuration parameter, 534
hba_file#######, 534
heap_page_items, 2454
heap_page_item_attrs, 2454
heap_tuple_infomask_flags, 2454
height, 256
hex format, 212
hierarchical database, 7
high availability, 679

history
  of PostgreSQL, xxxii
hmac, 2468
host, 261
hostmask, 261
hot standby, 679
hot_standby configuration parameter, 560
hot_standby_feedback configuration parameter, 561
hot_standby_feedback #######, 561
hot_standby#######, 560
hstore, 2427, 2429
hstore_to_array, 2429
hstore_to_json, 2429
hstore_to_jsonb, 2430
hstore_to_jsonb_loose, 2430
hstore_to_json_loose, 2430
hstore_to_matrix, 2429
huge_pages configuration parameter, 541
huge_pages_status configuration parameter, 596
huge_pages#######, 541
huge_page_size configuration parameter, 541
huge_page_size#######, 541
hypothetical-set aggregate
  built-in, 327

I

icount, 2437
ICU, 487, 487, 495, 636, 640, 1491, 1492, 1498, 1498
icu_unicode_version, 355
icu_validation_level configuration parameter, 590
icu_validation_level#######, 590
ident, 615
identifier
  length, 30
  syntax of, 30
IDENTIFY_SYSTEM, 2102
identity column, 53
ident_file configuration parameter, 534
ident_file#######, 534
idle_in_transaction_session_timeout configuration parameter, 587
idle_in_transaction_session_timeout#######, 587
idle_session_timeout configuration parameter, 587
idle_session_timeout#######, 587
idx, 2437
IFNULL, 312
ignore_checksum_failure configuration parameter, 600
ignore_checksum_failure#######, 600
ignore_invalid_pages configuration parameter, 600
ignore_invalid_pages#######, 600
ignore_system_indexes configuration parameter, 598
ignore_system_indexes#######, 598
IMMUTABLE, 1058
IMPORT FOREIGN SCHEMA, 1707
IN, 331, 333
INCLUDE
  in index definitions, 393
  #########, 393
include
  in configuration file, 533
  ##########, 533
include_dir
  in configuration file, 533
  ##########, 533
include_if_exists
  in configuration file, 533
  ##########, 533
index, 385, 2449
  and ORDER BY, 388
  B-Tree, 386, 2203
  BRIN, 387, 2238
  building concurrently, 1525
  combining multiple indexes, 389
  covering, 393
  examining usage, 396
  on expressions, 390
  for user-defined data type, 1093
  GIN, 387, 2233
    text search, 428
  GiST, 386, 2208
    text search, 428
  hash, 386
  Hash, 2250
  index-only scans, 393
  locks, 444
  multicolumn, 387
  partial, 390
  rebuilding concurrently, 1740
  SP-GiST, 387, 2223
  unique, 389
Index Access Method, 2186
index scan, 563
index-only scan, 393
indexam
  Index Access Method, 2186
  ##############, 2186
index_am_handler, 186
inet (data type), 149
inet_client_addr, 339
inet_client_port, 339
inet_merge, 261
inet_same_family, 261
inet_server_addr, 339
inet_server_port, 339
inet######, 149
infinity
  floating point, 130
  numeric (data type), 129
information schema, 990
inheritance, 19, 80
initcap, 203
initdb, 510, 1954
Initialization Fork, 2256
initplan, 452
input function, 1086
INSERT, 8, 97, 1709

RETURNING, 99
inserting, 97
installation, 479
binaries, 478
instr function, 1214
instr##, 1214
int2 (## smallint)
int4 (## integer)
int8 (## bigint)
intagg, 2435
intarray, 2437
integer, 35, 128
integer_datetimes configuration parameter, 596
integer_datetimes#######, 596
interfaces
externally maintained, 2559
internal, 186
INTERSECT, 115
interval, 136, 142
output format, 143
(## formatting)
IntervalStyle configuration parameter, 589
IntervalStyle#######, 589
intset, 2437
int_array_aggregate, 2435
int_array_enum, 2435
inverse distribution, 327
in_hot_standby configuration parameter, 596
in_hot_standby#######, 596
in_range support functions, 2204
in_range######, 2204
io_combine_limit configuration parameter, 547
IS DISTINCT FROM, 191, 333
IS DOCUMENT, 272
IS FALSE, 192
IS JSON, 285
IS NOT DISTINCT FROM, 191, 333
IS NOT DOCUMENT, 272
IS NOT FALSE, 192
IS NOT NULL, 191
IS NOT TRUE, 192
IS NOT UNKNOWN, 192
IS NULL, 191, 594
IS TRUE, 192
IS UNKNOWN, 192
isclosed, 256
isempty, 320, 321
isfinite, 240
isn, 2440
ISNULL, 191
isn_weak, 2441
isopen, 256
is_array_ref
in PL/Perl, 1235
PL/Perl####, 1235
is_valid, 2441

J

JIT, 780
jit configuration parameter, 568
jit_above_cost configuration parameter, 566
jit_above_cost#######, 566
jit_debugging_support configuration parameter, 600
jit_debugging_support#######, 600
jit_dump_bitcode configuration parameter, 600
jit_dump_bitcode#######, 600
jit_expressions configuration parameter, 601
jit_expressions#######, 601
jit_inline_above_cost configuration parameter, 566
jit_inline_above_cost#######, 566
jit_optimize_above_cost configuration parameter, 566
jit_optimize_above_cost#######, 566
jit_profiling_support configuration parameter, 601
jit_profiling_support#######, 601
jit_provider configuration parameter, 591
jit_provider#######, 591
jit_tuple_deforming configuration parameter, 601
jit_tuple_deforming#######, 601
jit#######, 568
join, 10, 101
controlling the order, 461
cross, 101
left, 102
natural, 102
outer, 11, 101
right, 102
self, 11
join_collapse_limit configuration parameter, 568
join_collapse_limit#######, 568
JSON, 156
functions and operators, 281
######, 281
json constructor, 285
JSONB, 156
jsonb
containment, 158
existence, 158
indexes on, 160
########, 160
##, 158
##, 158
jsonb_agg, 323
jsonb_agg_strict, 324
jsonb_array_elements, 286
jsonb_array_elements_text, 286
jsonb_array_length, 287
jsonb_build_array, 284
jsonb_build_object, 284
jsonb_each, 287
jsonb_each_text, 287
jsonb_extract_path, 287
jsonb_extract_path_text, 287
jsonb_insert, 290
jsonb_object, 285

jsonb_object_agg, 323
jsonb_object_agg_strict, 324
jsonb_object_agg_unique, 324
jsonb_object_agg_unique_strict, 324
jsonb_object_keys, 288
jsonb_path_exists, 291
jsonb_path_exists_tz, 292
jsonb_path_match, 291
jsonb_path_match_tz, 292
jsonb_path_query, 291
jsonb_path_query_array, 291
jsonb_path_query_array_tz, 292
jsonb_path_query_first, 291
jsonb_path_query_first_tz, 292
jsonb_path_query_tz, 292
jsonb_populate_record, 288
jsonb_populate_recordset, 289
jsonb_populate_record_valid, 288
jsonb_pretty, 292
jsonb_set, 290
jsonb_set_lax, 290
jsonb_strip_nulls, 290
jsonb_to_record, 289
jsonb_to_recordset, 290
jsonb_to_tsvector, 265
jsonb_typeof, 292
jsonpath, 164
json_agg, 323
json_agg_strict, 324
json_array, 284
json_arrayagg, 324
json_array_elements, 286
json_array_elements_text, 286
json_array_length, 287
json_build_array, 284
json_build_object, 284
json_each, 287
json_each_text, 287
json_extract_path, 287
json_extract_path_text, 287
json_object, 284, 285
json_objectagg, 323
json_object_agg, 323
json_object_agg_strict, 323
json_object_agg_unique, 324
json_object_agg_unique_strict, 324
json_object_keys, 287
json_populate_record, 288
json_populate_recordset, 289
json_scalar, 285
json_strip_nulls, 290
json_table, 305
json_to_record, 289
json_to_recordset, 289
json_to_tsvector, 265
json_typeof, 292
JSON_#####, 303
JSON_###, 303
JSON##, 304
Julian date, 2299
Just-In-Time compilation (## JIT)
justify_days, 240
justify_hours, 240
justify_interval, 241

K

key word
    list of, 2301
    syntax of, 30
krb_caseins_users configuration parameter, 538
krb_caseins_users#######, 538
krb_server_keyfile configuration parameter, 538
krb_server_keyfile#######, 538

L

label (## alias)
lag, 329
language_handler, 186
large object, 884
lastval, 310
last_value, 329
LATERAL
    FROM###, 107
    in the FROM clause, 107
latitude, 2418
lca, 2449
lcm, 195
lc_messages configuration parameter, 590
lc_messages#######, 590
lc_monetary configuration parameter, 590
lc_monetary#######, 590
lc_numeric configuration parameter, 590
lc_numeric#######, 590
lc_time configuration parameter, 590
lc_time#######, 590
LDAP, 486, 496, 616
LDAP connection parameter lookup, 868
LDAP############, 868
ldconfig, 503
lead, 329
LEAST, 313
    determination of result type, 382
    ######, 382
left, 203
left join, 102
length, 203, 210, 213, 257, 264
    of a binary string (## binary strings, length)
    of a character string (## character string, length)
length(tsvector), 409
levenshtein, 2425
levenshtein_less_equal, 2425
lex, 479
libedit, 479
    in psql, 1939
    psql##, 1939

libperl, 480
libpq, 796
    chunked mode, 842
    pipeline mode, 838
    single-row mode, 842
    #########, 838
    ######, 842
libpq-fe.h, 796, 813
libpq-int.h, 813
libpython, 480
library initialization function, 1060
LIKE, 215
    #####, 635
LIKE_REGEX, 229
    in SQL/JSON, 302
    SQL/JSON####, 302
LIMIT, 116
line, 147, 258
line segment, 147
linear regression, 326
Linux
    IPC configuration, 517
    IPC##, 517
    shared library, 1067
    ########, 513
    #######, 1067
LISTEN, 1715
listen_addresses configuration parameter, 535
listen_addresses#######, 535
llvm-config, 485, 495
ll_to_earth, 2418
ln, 195
lo, 2444
LOAD, 1717
load balancing, 679
locale, 634
localtime, 241
localtimestamp, 241
local_preload_libraries configuration parameter, 591
local_preload_libraries#######, 591
lock, 438
    advisory, 442
    monitoring, 734
LOCK, 438, 1718
lock_timeout configuration parameter, 586
lock_timeout #######, 586
log, 195
log shipping, 679
log10, 195
Logging
    current_logfiles file and the pg_current_logfile function, 340
    current_logfiles#####pg_current_logfile##, 340
    pg_current_logfile function, 340
    pg_current_logfile##, 340
logging_collector configuration parameter, 569
logging_collector#######, 569
Logical Decoding, 1323, 1326
logical_decoding_work_mem configuration parameter, 543
logical_decoding_work_mem#######, 543
log_autovacuum_min_duration
    configuration parameter, 574
    storage parameter, 1589
    #######, 1589
    #######, 574
log_btree_build_stats configuration parameter, 600
log_btree_build_stats#######, 600
log_checkpoints configuration parameter, 574
log_checkpoints#######, 574
log_connections configuration parameter, 574
log_connections#######, 574
log_destination configuration parameter, 568
log_destination#######, 568
log_directory configuration parameter, 570
log_directory#######, 570
log_disconnections configuration parameter, 574
log_disconnections#######, 574
log_duration configuration parameter, 574
log_duration#######, 574
log_error_verbosity configuration parameter, 574
log_error_verbosity#######, 574
log_executor_stats configuration parameter, 581
log_executor_stats#######, 581
log_filename configuration parameter, 570
log_filename#######, 570
log_file_mode configuration parameter, 570
log_file_mode #######, 570
log_hostname configuration parameter, 575
log_hostname#######, 575
log_line_prefix configuration parameter, 575
log_line_prefix#######, 575
log_lock_waits configuration parameter, 576
log_lock_waits#######, 576
log_min_duration_sample configuration parameter, 572
log_min_duration_sample#######, 572
log_min_duration_statement configuration parameter, 572
log_min_duration_statement#######, 572
log_min_error_statement configuration parameter, 571
log_min_error_statement#######, 571
log_min_messages configuration parameter, 571
log_min_messages#######, 571
log_parameter_max_length configuration parameter, 577
log_parameter_max_length_on_error configuration parameter, 577
log_parameter_max_length_on_error#######, 577
log_parameter_max_length#######, 577
log_parser_stats configuration parameter, 581
log_parser_stats#######, 581
log_planner_stats configuration parameter, 581
log_planner_stats#######, 581
log_recovery_conflict_waits configuration parameter, 576

log_recovery_conflict_waits#######, 576
log_replication_commands configuration parameter, 577
log_replication_commands#######, 577
log_rotation_age configuration parameter, 570
log_rotation_age#######, 570
log_rotation_size configuration parameter, 570
log_rotation_size#######, 570
log_startup_progress_interval configuration parameter, 573
log_startup_progress_interval#######, 573
log_statement configuration parameter, 577
log_statement_sample_rate configuration parameter, 572
log_statement_sample_rate#######, 572
log_statement_stats configuration parameter, 581
log_statement_stats#######, 581
log_statement#######, 577
log_temp_files configuration parameter, 577
log_temp_files#######, 577
log_timezone configuration parameter, 577
log_timezone#######, 577
log_transaction_sample_rate configuration parameter, 572
log_transaction_sample_rate#######, 572
log_truncate_on_rotation configuration parameter, 570
log_truncate_on_rotation#######, 570
longitude, 2418
looks_like_number
  in PL/Perl, 1235
  PL/Perl####, 1235
loop
  in PL/pgSQL, 1179
lower, 201, 320, 321
  #####, 635
lower_inc, 321, 321
lower_inf, 321, 321
lo_close, 887
lo_compat_privileges configuration parameter, 593
lo_compat_privileges#######, 593
lo_creat, 885, 888
lo_create, 884
lo_export, 885, 888
lo_from_bytea, 888
lo_get, 888
lo_import, 885, 888
lo_import_with_oid, 885
lo_lseek, 886
lo_lseek64, 887
lo_open, 885
lo_put, 888
lo_read, 886
lo_tell, 887
lo_tell64, 887
lo_truncate, 887
lo_truncate64, 887
lo_unlink, 888, 888
lo_write, 886
lpad, 201
lseg, 147, 258
LSN, 758
ltree, 2446
ltree2text, 2449
ltrim, 201, 209

M

MAC address (## macaddr)
MAC address (EUI-64 format) (## macaddr)
macaddr (data type), 150
macaddr (####), 150
macaddr8 (data type), 150
macaddr8 (####), 151
macaddr8_set7bit, 262
macOS
  installation on, 505
  IPC configuration, 517
  IPC##, 517
  shared library, 1067
  #########, 505
  #######, 1067
MAC#### (## macaddr)
MAC#### (EUI-64 ##) (## macaddr)
magic block, 1060
maintenance, 656
maintenance_io_concurrency configuration parameter, 547
maintenance_io_concurrency#######, 547
maintenance_work_mem configuration parameter, 542
maintenance_work_mem#######, 542
make, 479
makeaclitem, 343
make_date, 241
make_interval, 241
make_time, 241
make_timestamp, 241
make_timestamptz, 241
make_valid, 2441
MANPATH, 503
masklen, 261
materialized view
  implementation through rules, 1140
materialized views, 2071
max, 324
max_connections configuration parameter, 535
max_connections#######, 535
max_files_per_process configuration parameter, 544
max_files_per_process#######, 544
max_function_args configuration parameter, 596
max_function_args#######, 596
max_identifier_length configuration parameter, 596
max_identifier_length#######, 596
max_index_keys configuration parameter, 596
max_index_keys#######, 596
max_locks_per_transaction configuration parameter, 592
max_locks_per_transaction#######, 592

max_logical_replication_workers configuration parameter, 562
max_logical_replication_workers#######, 562
max_notify_queue_pages configuration parameter, 544
max_parallel_apply_workers_per_subscription configuration parameter, 562
max_parallel_apply_workers_per_subscription#######, 562
max_parallel_maintenance_workers configuration parameter, 547
max_parallel_maintenance_workers#######, 547
max_parallel_workers configuration parameter, 548
max_parallel_workers_per_gather configuration parameter, 547
max_parallel_workers_per_gather #######, 547
max_parallel_workers#######, 548
max_pred_locks_per_page configuration parameter, 593
max_pred_locks_per_page#######, 593
max_pred_locks_per_relation configuration parameter, 593
max_pred_locks_per_relation#######, 593
max_pred_locks_per_transaction configuration parameter, 593
max_pred_locks_per_transaction#######, 593
max_prepared_transactions configuration parameter, 542
max_prepared_transactions#######, 542
max_replication_slots configuration parameter
  in a sending server, 557
  in a subscriber, 562
max_slot_wal_keep_size configuration parameter, 557
max_slot_wal_keep_size#######, 557
max_stack_depth configuration parameter, 543
max_stack_depth#######, 543
max_standby_archive_delay configuration parameter, 560
max_standby_archive_delay#######, 560
max_standby_streaming_delay configuration parameter, 560
max_standby_streaming_delay#######, 560
max_sync_workers_per_subscription configuration parameter, 562
max_sync_workers_per_subscription#######, 562
max_wal_senders configuration parameter, 557
max_wal_senders#######, 557
max_wal_size configuration parameter, 552
max_wal_size#######, 552
max_worker_processes configuration parameter, 547
max_worker_processes#######, 547
md5, 203, 211
MD5, 612
median, 42
  (## percentile)
memory context
  in SPI, 1303
memory overcommit, 519
MERGE, 1721
  RETURNING, 99, 330
merge_action, 330
Meson, 479
metaphone, 2426
min, 324
MinGW
  installation on, 506
  #########, 506
min_dynamic_shared_memory configuration parameter, 544
min_dynamic_shared_memory#######, 544
min_parallel_index_scan_size configuration parameter, 566
min_parallel_index_scan_size#######, 566
min_parallel_table_scan_size configuration parameter, 566
min_parallel_table_scan_size#######, 566
min_scale, 196
min_wal_size configuration parameter, 553
min_wal_size#######, 553
mod, 196
mode
  statistical, 327
monitoring
  database activity, 696
MOVE, 1727
moving-aggregate mode, 1082
multirange (function), 322
multirange type, 178
Multiversion Concurrency Control, 433
MultiXactId, 661
multixact_member_buffers configuration parameter, 543
multixact_offset_buffers configuration parameter, 543
MVCC, 433
mxid_age, 352

N

name
  qualified, 76
  syntax of, 30
  unqualified, 77
NaN (## not a number) (## ##)
natural join, 102
negation, 188
NetBSD
  IPC configuration, 517
  IPC##, 517
  shared library, 1067
  ########, 513
  #######, 1067
netmask, 261
network, 261
  data types, 149
nextval, 310
NFS, 511
nlevel, 2449
non-durable, 465

nonblocking connection, 798, 833
normalize, 201
normalized, 200
normal_rand, 2530
NOT (operator), 188
not a number
    floating point, 130
    numeric (data type), 129
NOT IN, 331, 333
not-null constraint, 57
notation
    functions, 49
notice processing
    in libpq, 858
NOTIFY, 1729
    in libpq, 849
    libpq####, 849
notify_buffers configuration parameter, 543
NOTNULL, 191
NOT#####, 188
now, 241
npoints, 257
nth_value, 329
ntile, 329
null value
    with check constraints, 57
    comparing, 191
    in DISTINCT, 114
    in libpq, 829
    with unique constraints, 59
NULLIF, 313
NULL#
    ####, 57
    DISTINCT##, 114
    PL/Perl####, 1228
    PL/Python####, 1243
    #####, 59
    ######, 53
    ##, 191
number
    constant, 34
numeric, 35
numeric (data type), 128
numeric######, 128
numnode, 264, 410
num_nonnulls, 192
num_nulls, 192
NVL, 312

O

object identifier
    data type, 184
object-oriented database, 7
obj_description, 350
OCCURRENCES_REGEX, 229
octet_length, 201, 201, 209, 214
OFFSET, 116
oid, 184
OID
    libpq####, 831
oid2name, 2552
ON CONFLICT, 1709
ONLY, 101
OOM, 519
OpenBSD
    IPC configuration, 517
    IPC##, 517
    shared library, 1068
    ########, 513
    #######, 1068
OpenSSL, 485, 496
    (## SSL)
operator, 188
    invocation, 40
    logical, 188
    precedence, 37
    syntax, 36
    type resolution in an invocation, 376
    user-defined, 1090
operator class, 395
operator family, 395
optimization information
    for functions, 1080
    for operators, 1090
OR (operator), 188
Oracle
    PL/SQL##PL/pgSQL####, 1208
    porting from PL/SQL to PL/pgSQL, 1208
ORDER BY, 9, 116
    #####, 635
ordered-set aggregate, 40
    built-in, 327
ordering operator, 1102
ordinality, 338
OR#####, 188
outer join, 101
output function, 1086
OVER clause, 42
overcommit, 519
OVERLAPS, 242
overlay, 201, 210, 214
overloading
    functions, 1058
OVER#, 42
owner, 66

P

pageinspect, 2453
pages_per_range storage parameter, 1525
pages_per_range#######, 1525
page_checksum, 2453
page_header, 2453
palloc, 1066
PAM, 486, 496, 619
parallel query, 466

parallel_leader_participation configuration parameter, 548
parallel_leader_participation#######, 548
parallel_setup_cost configuration parameter, 565
parallel_setup_cost#######, 565
parallel_tuple_cost configuration parameter, 566
parallel_tuple_cost#######, 566
parallel_workers storage parameter, 1587
parallel_workers#######, 1587
parameter
    syntax, 38
parse_ident, 203
partition pruning, 92
partitioned table, 82
partitioning, 82
password
    authentication, 612
password file, 867
passwordcheck, 2463
password_encryption configuration parameter, 538
password_encryption#######, 538
path, 258
    for schemas, 583
PATH, 503
path (data type), 148
path######, 148
pattern matching, 214
patterns
    in psql and pg_dump, 1932
pclose, 257
peer, 615
percentile
    continuous, 327
    discrete, 327
percent_rank, 329
    hypothetical, 328
    ###, 328
performance, 446
perl, 479
Perl, 1227
permission (## privilege)
pfree, 1066
PGAPPNAME, 866
pgbench, 1839
PGcancel, 847, 847
PGcancelConn, 843
PGCHANNELBINDING, 866
PGCLIENTENCODING, 866
PGconn, 796
PGCONNECT_TIMEOUT, 866
pgcrypto, 2468
PGDATA, 510
PGDATABASE, 865
PGDATESTYLE, 866
PGEventProc, 861
PGGEQO, 866
PGGSSDELEGATION, 866
PGGSSENCMODE, 866
PGGSSLIB, 866
PGHOST, 865
PGHOSTADDR, 865
PGKRBSRVNAME, 866
PGLOADBALANCEHOSTS, 866
PGLOCALEDIR, 867
PGOPTIONS, 866
PGPASSFILE, 865
PGPASSWORD, 865
PGPORT, 865
pgp_armor_headers, 2472
pgp_key_id, 2471
pgp_pub_decrypt, 2471
pgp_pub_decrypt_bytea, 2471
pgp_pub_encrypt, 2471
pgp_pub_encrypt_bytea, 2471
pgp_sym_decrypt, 2471
pgp_sym_decrypt_bytea, 2471
pgp_sym_encrypt, 2471
pgp_sym_encrypt_bytea, 2471
PGREQUIREAUTH, 866
PGREQUIREPEER, 866
PGREQUIRESSL, 866
PGresult, 823
pgrowlocks, 2482, 2482
PGSERVICE, 866
PGSERVICEFILE, 866
PGSSLCERT, 866
PGSSLCERTMODE, 866
PGSSLCOMPRESSION, 866
PGSSLCRL, 866
PGSSLCRLDIR, 866
PGSSLKEY, 866
PGSSLMAXPROTOCOLVERSION, 866
PGSSLMINPROTOCOLVERSION, 866
PGSSLMODE, 866
PGSSLNEGOTIATION, 865
PGSSLROOTCERT, 866
PGSSLSNI, 866
pgstatginindex, 2493
pgstathashindex, 2494
pgstatindex, 2493
pgstattuple, 2492, 2492
pgstattuple_approx, 2494
PGSYSCONFDIR, 867
PGTARGETSESSIONATTRS, 866
PGTZ, 866
PGUSER, 865
pgxs, 1110
pg_advisory_lock, 369
pg_advisory_lock_shared, 369
pg_advisory_unlock, 369
pg_advisory_unlock_all, 369
pg_advisory_unlock_shared, 370
pg_advisory_xact_lock, 370
pg_advisory_xact_lock_shared, 370
pg_aggregate, 2015
pg_am, 2017

pg_amcheck, 1826
pg_amop, 2017
pg_amproc, 2018
pg_archivecleanup, 1959
pg_attrdef, 2019
pg_attribute, 2019
pg_authid, 2021
pg_auth_members, 2022
pg_available_extensions, 2064
pg_available_extension_versions, 2064
pg_available_wal_summaries, 355
pg_backend_memory_contexts, 2065
pg_backend_pid, 339
pg_backup_start, 358
pg_backup_stop, 358
pg_basebackup, 1831
pg_basetype, 345
pg_blocking_pids, 339
pg_buffercache, 2464
pg_buffercache_evict, 2464
pg_buffercache_pages, 2464
pg_buffercache_summary, 2464
pg_cancel_backend, 356
pg_cast, 2022
pg_char_to_encoding, 345
pg_checksums, 1961
pg_class, 2023
pg_client_encoding, 203
pg_collation, 2025
pg_collation_actual_version, 366
pg_collation_is_visible, 344
PG_COLOR, 2587
PG_COLORS, 2587
pg_column_compression, 364
pg_column_size, 364
pg_column_toast_chunk_id, 364
pg_combinebackup, 1861
pg_config, 1864, 2065
    ecpg##, 947
    libpq####, 873
    #####C##, 1066
pg_conf_load_time, 339
pg_constraint, 2026
pg_controldata, 1963
pg_control_checkpoint, 353
pg_control_init, 353
pg_control_recovery, 353
pg_control_system, 353
pg_conversion, 2028
pg_conversion_is_visible, 344
pg_copy_logical_replication_slot, 362
pg_copy_physical_replication_slot, 362
pg_createsubscriber, 1964
pg_create_logical_replication_slot, 362
pg_create_physical_replication_slot, 361
pg_create_restore_point, 358
pg_ctl, 510, 512, 1968
pg_current_logfile, 340
pg_current_snapshot, 351
pg_current_wal_flush_lsn, 358
pg_current_wal_insert_lsn, 358
pg_current_wal_lsn, 358
pg_current_xact_id, 351
pg_current_xact_id_if_assigned, 351
pg_cursors, 2066
pg_database, 631, 2028
pg_database_collation_actual_version, 366
pg_database_size, 364
pg_db_role_setting, 2029
pg_ddl_command, 186
pg_default_acl, 2030
pg_depend, 2030
pg_describe_object, 349
pg_description, 2032
pg_drop_replication_slot, 362
pg_dump, 1867
pg_dumpall, 1880
    use during upgrade, 523
    ###########, 523
pg_encoding_to_char, 345
pg_enum, 2032
pg_event_trigger, 2033
pg_event_trigger_ddl_commands, 371
pg_event_trigger_dropped_objects, 372
pg_event_trigger_table_rewrite_oid, 373
pg_event_trigger_table_rewrite_reason, 373
pg_export_snapshot, 361
pg_extension, 2033
pg_extension_config_dump, 1107
pg_filenode_relation, 366
pg_file_settings, 2066
pg_foreign_data_wrapper, 2034
pg_foreign_server, 2034
pg_foreign_table, 2035
pg_freespace, 2478
pg_freespacemap, 2478
pg_function_is_visible, 344
pg_get_catalog_foreign_keys, 345
pg_get_constraintdef, 345
pg_get_expr, 345
pg_get_functiondef, 345
pg_get_function_arguments, 345
pg_get_function_identity_arguments, 345
pg_get_function_result, 345
pg_get_indexdef, 345
pg_get_keywords, 345
pg_get_object_address, 350
pg_get_partkeydef, 346
pg_get_ruledef, 346
pg_get_serial_sequence, 346
pg_get_statisticsobjdef, 346
pg_get_triggerdef, 346
pg_get_userbyid, 346
pg_get_viewdef, 346
pg_get_wal_replay_pause_state, 360
pg_get_wal_resource_managers, 360

pg_get_wal_summarizer_state, 355
pg_group, 2067
pg_has_role, 342
pg_hba.conf, 603
pg_hba_file_rules, 2067
pg_ident.conf, 610
pg_identify_object, 349
pg_identify_object_as_address, 349
pg_ident_file_mappings, 2068
pg_import_system_collations, 366
pg_index, 2035
pg_indexam_has_property, 346
pg_indexes, 2069
pg_indexes_size, 365
pg_index_column_has_property, 346
pg_index_has_property, 346
pg_inherits, 2037
pg_init_privs, 2037
pg_input_error_info, 350
pg_input_is_valid, 350
pg_isready, 1886
pg_is_in_recovery, 359
pg_is_other_temp_schema, 340
pg_is_wal_replay_paused, 360
pg_jit_available, 340
pg_language, 2037
pg_largeobject, 2038
pg_largeobject_metadata, 2038
pg_last_committed_xact, 353
pg_last_wal_receive_lsn, 360
pg_last_wal_replay_lsn, 360
pg_last_xact_replay_timestamp, 360
pg_listening_channels, 340
pg_locks, 2069
pg_logical_emit_message, 364
pg_logical_slot_get_binary_changes, 363
pg_logical_slot_get_changes, 362
pg_logical_slot_peek_binary_changes, 363
pg_logical_slot_peek_changes, 362
pg_log_backend_memory_contexts, 356
pg_log_standby_snapshot, 361
pg_lsn, 186
pg_ls_archive_statusdir, 368
pg_ls_dir, 367
pg_ls_logdir, 368
pg_ls_logicalmapdir, 368
pg_ls_logicalsnapdir, 368
pg_ls_replslotdir, 368
pg_ls_tmpdir, 368
pg_ls_waldir, 368
pg_matviews, 2071
pg_mcv_list_items, 373
pg_my_temp_schema, 340
pg_namespace, 2039
pg_notification_queue_usage, 340
pg_notify, 1730
pg_opclass, 2039
pg_opclass_is_visible, 344
pg_operator, 2040
pg_operator_is_visible, 344
pg_opfamily, 2041
pg_opfamily_is_visible, 344
pg_options_to_table, 347
pg_parameter_acl, 2041
pg_partitioned_table, 2041
pg_partition_ancestors, 366
pg_partition_root, 366
pg_partition_tree, 366
pg_policies, 2072
pg_policy, 2042
pg_postmaster_start_time, 340
pg_prepared_statements, 2072
pg_prepared_xacts, 2073
pg_prewarm, 2480
pg_prewarm.autoprewarm configuration parameter, 2480
pg_prewarm.autoprewarm_interval configuration parameter, 2480
pg_prewarm.autoprewarm_interval#######, 2480
pg_prewarm.autoprewarm#######, 2480
pg_proc, 2043
pg_promote, 360
pg_publication, 2045
pg_publication_namespace, 2045
pg_publication_rel, 2046
pg_publication_tables, 2073
pg_range, 2046
pg_read_binary_file, 369
pg_read_file, 368
pg_receivewal, 1888
pg_receivexlog, 2588 (## pg_receivewal)
pg_recvlogical, 1892
pg_relation_filenode, 365
pg_relation_filepath, 366
pg_relation_size, 365
pg_reload_conf, 356
pg_relpages, 2494
pg_replication_origin, 2047
pg_replication_origin_advance, 363
pg_replication_origin_create, 363
pg_replication_origin_drop, 363
pg_replication_origin_oid, 363
pg_replication_origin_progress, 363
pg_replication_origin_session_is_setup, 363
pg_replication_origin_session_progress, 363
pg_replication_origin_session_reset, 363
pg_replication_origin_session_setup, 363
pg_replication_origin_status, 2074
pg_replication_origin_xact_reset, 363
pg_replication_origin_xact_setup, 363
pg_replication_slots, 2074
pg_replication_slot_advance, 363
pg_resetwal, 1973
pg_resetxlog, 2588 (## pg_resetwal)
pg_restore, 1896
pg_rewind, 1977

pg_rewrite, 2047
pg_roles, 2076
pg_rotate_logfile, 357
pg_rules, 2077
pg_safe_snapshot_blocking_pids, 340
pg_seclabel, 2047
pg_seclabels, 2077
pg_sequence, 2048
pg_sequences, 2077
pg_service.conf, 867
pg_settings, 2078
pg_settings_get_flags, 347
pg_shadow, 2080
pg_shdepend, 2048
pg_shdescription, 2049
pg_shmem_allocations, 2081
pg_shseclabel, 2050
pg_size_bytes, 365
pg_size_pretty, 365
pg_sleep, 251
pg_sleep_for, 251
pg_sleep_until, 251
pg_snapshot_xip, 351
pg_snapshot_xmax, 351
pg_snapshot_xmin, 351
pg_split_walfile_name, 359
pg_statio_all_indexes, 700, 730
pg_statio_all_sequences, 700, 730
pg_statio_all_tables, 700, 729
pg_statio_sys_indexes, 700
pg_statio_sys_sequences, 700
pg_statio_sys_tables, 700
pg_statio_user_indexes, 700
pg_statio_user_sequences, 700
pg_statio_user_tables, 700
pg_statistic, 458, 2050
pg_statistics_obj_is_visible, 344
pg_statistic_ext, 459, 2051
pg_statistic_ext_data, 459, 2051
pg_stats, 458, 2081
pg_stats_ext, 2082
pg_stats_ext_exprs, 2083
pg_stat_activity, 698, 700
pg_stat_all_indexes, 700, 728
pg_stat_all_tables, 699, 727
pg_stat_archiver, 699, 720
pg_stat_bgwriter, 699, 723
pg_stat_checkpointer, 699, 723
pg_stat_clear_snapshot, 732
pg_stat_database, 699, 724
pg_stat_database_conflicts, 699, 726
pg_stat_file, 369
pg_stat_get_activity, 732
pg_stat_get_backend_activity, 734
pg_stat_get_backend_activity_start, 734
pg_stat_get_backend_client_addr, 734
pg_stat_get_backend_client_port, 734
pg_stat_get_backend_dbid, 734
pg_stat_get_backend_idset, 734
pg_stat_get_backend_pid, 734
pg_stat_get_backend_start, 734
pg_stat_get_backend_subxact, 734
pg_stat_get_backend_userid, 734
pg_stat_get_backend_wait_event, 734
pg_stat_get_backend_wait_event_type, 734
pg_stat_get_backend_xact_start, 734
pg_stat_get_snapshot_timestamp, 732
pg_stat_get_xact_blocks_fetched, 732
pg_stat_get_xact_blocks_hit, 732
pg_stat_gssapi, 698, 720
pg_stat_io, 699, 721
pg_stat_progress_analyze, 698, 735
pg_stat_progress_basebackup, 698, 741
pg_stat_progress_cluster, 698, 736
pg_stat_progress_copy, 699, 737
pg_stat_progress_create_index, 698, 738
pg_stat_progress_vacuum, 698, 739
pg_stat_recovery_prefetch, 698, 718
pg_stat_replication, 698, 714
pg_stat_replication_slots, 699, 716
pg_stat_reset, 732
pg_stat_reset_replication_slot, 733
pg_stat_reset_shared, 732
pg_stat_reset_single_function_counters, 733
pg_stat_reset_single_table_counters, 733
pg_stat_reset_slru, 733
pg_stat_reset_subscription_stats, 733
pg_stat_slru, 699, 731
pg_stat_ssl, 698, 719
pg_stat_statements, 2484
  function, 2488
  ##, 2488
pg_stat_statements.max configuration parameter, 2488
pg_stat_statements.max#######, 2488
pg_stat_statements.save configuration parameter, 2489
pg_stat_statements.save#######, 2489
pg_stat_statements.track configuration parameter, 2488
pg_stat_statements.track_planning configuration parameter, 2489
pg_stat_statements.track_planning#######, 2489
pg_stat_statements.track_utility configuration parameter, 2488
pg_stat_statements.track_utility#######, 2488
pg_stat_statements.track#######, 2488
pg_stat_statements_info, 2487
pg_stat_statements_reset, 2488
pg_stat_subscription, 698, 718
pg_stat_subscription_stats, 699, 719
pg_stat_sys_indexes, 700
pg_stat_sys_tables, 699
pg_stat_user_functions, 700, 730
pg_stat_user_indexes, 700
pg_stat_user_tables, 699
pg_stat_wal, 699, 724
pg_stat_wal_receiver, 698, 717

pg_stat_xact_all_tables, 699
pg_stat_xact_sys_tables, 699
pg_stat_xact_user_functions, 700
pg_stat_xact_user_tables, 699
pg_subscription, 2053
pg_subscription_rel, 2054
pg_surgery, 2496
pg_switch_wal, 358
pg_sync_replication_slots, 364
pg_tables, 2085
pg_tablespace, 2054
pg_tablespace_databases, 347
pg_tablespace_location, 347
pg_tablespace_size, 365
pg_table_is_visible, 344
pg_table_size, 365
pg_temp, 584
  ######, 1519
pg_terminate_backend, 357
pg_test_fsync, 1981
pg_test_timing, 1982
pg_timezone_abbrevs, 2085
pg_timezone_names, 2086
pg_total_relation_size, 365
pg_transform, 2055
pg_trgm, 2498
pg_trgm.similarity_threshold configuration parameter, 2500
pg_trgm.similarity_threshold#######, 2500
pg_trgm.strict_word_similarity_threshold configuration parameter, 2500
pg_trgm.strict_word_similarity_threshold####### , 2500
pg_trgm.word_similarity_threshold configuration parameter, 2500
pg_trgm.word_similarity_threshold####### , 2500
pg_trigger, 2055
pg_trigger_depth, 340
pg_try_advisory_lock, 370
pg_try_advisory_lock_shared, 370
pg_try_advisory_xact_lock, 370
pg_try_advisory_xact_lock_shared, 370
pg_ts_config, 2056
pg_ts_config_is_visible, 344
pg_ts_config_map, 2057
pg_ts_dict, 2057
pg_ts_dict_is_visible, 344
pg_ts_parser, 2058
pg_ts_parser_is_visible, 344
pg_ts_template, 2058
pg_ts_template_is_visible, 344
pg_type, 2059
pg_typeof, 347
pg_type_is_visible, 344
pg_upgrade, 1985
pg_user, 2086
pg_user_mapping, 2062
pg_user_mappings, 2087
pg_verifybackup, 1905
pg_views, 2087
pg_visibility, 2503
pg_visible_in_snapshot, 351
pg_wait_events, 2088
pg_waldump, 1994
pg_walfile_name, 359
pg_walfile_name_offset, 359
pg_walinspect, 2505
pg_walsummary, 1997
pg_wal_lsn_diff, 359
pg_wal_replay_pause, 360
pg_wal_replay_resume, 360
pg_wal_summary_contents, 355
pg_xact_commit_timestamp, 353
pg_xact_commit_timestamp_origin, 353
pg_xact_status, 351
pg_xlogdump, 2588 (## pg_waldump)
phraseto_tsquery, 264, 405
pi, 196
PIC, 1067
PID
  ########PID###
    libpq####, 817
pipelining
  in libpq, 838
  protocol specification, 2096
PITR, 666
PITR standby, 679
PITR#####, 679
pkg-config, 479
  ecpg##, 947
  libpq####, 873
PL/Perl, 1227
PL/PerlU, 1237
PL/pgSQL, 1158
PL/Python, 1241
PL/SQL (Oracle)
  PL/pgSQL####, 1208
  porting to PL/pgSQL, 1208
PL/Tcl, 1217
plainto_tsquery, 264, 404
plan_cache_mode configuration parameter, 568
plan_cache_mode#######, 568
plperl.on_init configuration parameter, 1239
plperl.on_init#######, 1239
plperl.on_plperlu_init configuration parameter, 1240
plperl.on_plperlu_init#######, 1240
plperl.on_plperl_init configuration parameter, 1240
plperl.on_plperl_init#######, 1240
plperl.use_strict configuration parameter, 1240
plperl.use_strict#######, 1240
plpgsql.check_asserts configuration parameter, 1194
plpgsql.check_asserts #######, 1194
plpgsql.variable_conflict configuration parameter, 1203
plpgsql.variable_conflict#######, 1203
pltcl.start_proc configuration parameter, 1225

pltcl.start_proc#######, 1225
pltclu.start_proc configuration parameter, 1226
pltclu.start_proc#######, 1226
point, 147, 258
point-in-time recovery, 666
policy, 70
polygon, 148, 259
polymorphic function, 1042
polymorphic type, 1042
popcount (## bit_count)
popen, 257
populate_record, 2431
port, 806
port configuration parameter, 535
portal
    DECLARE, 1631
    in PL/pgSQL, 1187
port#######, 535
position, 201, 210, 214
POSITION_REGEX, 229
POSTGRES, xxxiii
postgres, 3, 512, 630, 1998
postgres user, 510
Postgres95, xxxiii
postgresql.auto.conf, 532
postgresql.conf, 531
postgres_fdw, 2508
postgres_fdw.application_name configuration parameter, 2515
postgres###, 510
post_auth_delay configuration parameter, 598
post_auth_delay#######, 598
power, 196
PQbackendPID, 817
PQbinaryTuples, 829
    with COPY, 850
PQcancel, 847
PQcancelBlocking, 844
PQcancelCreate, 843
PQcancelErrorMessage, 846
PQcancelFinish, 846
PQcancelPoll, 844
PQcancelReset, 847
PQcancelSocket, 846
PQcancelStart, 844
PQcancelStatus, 846
PQchangePassword, 856
PQclear, 826
PQclientEncoding, 853
PQclosePortal, 822
PQclosePrepared, 822
PQcmdStatus, 830
PQcmdTuples, 831
PQconndefaults, 800
PQconnectdb, 797
PQconnectdbParams, 796
PQconnectionNeedsPassword, 817
PQconnectionUsedGSSAPI, 817
PQconnectionUsedPassword, 817
PQconnectPoll, 798
PQconnectStart, 798
PQconnectStartParams, 798
PQconninfo, 801
PQconninfoFree, 855
PQconninfoParse, 801
PQconsumeInput, 836
PQcopyResult, 857
PQdb, 813
PQdescribePortal, 822
PQdescribePrepared, 822
PQencryptPassword, 856
PQencryptPasswordConn, 855
PQendcopy, 853
PQenterPipelineMode, 840
PQerrorMessage, 816
PQescapeBytea, 833
PQescapeByteaConn, 833
PQescapeIdentifier, 832
PQescapeLiteral, 831
PQescapeString, 832
PQescapeStringConn, 832
PQexec, 819
PQexecParams, 819
PQexecPrepared, 821
PQexitPipelineMode, 841
PQfformat, 828
    with COPY, 850
PQfinish, 801
PQfireResultCreateEvents, 856
PQflush, 837
PQfmod, 828
PQfn, 848
PQfname, 827
PQfnumber, 827
PQfreeCancel, 847
PQfreemem, 855
PQfsize, 829
PQftable, 828
PQftablecol, 828
PQftype, 828
PQgetCancel, 847
PQgetCopyData, 851
PQgetCurrentTimeUSec, 858
PQgetisnull, 829
PQgetlength, 829
PQgetline, 851
PQgetlineAsync, 852
PQgetResult, 836
PQgetssl, 819
PQgetSSLKeyPassHook_OpenSSL, 803
PQgetvalue, 829
PQhost, 814
PQhostaddr, 814
PQinitOpenSSL, 871
PQinitSSL, 872
PQinstanceData, 862

PQisBusy, 837
PQisnonblocking, 837
PQisthreadsafe, 872
PQlibVersion, 858
    (## PQserverVersion)
PQmakeEmptyPGresult, 856
PQnfields, 827
    with COPY, 850
PQnotifies, 849
PQnparams, 830
PQntuples, 827
PQoidStatus, 831
PQoidValue, 831
PQoptions, 815
PQparameterStatus, 815
PQparamtype, 830
PQpass, 814
PQping, 802
PQpingParams, 802
PQpipelineStatus, 840
PQpipelineSync, 841
PQport, 814
PQprepare, 821
PQprint, 830
PQprotocolVersion, 816
PQputCopyData, 850
PQputCopyEnd, 850
PQputline, 852
PQputnbytes, 852
PQregisterEventProc, 862
PQrequestCancel, 848
PQreset, 801
PQresetPoll, 802
PQresetStart, 802
PQresStatus, 824
PQresultAlloc, 857
PQresultErrorField, 825
PQresultErrorMessage, 824
PQresultInstanceData, 862
PQresultMemorySize, 858
PQresultSetInstanceData, 862
PQresultStatus, 823
PQresultVerboseErrorMessage, 824
PQsendClosePortal, 835
PQsendClosePrepared, 835
PQsendDescribePortal, 835
PQsendDescribePrepared, 835
PQsendFlushRequest, 841
PQsendPipelineSync, 841
PQsendPrepare, 834
PQsendQuery, 834
PQsendQueryParams, 834
PQsendQueryPrepared, 835
PQserverVersion, 816
PQsetChunkedRowsMode, 843
PQsetClientEncoding, 853
PQsetdb, 797
PQsetdbLogin, 797
PQsetErrorContextVisibility, 854
PQsetErrorVerbosity, 853
PQsetInstanceData, 862
PQsetnonblocking, 837
PQsetNoticeProcessor, 858
PQsetNoticeReceiver, 858
PQsetResultAttrs, 857
PQsetSingleRowMode, 842
PQsetSSLKeyPassHook_OpenSSL, 803
PQsetTraceFlags, 854
PQsetvalue, 857
PQsocket, 816
PQsocketPoll, 800
PQsslAttribute, 817
PQsslAttributeNames, 818
PQsslInUse, 817
PQsslStruct, 818
PQstatus, 815
PQtrace, 854
PQtransactionStatus, 815
PQtty, 815
PQunescapeBytea, 833
PQuntrace, 855
PQuser, 814
predicate locking, 436
PREPARE, 1731
PREPARE TRANSACTION, 1733
prepared statements
    creating, 1731
    executing, 1691
    removing, 1630
    showing the query plan, 1692
preparing a query
    in PL/pgSQL, 1204
pre_auth_delay configuration parameter, 598
pre_auth_delay#######, 598
primary key, 59
primary_conninfo configuration parameter, 559
primary_conninfo#######, 559
primary_slot_name configuration parameter, 560
primary_slot_name#######, 560
privilege, 66
    default, 68
    querying, 341
    with rules, 1152
    for schemas, 78
    with views, 1152
procedural language, 1156
    externally maintained, 2559
    handler for, 2154
procedure
    user-defined, 1044
procedures
    output parameter, 1050
protocol
    frontend-backend, 2089
ps
    to monitor activity, 696

##########, 696
psql, 5, 1908
Python, 1241

Q

qualified name, 76
query, 9, 100
query cancellation, 843
query plan, 446
query tree, 1133
querytree, 264, 410
quotation marks
and identifiers, 31
escaping, 31
quote_all_identifiers configuration parameter, 594
quote_all_identifiers #######, 594
quote_ident, 204
in PL/Perl, 1235
PL/Perl####, 1235
PL/pgSQL####, 1171
use in PL/pgSQL, 1171
quote_literal, 204
in PL/Perl, 1235
PL/Perl####, 1235
PL/pgSQL####, 1171
use in PL/pgSQL, 1171
quote_nullable, 204
in PL/Perl, 1235
PL/Perl####, 1235
PL/pgSQL####, 1171
use in PL/pgSQL, 1171

R

radians, 196
radius, 257
RADIUS, 618
RAISE
in PL/pgSQL, 1193
PL/pgSQL####, 1193
random, 197, 197
random_normal, 197
random_page_cost configuration parameter, 565
random_page_cost#######, 565
range type, 178
exclude, 182
indexes on, 182
range_agg, 324
range_intersect_agg, 324
range_merge, 321, 322
rank, 329
hypothetical, 328
###, 328
read committed, 434
read-only transaction, 585
setting, 1787
setting default, 585
readline, 479
Readline
in psql, 1939
psql##, 1939
READ_REPLICATION_SLOT, 2104
real, 130
REASSIGN OWNED, 1735
record, 186
recovery.conf, 2588, 2588
recovery.signal, 554
recovery_end_command configuration parameter, 555
recovery_end_command#######, 555
recovery_init_sync_method configuration parameter, 595
recovery_init_sync_method#######, 595
recovery_min_apply_delay configuration parameter, 561
recovery_min_apply_delay#######, 561
recovery_prefetch configuration parameter, 554
recovery_prefetch#######, 554
recovery_target configuration parameter, 555
recovery_target_action configuration parameter, 556
recovery_target_action#######, 556
recovery_target_inclusive configuration parameter, 556
recovery_target_inclusive#######, 556
recovery_target_lsn configuration parameter, 555
recovery_target_lsn#######, 555
recovery_target_name configuration parameter, 555
recovery_target_name#######, 555
recovery_target_time configuration parameter, 555
recovery_target_timeline configuration parameter, 556
recovery_target_timeline#######, 556
recovery_target_time#######, 555
recovery_target_xid configuration parameter, 555
recovery_target_xid#######, 555
recovery_target#######, 555
rectangle, 148
RECURSIVE
in common table expressions, 118
in views, 1625
#######, 1625
#########, 118
recursive_worktable_factor configuration parameter, 568
recursive_worktable_factor#######, 568
referential integrity, 15, 60
REFRESH MATERIALIZED VIEW, 1736
regclass, 184
regcollation, 184
regconfig, 184
regdictionary, 184
regexp_count, 204, 217
regexp_instr, 204, 217
regexp_like, 204, 217
regexp_match, 204, 217
regexp_matches, 204, 217
regexp_replace, 205, 217
regexp_split_to_array, 205, 217

regexp_split_to_table, 205, 217
regexp_substr, 205, 217
regnamespace, 184
regoper, 184
regoperator, 184
regproc, 184
regprocedure, 184
regression intercept, 326
regression slope, 326
regression test, 482, 493
regression tests, 783
regrole, 184
regr_avgx, 326
regr_avgy, 326
regr_count, 326
regr_intercept, 326
regr_r2, 326
regr_slope, 326
regr_sxx, 326
regr_sxy, 326
regr_syy, 326
regtype, 184
regular expression, 216, 217
    (## pattern matching)
reindex, 663
REINDEX, 1738
reindexdb, 1945
relation, 7
relational database, 7
RELEASE SAVEPOINT, 1743
remove_temp_files_after_crash configuration parameter, 601
remove_temp_files_after_crash#######, 601
repeat, 205
repeatable read, 435
replace, 205
replication, 679
Replication Origins, 1339
Replication Progress Tracking, 1339
replication slot
    logical replication, 1326
    streaming replication, 684
reporting errors
    in PL/pgSQL, 1193
reserved_connections configuration parameter, 535
reserved_connections#######, 535
RESET, 1745
restartpoint, 756, 756
restart_after_crash configuration parameter, 595
restart_after_crash#######, 595
restore_command configuration parameter, 554
restore_command#######, 554
RESTRICT
    DROP#, 94
    with DROP, 94
    foreign key action, 61
    ######, 61
restrict_nonsystem_relation_kind configuration parameter, 589
retryable error, 444
RETURN NEXT
    in PL/pgSQL, 1174
    PL/pgSQL####, 1174
RETURN QUERY
    in PL/pgSQL, 1174
    PL/pgSQL####, 1174
RETURNING, 99
RETURNING INTO
    in PL/pgSQL, 1168
    PL/pgSQL####, 1168
reverse, 205
REVOKE, 66, 1746
right, 205
right join, 102
role, 622, 626
    membership in, 624
ROLLBACK, 1750
rollback
    psql, 1935
ROLLBACK PREPARED, 1751
ROLLBACK TO SAVEPOINT, 1752
ROLLUP, 110
round, 196
routine maintenance, 656
row, 7, 52
ROW, 47
row estimation
    multivariate, 2273
    planner, 2269
row type, 173
    constructor, 47
row-level security, 70
row-wise comparison, 333
row_number, 329
row_security configuration parameter, 584
row_security_active, 342
row_security#######, 584
row_to_json, 284
rpad, 201
rtrim, 202, 210
rule, 1133
    and materialized views, 1140
    and views, 1134
    for DELETE, 1143
    for INSERT, 1143
    for SELECT, 1134
    compared with triggers, 1154
    for UPDATE, 1143

S

SAVEPOINT, 1754
savepoints
    defining, 1754
    releasing, 1743
    rolling back, 1752
scalar (## expression)

scale, 196
schema, 76, 629
  creating, 76
  current, 77, 339
  public, 77
  removing, 77
SCRAM, 612
scram_iterations configuration parameter, 538
scram_iterations#######, 538
search path, 77
  current, 339
  object visibility, 344
search_path configuration parameter, 78, 583
  use in securing functions, 1519
search_path#######, 78, 583
  ############, 1519
SECURITY LABEL, 1756
sec_to_gc, 2418
seg, 2517
segment_size configuration parameter, 596
segment_size#######, 596
SELECT, 9, 100, 1759
  determination of result type, 384
  select list, 113
  ######, 384
  #####, 113
SELECT INTO, 1777
  in PL/pgSQL, 1168
  PL/pgSQL####, 1168
semaphores, 515
send_abort_for_crash configuration parameter, 601
send_abort_for_crash#######, 601
send_abort_for_kill configuration parameter, 601
send_abort_for_kill#######, 601
sepgsql, 2520
sepgsql.debug_audit configuration parameter, 2522
sepgsql.debug_audit#######, 2522
sepgsql.permissive configuration parameter, 2522
sepgsql.permissive#######, 2522
sequence, 310
  and serial type, 131
sequential scan, 564
seq_page_cost configuration parameter, 565
seq_page_cost#######, 565
serial, 131
serial2, 131
serial4, 131
serial8, 131
serializable, 436
Serializable Snapshot Isolation, 433
serializable_buffers configuration parameter, 543
serialization anomaly, 436
serialization failure, 444
server log, 568
  log file maintenance, 663
Server Name Indication, 811
server spoofing, 524
server_encoding configuration parameter, 596
server_encoding#######, 596
server_version configuration parameter, 596
server_version_num configuration parameter, 596
server_version_num#######, 596
server_version#######, 596
session_preload_libraries configuration parameter, 591
session_preload_libraries#######, 591
session_replication_role configuration parameter, 586
session_replication_role#######, 586
session_user, 340
SET, 356, 1779
SET CONSTRAINTS, 1782
set difference, 115
set intersection, 115
set operation, 115
set returning functions
  functions, 335
SET ROLE, 1783
SET SESSION AUTHORIZATION, 1785
SET TRANSACTION, 1787
set union, 115
SET XML OPTION, 588
setseed, 198
setval, 310
setweight, 264, 409
  setweight for specific lexeme(s), 264
set_bit, 211, 214
set_byte, 211
set_config, 356
set_limit, 2498
set_masklen, 261
sha224, 211
sha256, 211
sha384, 211
sha512, 211
shared library, 502
shared memory, 515
shared_buffers configuration parameter, 540
shared_buffers#######, 540
shared_memory_size configuration parameter, 596
shared_memory_size_in_huge_pages configuration parameter, 597
shared_memory_size_in_huge_pages#######, 597
shared_memory_size#######, 596
shared_memory_type configuration parameter, 544
shared_memory_type#######, 544
shared_preload_libraries, 1077, 1077, 1078, 1078
shared_preload_libraries configuration parameter, 591
shared_preload_libraries#######, 591
shobj_description, 350
SHOW, 356, 1790, 2103
show_limit, 2498
show_trgm, 2498
shutdown, 521
SIGHUP, 532, 603, 610
SIGINT, 521
sign, 196
signal

backend processes, 356
significant digits, 589
SIGQUIT, 521
SIGTERM, 521
SIMILAR TO, 216
similarity, 2498
sin, 199
sind, 199
single-user mode, 2001
sinh, 199
skeys, 2429
sleep, 251
slice, 2430
sliced bread (## TOAST)
slope, 257
SLRU, 731
smallint, 128
smallserial, 131
Solaris
installation on, 506
shared library, 1068
########, 513
#########, 506
#######, 1068
SOME, 325, 331, 333
sort, 2437
sorting, 116
sort_asc, 2437
sort_desc, 2437
soundex, 2423
SP-GiST (## index) (## ######)
SPI, 1255
examples, 2527
#, 2527
spi_commit
in PL/Perl, 1234
SPI_commit, 1314
SPI_commit_and_chain, 1314
SPI_connect, 1256
SPI_connect_ext, 1256
SPI_copytuple, 1307
spi_cursor_close
in PL/Perl, 1232
PL/Perl####, 1232
SPI_cursor_close, 1289
SPI_cursor_fetch, 1285
SPI_cursor_find, 1284
SPI_cursor_move, 1286
SPI_cursor_open, 1279
SPI_cursor_open_with_args, 1280
SPI_cursor_open_with_paramlist, 1282
SPI_cursor_parse_open, 1283
SPI_exec, 1261
SPI_execp, 1278
SPI_execute, 1258
SPI_execute_extended, 1262
SPI_execute_plan, 1274
SPI_execute_plan_extended, 1275
SPI_execute_plan_with_paramlist, 1277
SPI_execute_with_args, 1264
spi_exec_prepared
in PL/Perl, 1233
PL/Perl####, 1233
spi_exec_query
in PL/Perl, 1231
PL/Perl####, 1231
spi_fetchrow
in PL/Perl, 1232
PL/Perl####, 1232
SPI_finish, 1257
SPI_fname, 1295
SPI_fnumber, 1296
spi_freeplan
in PL/Perl, 1233
PL/Perl####, 1233
SPI_freeplan, 1313
SPI_freetuple, 1311
SPI_freetuptable, 1312
SPI_getargcount, 1271
SPI_getargtypeid, 1272
SPI_getbinval, 1298
SPI_getnspname, 1302
SPI_getrelname, 1301
SPI_gettype, 1299
SPI_gettypeid, 1300
SPI_getvalue, 1297
SPI_is_cursor_plan, 1273
SPI_keepplan, 1290
SPI_modifytuple, 1309
SPI_palloc, 1304
SPI_pfree, 1306
spi_prepare
in PL/Perl, 1233
PL/Perl####, 1233
SPI_prepare, 1266
SPI_prepare_cursor, 1268
SPI_prepare_extended, 1269
SPI_prepare_params, 1270
spi_query
in PL/Perl, 1232
PL/Perl####, 1232
spi_query_prepared
in PL/Perl, 1233
PL/Perl####, 1233
SPI_register_relation, 1292
SPI_register_trigger_data, 1294
SPI_repalloc, 1305
SPI_result_code_string, 1303
SPI_returntuple, 1308
spi_rollback
in PL/Perl, 1234
SPI_rollback, 1315
SPI_rollback_and_chain, 1315
SPI_saveplan, 1291
SPI_scroll_cursor_fetch, 1287
SPI_scroll_cursor_move, 1288

SPI_start_transaction, 1316
SPI_unregister_relation, 1293
split_part, 205
SQL/CLI, 2324
SQL/Foundation, 2324
SQL/Framework, 2324
SQL/JRT, 2324
SQL/JSON
    functions and expressions, 281
    ####, 281
SQL/JSON path language, 292
SQL/JSON####, 292
SQL/MDA, 2324
SQL/MED, 2324
SQL/OLB, 2324
SQL/PGQ, 2324
SQL/PSM, 2324
SQL/Schemata, 2324
SQL/XML, 2324
    limits and conformance, 2344
    ######, 2344
SQL###, 1041
sqrt, 196
ssh, 529
SSI, 433
SSL
    in libpq, 819
    libpq##, 809
    TLS, 526, 868
ssl configuration parameter, 538
sslinfo, 2528
ssl_ca_file configuration parameter, 538
ssl_ca_file #######, 538
ssl_cert_file configuration parameter, 538
ssl_cert_file #######, 538
ssl_cipher, 2528
ssl_ciphers configuration parameter, 539
ssl_ciphers#######, 539
ssl_client_cert_present, 2528
ssl_client_dn, 2528
ssl_client_dn_field, 2528
ssl_client_serial, 2528
ssl_crl_dir configuration parameter, 539
ssl_crl_dir#######, 539
ssl_crl_file configuration parameter, 538
ssl_crl_file #######, 538
ssl_dh_params_file configuration parameter, 540
ssl_dh_params_file#######, 540
ssl_ecdh_curve configuration parameter, 539
ssl_ecdh_curve#######, 539
ssl_extension_info, 2529
ssl_issuer_dn, 2528
ssl_issuer_field, 2529
ssl_is_used, 2528
ssl_key_file configuration parameter, 539
ssl_key_file #######, 539
ssl_library configuration parameter, 597
ssl_library#######, 597
ssl_max_protocol_version configuration parameter, 540
ssl_max_protocol_version#######, 540
ssl_min_protocol_version configuration parameter, 540
ssl_min_protocol_version#######, 540
ssl_passphrase_command configuration parameter, 540
ssl_passphrase_command_supports_reload configuration parameter, 540
ssl_passphrase_command_supports_reload#######, 540
ssl_passphrase_command#######, 540
ssl_prefer_server_ciphers configuration parameter, 539
ssl_prefer_server_ciphers#######, 539
ssl_version, 2528
ssl#######, 538
SSPI, 614
STABLE, 1058
standard deviation, 326
    population, 326
    sample, 327
standard_conforming_strings configuration parameter, 594
standard_conforming_strings#######, 594
standby server, 679
standby.signal, 554, 682, 683
    pg_basebackup --write-recovery-conf, 1832
    ############, 692
standby_mode (## standby.signal)
START TRANSACTION, 1792
starts_with, 206
START_REPLICATION, 2105
statement_timeout configuration parameter, 586
statement_timeout#######, 586
statement_timestamp, 242
statistics, 326, 697
    of the planner, 457, 459, 657
stats_fetch_consistency configuration parameter, 581
stats_fetch_consistency#######, 581
stddev, 326
stddev_pop, 326
stddev_samp, 327
STONITH, 679
storage parameters, 1587
Streaming Replication, 679
strict_word_similarity, 2498
string (## character string)
strings
    backslash quotes, 593
    escape warning, 593
    standard conforming, 594
string_agg, 324
string_to_array, 206
string_to_table, 206
strip, 264, 409
strpos, 206
subarray, 2437

subltree, 2448
subpath, 2448
subplan, 452
    hashed, 452
subquery, 12, 45, 105, 331
subscript, 39
substr, 206, 211
substring, 202, 210, 214, 216, 217
SUBSTRING_REGEX, 229
subtransactions
    in PL/Tcl, 1224
subtransaction_buffers configuration parameter, 543
sum, 325
summarize_wal configuration parameter, 556
superuser_reserved_connections configuration parameter, 536
superuser_reserved_connections#######, 536
support functions
    in_range, 2204
suppress_redundant_updates_trigger, 370
svals, 2429
synchronized_standby_slots configuration parameter, 558
synchronize_seqscans configuration parameter, 594
synchronize_seqscans#######, 594
synchronous commit, 754
Synchronous Replication, 679
synchronous_commit configuration parameter, 549
synchronous_commit#######, 549
synchronous_standby_names configuration parameter, 558
synchronous_standby_names#######, 558
sync_replication_slots configuration parameter, 562
syntax
    SQL, 30
syslog_facility configuration parameter, 571
syslog_facility#######, 571
syslog_ident configuration parameter, 571
syslog_ident#######, 571
syslog_sequence_numbers configuration parameter, 571
syslog_sequence_numbers#######, 571
syslog_split_messages configuration parameter, 571
syslog_split_messages#######, 571
system catalog
    schema, 79
systemd, 486, 496, 513
    RemoveIPC, 518
system_user, 340

T

table, 7, 52
    creating, 52
    inheritance, 80
    modifying, 64
    partitioning, 82
    removing, 53
    renaming, 66
Table Access Method, 2185
TABLE command, 1759
table expression, 100
table function, 105
    XMLTABLE, 275
table sampling method, 2174
tableam
    Table Access Method, 2185
    ############, 2185
tablefunc, 2530
tableoid, 63
TABLESAMPLE method, 2174
TABLESAMPLE####, 2174
tablespace, 632
    default, 584
    temporary, 584
table_am_handler, 186
TABLE####, 1759
tan, 199
tand, 199
tanh, 200
Tcl, 1217
tcn, 2539
tcp_keepalives_count configuration parameter, 537
tcp_keepalives_count#######, 537
tcp_keepalives_idle configuration parameter, 537
tcp_keepalives_idle#######, 537
tcp_keepalives_interval configuration parameter, 537
tcp_keepalives_interval#######, 537
tcp_user_timeout configuration parameter, 537
tcp_user_timeout#######, 537
template0, 630, 630
template1, 630, 630
temp_buffers configuration parameter, 541
temp_buffers#######, 541
temp_file_limit configuration parameter, 544
temp_file_limit#######, 544
temp_tablespaces configuration parameter, 584
temp_tablespaces#######, 584
test, 783
test_decoding, 2540
text, 132, 262
text search, 398
    data types, 152
    functions and operators, 152
    indexes, 428
text2ltree, 2449
threads
    with libpq, 872
tid, 184
time, 136, 138
    constants, 139
    current, 250
    output format, 140
        (## formatting)
    ####, 140
        (## ####)
    ##, 139

time span, 136
time with time zone, 136, 138
time without time zone, 136, 138
time zone, 141, 589
    conversion, 248
    input abbreviations, 2296
    POSIX-style specification, 2297
time zone data, 487, 498
time zone names, 589
timelines, 666
TIMELINE_HISTORY, 2103
timeofday, 242
timeout
    client authentication, 537
    deadlock, 592
    ########, 537
timestamp, 136, 139
timestamp with time zone, 136, 139
timestamp without time zone, 136, 139
timestamptz, 136
TimeZone configuration parameter, 589
timezone_abbreviations configuration parameter, 589
timezone_abbreviations#######, 589
TimeZone#######, 589
TOAST, 2253
    and user-defined types, 1089
    per-column storage settings, 1423, 1580
    versus large objects, 884
    #######, 1089
    ########, 1423, 1580
    ########, 1447
    ##########, 884
toast_tuple_target storage parameter, 1587
toast_tuple_target#######, 1587
token, 30
to_ascii, 206
to_bin, 206
to_char, 230
    #####, 635
to_date, 230
to_hex, 207
to_json, 284
to_jsonb, 284
to_number, 230
to_oct, 207
to_regclass, 347
to_regcollation, 347
to_regnamespace, 347
to_regoper, 347
to_regoperator, 347
to_regproc, 347
to_regprocedure, 347
to_regrole, 348
to_regtype, 348
to_regtypemod, 348
to_timestamp, 230, 242
to_tsquery, 264, 404
to_tsvector, 265, 403
trace_locks configuration parameter, 599
trace_locks#######, 599
trace_lock_oidmin configuration parameter, 599
trace_lock_oidmin#######, 599
trace_lock_table configuration parameter, 599
trace_lock_table#######, 599
trace_lwlocks configuration parameter, 599
trace_lwlocks#######, 599
trace_notify configuration parameter, 599
trace_notify#######, 599
trace_sort configuration parameter, 599
trace_sort#######, 599
trace_userlocks configuration parameter, 599
trace_userlocks#######, 599
track_activities configuration parameter, 580
track_activities#######, 580
track_activity_query_size configuration parameter, 580
track_activity_query_size#######, 580
track_commit_timestamp configuration parameter, 558
track_commit_timestamp#######, 558
track_counts configuration parameter, 580
track_counts#######, 580
track_functions configuration parameter, 581
track_functions#######, 581
track_io_timing configuration parameter, 580
track_io_timing#######, 580
track_wal_io_timing configuration parameter, 580
track_wal_io_timing#######, 580
transaction, 16
transaction ID
    wraparound, 658
transaction isolation, 433
transaction isolation level, 433, 585
    read committed, 434
    repeatable read, 435
    serializable, 436
    setting, 1787
    setting default, 585
transaction log (## WAL)
transaction_buffers configuration parameter, 543
transaction_deferrable configuration parameter, 585
transaction_deferrable#######, 585
transaction_isolation configuration parameter, 585
transaction_isolation#######, 585
transaction_read_only configuration parameter, 585
transaction_read_only#######, 585
transaction_timeout configuration parameter, 586
transaction_timestamp, 242
transform_null_equals configuration parameter, 594
transform_null_equals#######, 594
transition tables, 1609
    (## ephemeral named relation)
    implementation in PLs, 1294
    referencing from C trigger, 1117
translate, 207
TRANSLATE_REGEX, 229
transparent huge pages, 541

trigger, 186, 1115
    arguments for trigger functions, 1117
    constraint trigger, 1610
    for updating a derived tsvector column, 412
    in C, 1117
    in PL/pgSQL, 1195
    in PL/Python, 1247
    in PL/Tcl, 1221
    compared with rules, 1154
triggered_change_notification, 2539
trim, 202, 210
trim_array, 316
trim_scale, 196
true, 144
trunc, 197, 262
TRUNCATE, 1793
trusted
    PL/Perl, 1236
tsm_handler, 186
tsm_system_rows, 2541
tsm_system_time, 2542
tsquery (data type), 153
tsquery_phrase, 266, 410
tsquery######, 153
tsvector (data type), 152
tsvector concatenation, 409
tsvector_to_array, 266
tsvector_update_trigger, 370
tsvector_update_trigger_column, 371
tsvector###, 409
tsvector######, 152
ts_debug, 267, 424
ts_delete, 265
ts_filter, 265
ts_headline, 265, 408
ts_lexize, 267, 427
ts_parse, 267, 426
ts_rank, 266, 406
ts_rank_cd, 266, 406
ts_rewrite, 266, 411
ts_stat, 268, 413
ts_token_type, 267, 426
tuple_data_split, 2454
txid_current, 352
txid_current_if_assigned, 352
txid_current_snapshot, 352
txid_snapshot_xip, 352
txid_snapshot_xmax, 352
txid_snapshot_xmin, 352
txid_status, 353
txid_visible_in_snapshot, 352
type (## data type)
type cast, 44
typedef
    in ECPG, 907

U

UESCAPE, 31, 33
unaccent, 2543, 2544
Unicode escape
    in identifiers, 31
    in string constants, 33
Unicode normalization, 200, 201
unicode_assigned, 202
unicode_version, 355
Unicode#####
    ######, 33
    ####, 31
UNION, 115
    determination of result type, 382
    ######, 382
uniq, 2437
unique constraint, 58
unistr, 207
unix_socket_directories configuration parameter, 536
unix_socket_directories#######, 536
unix_socket_group configuration parameter, 536
unix_socket_group#######, 536
unix_socket_permissions configuration parameter, 536
unix_socket_permissions#######, 536
Unix########, 805
unknown, 186
UNLISTEN, 1795
unnest, 316
    for multirange, 322
    for tsvector, 267
unqualified name, 77
updatable views, 1627
UPDATE, 14, 98, 1797
    RETURNING, 99
update_process_title configuration parameter, 580
update_process_title#######, 580
updating, 98
upgrading, 522
UPLOAD_MANIFEST, 2107
upper, 202, 320, 321
    #####, 635
upper_inc, 321, 321
upper_inf, 321, 321
UPSERT, 1709
URI, 803
user, 340, 622
    current, 339
user mapping, 94
User name maps, 610
user_catalog_table storage parameter, 1589
user_catalog_table#######, 1589
UUID, 154, 486, 496
    generating, 154
    ##, 154
uuid-ossp, 2545
uuid_extract_timestamp, 268
uuid_extract_version, 268
uuid_generate_v1, 2545
uuid_generate_v1mc, 2545
uuid_generate_v3, 2545

V

vacuum, 656
VACUUM, 1802
vacuumdb, 1948
vacuumlo, 2557
vacuum_buffer_usage_limit configuration parameter, 542
vacuum_buffer_usage_limit#######, 542
vacuum_cost_delay configuration parameter, 545
vacuum_cost_delay#######, 545
vacuum_cost_limit configuration parameter, 545
vacuum_cost_limit#######, 545
vacuum_cost_page_dirty configuration parameter, 545
vacuum_cost_page_dirty#######, 545
vacuum_cost_page_hit configuration parameter, 545
vacuum_cost_page_hit#######, 545
vacuum_cost_page_miss configuration parameter, 545
vacuum_cost_page_miss#######, 545
vacuum_failsafe_age configuration parameter, 587
vacuum_failsafe_age#######, 587
vacuum_freeze_min_age configuration parameter, 587
vacuum_freeze_min_age#######, 587
vacuum_freeze_table_age configuration parameter, 587
vacuum_freeze_table_age#######, 587
vacuum_index_cleanup storage parameter, 1587
vacuum_index_cleanup#######, 1587
vacuum_multixact_failsafe_age configuration parameter, 588
vacuum_multixact_failsafe_age#######, 588
vacuum_multixact_freeze_min_age configuration parameter, 588
vacuum_multixact_freeze_min_age#######, 588
vacuum_multixact_freeze_table_age configuration parameter, 587
vacuum_multixact_freeze_table_age#######, 587
vacuum_truncate storage parameter, 1588
vacuum_truncate#######, 1588
value expression, 38
VALUES, 117, 1806
  determination of result type, 382
  ######, 382
varchar, 132
variadic function, 1050
variadic##, 1050
variance, 327
  population, 327
  sample, 327
var_pop, 327
var_samp, 327
version, 5, 355
  compatibility, 522
view, 15
  implementation through rules, 1134
  materialized, 1140
  updating, 1147
Visibility Map, 2256
Visual Studio
  installation on, 507
VM (## Visibility Map) (## ######)
void, 186
VOLATILE, 1058
volatility
  functions, 1058
VPATH, 481, 1113

W

WAL, 752
wal_block_size configuration parameter, 597
wal_block_size#######, 597
wal_buffers configuration parameter, 551
wal_buffers#######, 551
wal_compression configuration parameter, 550
wal_compression#######, 550
wal_consistency_checking configuration parameter, 600
wal_consistency_checking#######, 600
wal_debug configuration parameter, 600
wal_debug#######, 600
wal_decode_buffer_size configuration parameter, 554
wal_decode_buffer_size#######, 554
wal_init_zero configuration parameter, 551
wal_init_zero#######, 551
wal_keep_size configuration parameter, 557
wal_keep_size#######, 557
wal_level configuration parameter, 548
wal_level#######, 548
wal_log_hints configuration parameter, 550
wal_log_hints#######, 550
wal_receiver_create_temp_slot configuration parameter, 560
wal_receiver_create_temp_slot#######, 560
wal_receiver_status_interval configuration parameter, 560
wal_receiver_status_interval#######, 560
wal_receiver_timeout configuration parameter, 561
wal_receiver_timeout #######, 561
wal_recycle configuration parameter, 551
wal_recycle#######, 551
wal_retrieve_retry_interval configuration parameter, 561
wal_retrieve_retry_interval#######, 561
wal_segment_size configuration parameter, 597
wal_segment_size#######, 597
wal_sender_timeout configuration parameter, 558
wal_sender_timeout#######, 558
wal_skip_threshold configuration parameter, 551
wal_skip_threshold#######, 551
wal_summary_keep_time configuration parameter, 556
wal_sync_method configuration parameter, 550
wal_sync_method#######, 550
wal_writer_delay configuration parameter, 551
wal_writer_delay#######, 551
wal_writer_flush_after configuration parameter, 551

wal_writer_flush_after #######, 551
warm standby, 679
websearch_to_tsquery, 264
WHERE, 108
where to log, 568
WHILE
    in PL/pgSQL, 1180
    PL/pgSQL####, 1181
width, 257
width_bucket, 197
window function, 17
    built-in, 329
    invocation, 42
    order of execution, 113
WITH
    in SELECT, 118, 1759
    SELECT####, 118, 1759
WITH CHECK OPTION, 1625
WITHIN GROUP, 40
witness server, 679
witness###, 679
word_similarity, 2498
work_mem configuration parameter, 542
work_mem#######, 542
wraparound
    of multixact IDs, 661
    of transaction IDs, 658

X

xid, 184
xid8, 184
xmax, 63
xmin, 63
XML, 155
XML export, 277
XML Functions, 268
XML option, 155, 588
xml2, 2547
xmlagg, 272, 325
xmlbinary configuration parameter, 588
xmlbinary#######, 588
xmlcomment, 269
xmlconcat, 269
xmlelement, 270
XMLEXISTS, 273
xmlforest, 271
xmloption configuration parameter, 588
xmloption#######, 588
xmlparse, 155
xmlpi, 271
xmlroot, 271
xmlserialize, 155
xmltable, 275
xmltext, 268
xml_is_well_formed, 273
xml_is_well_formed_content, 273
xml_is_well_formed_document, 273
XML#####, 588
XPath, 274
xpath_exists, 274
xpath_table, 2548
XQuery regular expressions, 229
XQuery####, 229
xslt_process, 2550

Y

yacc, 479

Z

zero_damaged_pages configuration parameter, 600
zero_damaged_pages#######, 600
zlib, 479, 487, 497