

[PostgreSQL BASICS] (CheatSheet)

1. Database Operations

- **Create a database:** `CREATE DATABASE dbname;`
- **Drop a database:** `DROP DATABASE dbname;`
- **List all databases:** `\l`

2. Connection and Security

- **Connect to a database:** `\c dbname`
- **List active connections:** `SELECT * FROM pg_stat_activity;`
- **Cancel a running query:** `SELECT pg_cancel_backend(pid);`
- **Terminate a connection:** `SELECT pg_terminate_backend(pid);`

3. Table Operations

- **Create a table with a serial primary key:** `CREATE TABLE tablename (id SERIAL PRIMARY KEY, column1 TYPE);`
- **Add a column with a default value:** `ALTER TABLE tablename ADD COLUMN column_name TYPE DEFAULT 'default_value';`
- **Rename a column:** `ALTER TABLE tablename RENAME COLUMN old_name TO new_name;`
- **Change a column's type:** `ALTER TABLE tablename ALTER COLUMN column_name TYPE new_type USING column_name::new_type;`
- **Create an unlogged table:** `CREATE UNLOGGED TABLE tablename (columns);`
(Improves performance by not writing to the WAL)

4. Indexes and Constraints

- **Create a GIN index:** `CREATE INDEX idx_name ON tablename USING GIN(column_name);`
- **Create a partial index:** `CREATE INDEX idx_name ON tablename (column_name) WHERE condition;`
- **Add a foreign key constraint:** `ALTER TABLE tablename ADD CONSTRAINT constraint_name FOREIGN KEY (column_name) REFERENCES other_table (column_name);`
- **Create an exclusion constraint:** `ALTER TABLE tablename ADD CONSTRAINT constraint_name EXCLUDE USING gist (column_name WITH operator);`

5. Advanced Data Types

- **Use of array data type:** `SELECT * FROM tablename WHERE array_column @> ARRAY[value];`
- **Query JSON data:** `SELECT * FROM tablename WHERE json_column->>'key' = 'value';`
- **Use of hstore for key-value pairs:** `SELECT * FROM tablename WHERE hstore_column -> 'key' = 'value';`
- **UUID generation:** `SELECT uuid_generate_v4();`

6. Text Search

- **Full-text search:** `SELECT * FROM tablename WHERE to_tsvector('english', text_column) @@ to_tsquery('english', 'search_query');`
- **Ranking search results:** `SELECT *, ts_rank_cd(to_tsvector('english', text_column), to_tsquery('english', 'search_query')) FROM tablename ORDER BY rank DESC;`

7. Advanced Querying

- **Common Table Expressions (CTE):** `WITH cte_name AS (SELECT * FROM table) SELECT * FROM cte_name;`
- **Window functions:** `SELECT column, AVG(column) OVER (PARTITION BY column2) FROM tablename;`
- **Lateral joins:** `SELECT * FROM table1, LATERAL (SELECT * FROM table2 WHERE table2.column = table1.column) AS subquery;`

8. System Administration

- **View running queries:** `SELECT * FROM pg_stat_activity WHERE state = 'active';`
- **Analyze a table:** `ANALYZE tablename;`
- **Vacuum a table:** `VACUUM (VERBOSE, ANALYZE) tablename;`
- **Configuration parameters:** `SHOW all;`

9. Functions and Procedures

- **Create a function:** `CREATE FUNCTION function_name() RETURNS return_type AS $$ BEGIN -- function_body; END; $$ LANGUAGE plpgsql;`
- **Create a procedure:** `CREATE PROCEDURE procedure_name() LANGUAGE plpgsql AS $$ BEGIN -- procedure_body; END; $$;`

- **Invoke a procedure:** `CALL procedure_name();`

10. Extensions and External Modules

- **Install an extension:** `CREATE EXTENSION IF NOT EXISTS extension_name;`
- **List available extensions:** `SELECT * FROM pg_available_extensions;`
- **Use PostGIS for geographic information systems:** `CREATE EXTENSION postgis;`

11. Backup and Restore

- **Backup a database:** `pg_dump dbname > outfile`
- **Restore a database:** `pg_restore -d dbname infile`

12. Replication and High Availability

- **Setting up streaming replication:** *(Requires configuration in `postgresql.conf` and `pg_hba.conf`)*
- **Monitoring replication lag:** `SELECT pg_last_wal_receive_lsn() - pg_last_wal_replay_lsn() AS lag;`

13. Partitioning

- **Create a range partition:** `CREATE TABLE parent_table (id INT NOT NULL, created_at TIMESTAMPTZ NOT NULL) PARTITION BY RANGE (created_at);`
- **Add a partition:** `CREATE TABLE child_table PARTITION OF parent_table FOR VALUES FROM ('2024-01-01') TO ('2024-02-01');`

14. Monitoring and Performance Tuning

- **View slow queries:** `SELECT * FROM pg_stat_statements WHERE total_time / calls > 100 ORDER BY total_time DESC;`
- **Check table size:** `SELECT pg_size_pretty(pg_total_relation_size('tablename'));`

15. Logical Replication and Streaming

- **Set up a publication:** `CREATE PUBLICATION my_publication FOR TABLE mytable;`
- **Set up a subscription:** `CREATE SUBSCRIPTION my_subscription CONNECTION 'dbname=mydb host=myhost' PUBLICATION my_publication;`

16. Advanced Extensions

- **TimescaleDB for time-series data:** `CREATE EXTENSION IF NOT EXISTS timescaledb CASCADE;`
- **Citus for horizontal scaling:** `CREATE EXTENSION citus;`

17. JSON Operations

- **Query nested JSON:** `SELECT json_column->'nested'->>'key' FROM tablename;`
- **Update JSON value:** `UPDATE tablename SET json_column = jsonb_set(json_column, '{key}', '"new_value"') WHERE id = 1;`
- **Aggregate JSON objects:** `SELECT jsonb_agg(json_column) FROM tablename;`

18. Range Types

- **Create range column:** `CREATE TABLE range_table (range_col INT4RANGE NOT NULL);`
- **Query overlapping ranges:** `SELECT * FROM range_table WHERE range_col && '[2024-01-01,2024-01-31]'::daterange;`

19. Advisory Locks

- **Obtain an advisory lock:** `SELECT pg_try_advisory_lock(12345);`
- **Release an advisory lock:** `SELECT pg_advisory_unlock(12345);`

20. Performance Diagnostics

- **Explain analyze a query:** `EXPLAIN ANALYZE SELECT * FROM tablename;`
- **Check index usage:** `SELECT * FROM pg_stat_user_indexes;`
- **View lock contention:** `SELECT * FROM pg_locks WHERE not granted;`

21. Security and Permissions

- **Grant table access:** `GRANT SELECT ON tablename TO user_role;`
- **Revoke table access:** `REVOKE SELECT ON tablename FROM user_role;`
- **List user roles:** `\du`

22. Tablespaces

- **Create a tablespace:** `CREATE TABLESPACE fastspace LOCATION '/ssd1/postgresql/data';`
- **Assign a table to a tablespace:** `CREATE TABLE mytable (id SERIAL) TABLESPACE fastspace;`

23. Foreign Data Wrappers

- **Access external data:** `CREATE EXTENSION postgres_fdw;`
- **Create a foreign server:** `CREATE SERVER foreign_server FOREIGN DATA WRAPPER postgres_fdw OPTIONS (host 'hostname', dbname 'dbname');`
- **Import a foreign schema:** `IMPORT FOREIGN SCHEMA public FROM SERVER foreign_server INTO local_schema;`

24. Large Objects

- **Create a large object:** `SELECT lo_create(0);`
- **Manage large object permissions:** `SELECT lo_manage(oid);`
- **Access large object data:** `SELECT lo_get(oid, offset, length);`

25. Event Triggers

- **Create an event trigger:** `CREATE EVENT TRIGGER my_trigger ON ddl_command_start EXECUTE FUNCTION my_function();`
- **List event triggers:** `\det`

26. Write-Ahead Logging (WAL)

- **View WAL files:** `SELECT * FROM pg_ls_waldir();`
- **Configure WAL level:** `ALTER SYSTEM SET wal_level = logical;`

27. Server Configuration

- **Reload configuration without restarting:** `SELECT pg_reload_conf();`
- **View runtime configuration:** `SELECT * FROM pg_settings;`
- **Adjust configuration settings:** `ALTER SYSTEM SET configuration_parameter = 'value';`

28. Backup and Point-in-Time Recovery (PITR)

- **Create a base backup:** `pg_basebackup -D /var/lib/postgresql/data -Fp -Xs -P`

- **Recover to a specific point in time:** `pg_restore --dbname=newdb --time='2024-02-25 14:00:00' backupfile`

29. Extensions for Analytic Functions

- **Install MADlib for machine learning:** `CREATE EXTENSION madlib;`
- **PL/Python for executing Python scripts:** `CREATE EXTENSION plpythonu;`

30. Miscellaneous

- **List all settings and their descriptions:** `SELECT name, setting, short_desc FROM pg_settings;`
- **Reset a sequence:** `ALTER SEQUENCE seq RESTART WITH 1;`
- **Find duplicate rows:** `SELECT column, COUNT(*) FROM tablename GROUP BY column HAVING COUNT(*) > 1;`