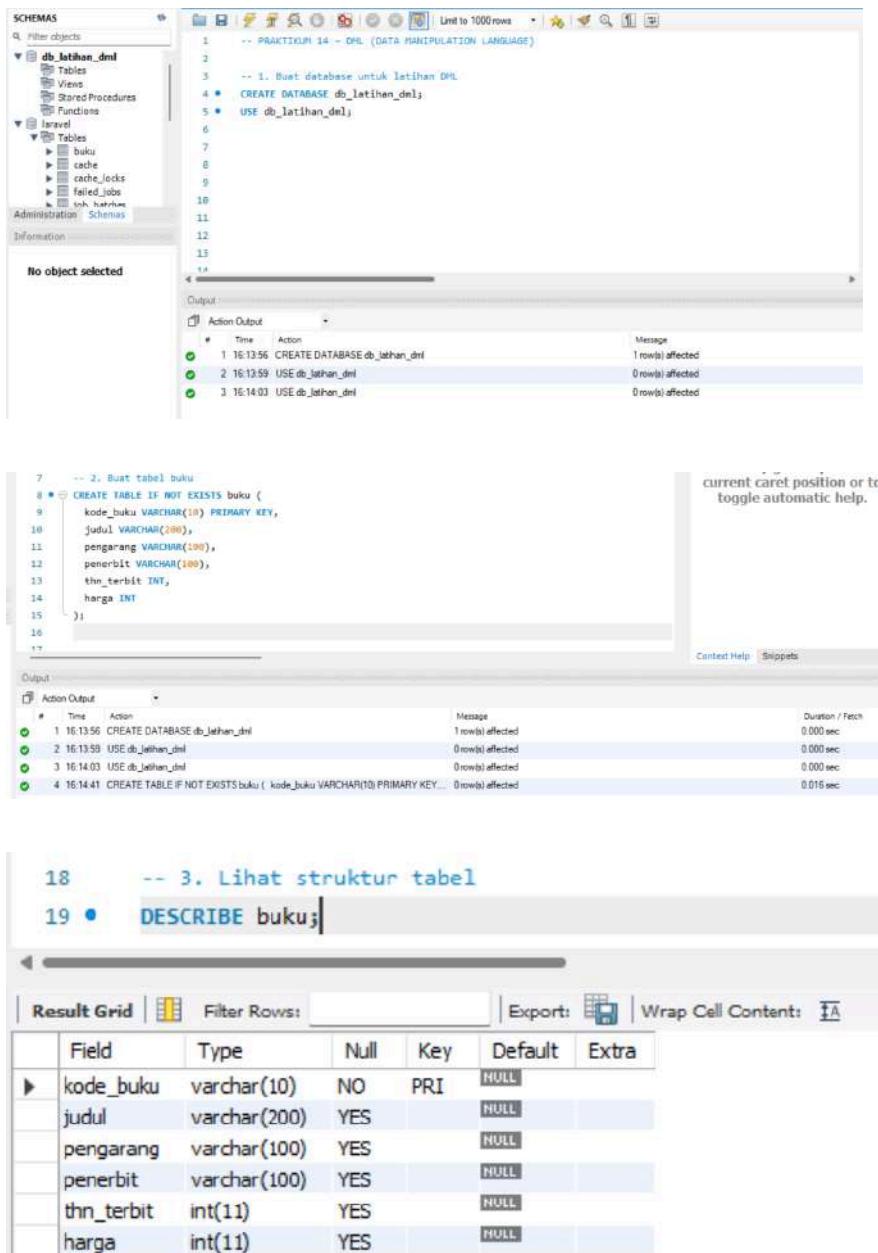


**TUGAS MANDIRI PERTEMUAN 14**  
**NAYLA PUTRI CAHYA RAMADANI**  
**2024071020**



The screenshot shows the MySQL Workbench interface. On the left, the 'SCHEMAS' tree shows the 'db\_latihan\_dml' schema selected. The central pane displays SQL code for creating a database and a table, with execution logs in the 'Output' tab. The bottom pane shows the structure of the 'buku' table in a 'Result Grid'.

```

-- PRAKTIKUM 14 - DML (DATA MANIPULATION LANGUAGE)
-- 1. Buat database untuk latihan DML
CREATE DATABASE db_latihan_dml;
USE db_latihan_dml;

-- 2. Buat tabel buku
CREATE TABLE IF NOT EXISTS buku (
    kode_buku VARCHAR(10) PRIMARY KEY,
    judul VARCHAR(200),
    pengarang VARCHAR(100),
    penerbit VARCHAR(100),
    thn_terbit INT,
    harga INT
);

-- 3. Lihat struktur tabel
DESCRIBE buku;

```

Output:

#	Time	Action	Message	Duration / Fetch
1	16:13:56	CREATE DATABASE db_latihan_dml	1 row(s) affected	0.000 sec
2	16:13:59	USE db_latihan_dml	0 row(s) affected	0.000 sec
3	16:14:03	USE db_latihan_dml	0 row(s) affected	0.000 sec
4	16:14:41	CREATE TABLE IF NOT EXISTS buku ( kode_buku VARCHAR(10) PRIMARY KEY...	0 row(s) affected	0.016 sec

Result Grid:

Field	Type	Null	Key	Default	Extra
kode_buku	varchar(10)	NO	PRI	NULL	
judul	varchar(200)	YES		NULL	
pengarang	varchar(100)	YES		NULL	
penerbit	varchar(100)	YES		NULL	
thn_terbit	int(11)	YES		NULL	
harga	int(11)	YES		NULL	

```

20      -- 4. Isi data tabel buku
21 •  INSERT INTO buku (kode_buku, judul, pengarang, penerbit, thn_terbit, harga) VALUES
22 ('BK01','Diagram UML','Penulis A','Graha Pustaka',2004,50000),
23 ('BK02','Basis Data','Dewi Lestari','Abadi Jaya',2003,45000),
24 ('BK03','Algoritma','Raden Kraton','Graha Pustaka',2006,60000),
25 ('BK04','Programming 1','Budi','Abadi Jaya',2001,35000),
26 ('BK05','Multimedia','Siti','Pustaka Kita',2007,30000);
27
28

```

Output

Action Output	
#	Action
1	16:13:56 CREATE DATABASE db_latihan_dml
2	16:13:59 USE db_latihan_dml
3	16:14:03 USE db_latihan_dml
4	16:14:41 CREATE TABLE IF NOT EXISTS buku ( kode_buku VARCHAR(10) PRIMARY KE...
5	16:15:03 DESCRIBE buku
6	16:15:40 INSERT INTO buku (kode_buku, judul, pengarang, penerbit, thn_terbit, harga) VALUES ('BK01','Diagram UML','Penulis A','Graha Pustaka',2004,50000), ('BK02','Basis Data','Dewi Lestari','Abadi Jaya',2003,45000), ('BK03','Algoritma','Raden Kraton','Graha Pustaka',2006,60000), ('BK04','Programming 1','Budi','Abadi Jaya',2001,35000), ('BK05','Multimedia','Siti','Pustaka Kita',2007,30000); 5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0

```
29      -- 5. Tampilkan seluruh record descending harga
```

```
30 •  SELECT * FROM buku ORDER BY harga DESC;
```

Result Grid					
kode_buku	judul	pengarang	penerbit	thn_terbit	harga
BK03	Algoritma	Raden Kraton	Graha Pustaka	2006	60000
BK01	Diagram UML	Penulis A	Graha Pustaka	2004	50000
BK02	Basis Data	Dewi Lestari	Abadi Jaya	2003	45000
BK04	Programming 1	Budi	Abadi Jaya	2001	35000
BK05	Multimedia	Siti	Pustaka Kita	2007	30000
*	NULL	NULL	NULL	NULL	NULL

```
32      -- 6. Total harga
```

```
33 •  SELECT SUM(harga) AS total_harga FROM buku;
```

34

35

36

Result Grid	
total_harga	
▶	220000

```
35      -- 7. Buku termurah
```

```
36 •  SELECT * FROM buku WHERE harga = (SELECT MIN(harga) FROM buku);
```

37

38

39

Result Grid					
kode_buku	judul	pengarang	penerbit	thn_terbit	harga
BK05	Multimedia	Siti	Pustaka Kita	2007	30000
*	NULL	NULL	NULL	NULL	NULL

```

38      -- 8. Rata-rata harga
39 •   SELECT AVG(harga) AS rata_harga FROM buku;
40
41
42

```

Result Grid | Filter Rows: Export: Wrap

	rata_harga
▶	44000.0000

```

41      -- 9. Alias tabel bk
42 •   SELECT bk.judul, bk.penerbit, bk.harga FROM buku AS bkj

```

Result Grid | Filter Rows: Export: Wrap Cell Content:

	judul	penerbit	harga
▶	Diagram UML	Graha Pustaka	50000
	Basis Data	Abadi Jaya	45000
	Algoritma	Graha Pustaka	60000
	Programming 1	Abadi Jaya	35000
	Multimedia	Pustaka Kita	30000

```
44      -- 10. Jumlah data
```

```
45 •   SELECT COUNT(*) AS jumlah_data FROM buku;
```

Result Grid | Filter Rows: Export: Wrap

	jumlah_data
▶	5

```

47      -- 11. Update judul Diagram UML menjadi UML Dasar
48 •   UPDATE buku
49     SET judul = 'UML Dasar'
50     WHERE kode_buku = 'BK01';
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
709
710
711
712
713
714
715
716
717
718
719
719
720
721
722
723
724
725
726
727
728
729
729
730
731
732
733
734
735
736
737
738
739
739
740
741
742
743
744
745
746
747
748
749
749
750
751
752
753
754
755
756
757
758
759
759
760
761
762
763
764
765
766
767
768
769
769
770
771
772
773
774
775
776
777
778
779
779
780
781
782
783
784
785
786
787
787
788
789
789
790
791
792
793
794
795
796
797
797
798
799
799
800
801
802
803
804
805
806
807
808
809
809
810
811
812
813
814
815
816
817
817
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
837
838
839
839
840
841
842
843
844
845
846
847
847
848
849
849
850
851
852
853
854
855
856
857
857
858
859
859
860
861
862
863
864
865
866
867
867
868
869
869
870
871
872
873
874
875
876
877
877
878
879
879
880
881
882
883
884
885
886
887
887
888
889
889
890
891
892
893
894
895
896
897
897
898
899
899
900
901
902
903
904
905
906
907
908
909
909
910
911
912
913
914
915
915
916
917
917
918
919
919
920
921
922
923
924
925
926
927
927
928
929
929
930
931
932
933
934
935
936
937
937
938
939
939
940
941
942
943
944
945
946
946
947
948
948
949
949
950
951
952
953
954
955
956
957
957
958
959
959
960
961
962
963
964
965
966
966
967
968
968
969
969
970
971
972
973
974
975
976
977
977
978
979
979
980
981
982
983
984
985
986
987
987
988
989
989
990
991
992
993
994
995
996
997
997
998
999
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1009
1010
1011
1012
1013
1014
1015
1016
1016
1017
1018
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1026
1027
1028
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1036
1037
1038
1038
1039
1039
1040
1041
1042
1043
1044
1045
1045
1046
1047
1047
1048
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1056
1057
1058
1058
1059
1059
1060
1061
1062
1063
1064
1065
1065
1066
1067
1067
1068
1068
1069
1069
1070
1071
1072
1073
1074
1075
1075
1076
1077
1077
1078
1078
1079
1079
1080
1081
1082
1083
1084
1085
1085
1086
1087
1087
1088
1088
1089
1089
1090
1091
1092
1093
1094
1094
1095
1096
1096
1097
1097
1098
1098
1099
1099
1100
1101
1102
1103
1104
1104
1105
1106
1106
1107
1107
1108
1109
1109
1110
1110
1111
1112
1112
1113
1113
1114
1114
1115
1115
1116
1116
1117
1117
1118
1118
1119
1119
1120
1120
1121
1121
1122
1122
1123
1123
1124
1124
1125
1125
1126
1126
1127
1127
1128
1128
1129
1129
1130
1130
1131
1131
1132
1132
1133
1133
1134
1134
1135
1135
1136
1136
1137
1137
1138
1138
1139
1139
1140
1140
1141
1141
1142
1142
1143
1143
1144
1144
1145
1145
1146
1146
1147
1147
1148
1148
1149
1149
1150
1150
1151
1151
1152
1152
1153
1153
1154
1154
1155
1155
1156
1156
1157
1157
1158
1158
1159
1159
1160
1160
1161
1161
1162
1162
1163
1163
1164
1164
1165
1165
1166
1166
1167
1167
1168
1168
1169
1169
1170
1170
1171
1171
1172
1172
1173
1173
1174
1174
1175
1175
1176
1176
1177
1177
1178
1178
1179
1179
1180
1180
1181
1181
1182
1182
1183
1183
1184
1184
1185
1185
1186
1186
1187
1187
1188
1188
1189
1189
1190
1190
1191
1191
1192
1192
1193
1193
1194
1194
1195
1195
1196
1196
1197
1197
1198
1198
1199
1199
1200
1200
1201
1201
1202
1202
1203
1203
1204
1204
1205
1205
1206
1206
1207
1207
1208
1208
1209
1209
1210
1210
1211
1211
1212
1212
1213
1213
1214
1214
1215
1215
1216
1216
1217
1217
1218
1218
1219
1219
1220
1220
1221
1221
1222
1222
1223
1223
1224
1224
1225
1225
1226
1226
1227
1227
1228
1228
1229
1229
1230
1230
1231
1231
1232
1232
1233
1233
1234
1234
1235
1235
1236
1236
1237
1237
1238
1238
1239
1239
1240
1240
1241
1241
1242
1242
1243
1243
1244
1244
1245
1245
1246
1246
1247
1247
1248
1248
1249
1249
1250
1250
1251
1251
1252
1252
1253
1253
1254
1254
1255
1255
1256
1256
1257
1257
1258
1258
1259
1259
1260
1260
1261
1261
1262
1262
1263
1263
1264
1264
1265
1265
1266
1266
1267
1267
1268
1268
1269
1269
1270
1270
1271
1271
1272
1272
1273
1273
1274
1274
1275
1275
1276
1276
1277
1277
1278
1278
1279
1279
1280
1280
1281
1281
1282
1282
1283
1283
1284
1284
1285
1285
1286
1286
1287
1287
1288
1288
1289
1289
1290
1290
1291
1291
1292
1292
1293
1293
1294
1294
1295
1295
1296
1296
1297
1297
1298
1298
1299
1299
1300
1300
1301
1301
1302
1302
1303
1303
1304
1304
1305
1305
1306
1306
1307
1307
1308
1308
1309
1309
1310
1310
1311
1311
1312
1312
1313
1313
1314
1314
1315
1315
1316
1316
1317
1317
1318
1318
1319
1319
1320
1320
1321
1321
1322
1322
1323
1323
1324
1324
1325
1325
1326
1326
1327
1327
1328
1328
1329
1329
1330
1330
1331
1331
1332
1332
1333
1333
1334
1334
1335
1335
1336
1336
1337
1337
1338
1338
1339
1339
1340
1340
1341
1341
1342
1342
1343
1343
1344
1344
1345
1345
1346
1346
1347
1347
1348
1348
1349
1349
1350
1350
1351
1351
1352
1352
1353
1353
1354
1354
1355
1355
1356
1356
1357
1357
1358
1358
1359
1359
1360
1360
1361
1361
1362
1362
1363
1363
1364
1364
1365
1365
1366
1366
1367
1367
1368
1368
1369
1369
1370
1370
1371
1371
1372
1372
1373
1373
1374
1374
1375
1375
1376
1376
1377
1377
1378
1378
1379
1379
1380
1380
1381
1381
1382
1382
1383
1383
1384
1384
1385
1385
1386
1386
1387
1387
1388
1388
1389
1389
1390
1390
1391
1391
1392
1392
1393
1393
1394
1394
1395
1395
1396
1396
1397
1397
1398
1398
1399
1399
1400
1400
1401
1401
1402
1402
1403
1403
1404
1404
1405
1405
1406
1406
1407
1407
1408
1408
1409
1409
1410
1410
1411
1411
1412
1412
1413
1413
1414
1414
1415
1415
1416
1416
1417
1417
1418
1418
1419
1419
1420
1420
1421
1421
1422
1422
1423
1423
1424
1424
1425
1425
1426
1426
1427
1427
1428
1428
1429
1429
1430
1430
1431
1431
1432
1432
1433
1433
1434
1434
1435
1435
1436
1436
1437
1437
1438
1438
1439
1439
1440
1440
1441
1441
1442
1442
1443
1443
1444
1444
1445
1445
1446
1446
1447
1447
1448
1448
1449
1449
1450
1450
1451
1451
1452
1452
1453
1453
1454
1454
1455
1455
1456
1456
1457
1457
1458
1458
1459
1459
1460
1460
1461
1461
1462
1462
1463
1463
1464
1464
1465
1465
1466
1466
1467
1467
1468
1468
1469
1469
1470
1470
1471
1471
1472
1472
1473
1473
1474
1474
1475
1475
1476
1476
1477
1477
1478
1478
1479
1479
1480
1480
1481
1481
1482
1482
1483
1483
1484
1484
1485
1485
1486
1486
1487
1487
1488
1488
1489
1489
1490
1490
1491
1491
1492
1492
1493
1493
1494
1494
1495
1495
1496
1496
1497
1497
1498
1498
1499
1499
1500
1500
1501
1501
1502
1502
1503
1503
1504
1504
1505
1505
1506
1506
1507
1507
1508
1508
1509
1509
1510
1510
1511
1511
1512
1512
1513
1513
1514
1514
1515
1515
1516
1516
1517
1517
1518
1518
1519
1519
1520
1520
1521
1521
1522
1522
1523
1523
1524
1524
1525
1525
1526
1526
1527
1527
1528
1528
1529
1529
1530
1530
1531
1531
1532
1532
1533
1533
1534
1534
1535
1535
1536
1536
1537
1537
1538
1538
1539
1539
1540
1540
1541
1541
1542
1542
1543
1543
1544
1544
1545
1545
1546
1546
1547
1547
1548
1548
1549
1549
1550
1550
1551
1551
1552
1552
1553
1553
1554
1554
1555
1555
1556
1556
1557
1557
1558
1558
1559
1559
1560
1560
1561
1561
1562
1562
1563
1563
1564
1564
1565
1565
1566
1566
1567
1567
1568
1568
1569
1569
1570
1570
1571
1571
1572
1572
1573
1573
1574
1574
1575
1575
1576
1576
1577
1577
1578
1578
1579
1579
1580
1580
1581
1581
1582
1582
1583
1583
1584
1584
1585
1585
1586
1586
1587
1587
1588
1588
1589
1589
1590
1590
1591
1591
1592
1592
1593
1593
1594
1594
1595
1595
1596
1596
1597
1597
1598
1598
1599
1599
1600
1600
1601
1601
1602
1602
1603
1603
1604
1604
1605
1605
1606
1606
1607
1607
1608
1608
1609
1609
1610
1610
1611
1611
1612
1612
1613
1613
1614
1614
1615
1615
1616
1616
1617
1617
1618
1618
1619
1619
1620
1620
1621
1621
1622
1622
1623
1623
1624
1624
1625
1625
1626
1626
1627
1627
1628
1628
1629
1629
1630
1630
1631
1631
1632
1632
1633
1633
1634
1634
1635
1635
1636
1636
1637
1637
1638
163
```

55 -- 12. Insert BK05

56 • INSERT INTO buku (kode\_buku, judul, pengarang, penerbit, thn\_terbit, harga)  
VALUES ('BK05','Algoritma Lanjut','Raden Iratno','Graha Pustaka',2000,40000);

57

58

Output

Action Output

#	Time	Action	Message
0	13 16:19:17	UPDATE buku SET judul = 'UML Dasar' WHERE judul = 'Diagram UML'	Error Code: 1175. You are using safe update mode and you tried to update a table ...
1	14 16:19:59	UPDATE buku SET judul = 'UML Dasar' WHERE judul = 'Diagram UML'	Error Code: 1175. You are using safe update mode and you tried to update a table ...
2	15 16:20:12	SELECT * FROM buku WHERE judul = 'Diagram UML' LIMIT 0, 1000	1 row(s) returned
3	16 16:20:47	UPDATE buku SET judul = 'UML Dasar' WHERE judul = 'Diagram UML'	Error Code: 1175. You are using safe update mode and you tried to update a table ...
4	17 16:21:01	UPDATE buku SET judul = 'UML Dasar' WHERE kode_buku = 'BK01'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
5	18 16:21:44	INSERT INTO buku (kode_buku, judul, pengarang, penerbit, thn_jrbl, harga) VALUES ('BK05')	1 row(s) affected

59 -- 13. Hapus BK05 dan tampilkan isi tabel

60 • DELETE FROM buku WHERE kode\_buku = 'BK05';

61 • SELECT \* FROM buku;

62

63

64

Output

Action Output

#	Time	Action	Message	Duration / Fetch
0	14 16:19:59	UPDATE buku SET judul = 'UML Dasar' WHERE judul = 'Diagram UML'	Error Code: 1175. You are using safe update mode and you tried to update a table ...	0.000 sec
1	15 16:20:12	SELECT * FROM buku WHERE judul = 'Diagram UML' LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec
2	16 16:20:47	UPDATE buku SET judul = 'UML Dasar' WHERE judul = 'Diagram UML'	Error Code: 1175. You are using safe update mode and you tried to update a table ...	0.000 sec
3	17 16:21:01	UPDATE buku SET judul = 'UML Dasar' WHERE kode_buku = 'BK01'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0	0.015 sec
4	18 16:21:44	INSERT INTO buku (kode_buku, judul, pengarang, penerbit, thn_jrbl, harga) VALUES ('BK05')	1 row(s) affected	0.000 sec
5	19 16:22:06	DELETE FROM buku WHERE kode_buku = 'BK05'	1 row(s) affected	0.000 sec

67 -- PRAKTIKUM 15 - OPERATOR

68

69 -- Buat database latihan operator

70 • CREATE DATABASE db\_latihan\_operator;

71 • USE db\_latihan\_operator;

72 -- Buat tabel member

73 • CREATE TABLE IF NOT EXISTS member (

74     id\_member INT AUTO\_INCREMENT PRIMARY KEY,

75     nama\_member VARCHAR(200),

76     semester INT,

77     usia INT,

78     alamat VARCHAR(200)

79 );

80

Output

Action Output

#	Time	Action	Message
0	17 16:21:01	UPDATE buku SET judul = 'UML Dasar' WHERE kode_buku = 'BK01'	1 row(s) affected Rows matched: 1 Changed: 1 Warnings: 0
1	18 16:21:44	INSERT INTO buku (kode_buku, judul, pengarang, penerbit, thn_jrbl, harga) VALUES ('BK05')	1 row(s) affected
2	19 16:22:06	DELETE FROM buku WHERE kode_buku = 'BK05'	1 row(s) affected
3	20 16:22:39	CREATE DATABASE db_latihan_operator	1 row(s) affected
4	21 16:22:42	USE db_latihan_operator	0 row(s) affected
5	22 16:22:54	CREATE TABLE IF NOT EXISTS member ( id_member INT AUTO_INCREMENT ...	0 row(s) affected

85 ('Rani',1,18,'Jl. Rambutan'),

86 ('Rino',5,22,'Jl. Mangga');

87

88

89 -- 7. Tampilkan seluruh record

90 • SELECT \* FROM member;

Result Grid

Result Grid   Filter Rows:   Edit:   Export/Import:   Wrap Cell Content:				
id_member	nama_member	semester	usia	alamat
1	Eni Susanti	3	20	Jl. Melati
2	Rere Kurnia	2	19	Jl. Mawar
3	Andi Pratama	4	21	Jl. Kenanga
4	Rani	1	18	Jl. Rambutan
5	Rino	5	22	Jl. Mangga
	HULL	HULL	HULL	HULL

member 9 ×

Output

Action Output

#	Time	Action	Message
0	19 16:22:06	DELETE FROM buku WHERE kode_buku = 'BK05'	1 row(s) affected
1	20 16:22:39	CREATE DATABASE db_latihan_operator	1 row(s) affected
2	21 16:22:42	USE db_latihan_operator	0 row(s) affected
3	22 16:22:54	CREATE TABLE IF NOT EXISTS member ( id_member INT AUTO_INCREMENT ...	0 row(s) affected
4	23 16:23:13	INSERT INTO member (nama_member, semester, usia, alamat) VALUES ('Emi Sus...', 'Rini', 5, 'Jl. Mangga')	5 row(s) affected Records: 5 Duplicates: 0 Warnings: 0
5	24 16:23:18	SELECT * FROM member LIMIT 0, 1000	5 row(s) returned

```
92      -- 8. Nama = Erni Susanti  
93 •   SELECT * FROM member WHERE nama_member = 'Erni Susanti';
```

	id_member	nama_member	semester	usia	alamat
▶	1 NULL	Erni Susanti NULL	3 NULL	20 NULL	Jl. Melati NULL

```
95      -- 9. Usia < 21  
96 •   SELECT * FROM member WHERE usia < 21;
```

	id_member	nama_member	semester	usia	alamat
▶	1	Erni Susanti	3	20	Jl. Melati
	2	Rere Kurnia	2	19	Jl. Mawar
*	4 NULL	Rani NULL	1 NULL	18 NULL	Jl. Rambutan NULL

```
98      -- 10. Kecuali Rere Kurnia  
99 •   SELECT * FROM member WHERE nama_member <> 'Rere Kurnia';
```

	id_member	nama_member	semester	usia	alamat
▶	1	Erni Susanti	3	20	Jl. Melati
	3	Andi Pratama	4	21	Jl. Kenanga
	4	Rani	1	18	Jl. Rambutan
*	5 NULL	Rino NULL	5 NULL	22 NULL	Jl. Mangga NULL

```
101     -- 11. Field nama_member, semester, urut semester  
102 •   SELECT nama_member, semester FROM member ORDER BY semester;
```

	nama_member	semester
▶	Rani	1
	Rere Kurnia	2
	Erni Susanti	3
	Andi Pratama	4
	Rino	5

```
104      -- 12. Usia between 19-20
```

```
105 •  SELECT nama_member, semester, usia FROM member WHERE usia BETWEEN 19 AND 20;
```

	nama_member	semester	usia
▶	Erni Susanti	3	20
▶	Rere Kurnia	2	19

```
107      -- 13. usia > 18 AND semester > 2
```

```
108 •  SELECT * FROM member WHERE usia > 18 AND semester > 2;
```

	id_member	nama_member	semester	usia	alamat
▶	1	Erni Susanti	3	20	Jl. Melati
▶	3	Andi Pratama	4	21	Jl. Kenanga
▶	5	Rino	5	22	Jl. Mangga
*	NULL	NULL	NULL	NULL	NULL

```
110      -- 14. usia > 18 OR semester > 2
```

```
111 •  SELECT * FROM member WHERE usia > 18 OR semester > 2;
```

	id_member	nama_member	semester	usia	alamat
▶	1	Erni Susanti	3	20	Jl. Melati
▶	2	Rere Kurnia	2	19	Jl. Mawar
▶	3	Andi Pratama	4	21	Jl. Kenanga
▶	5	Rino	5	22	Jl. Mangga
*	NULL	NULL	NULL	NULL	NULL

```
113      -- 15. Nama depan huruf R
```

```
114 •  SELECT nama_member, alamat FROM member WHERE nama_member LIKE 'R%' ORDER BY nama_member;
```

	nama_member	alamat
▶	Rani	Jl. Rambutan
▶	Rere Kurnia	Jl. Mawar
▶	Rino	Jl. Mangga

```
116      -- 16. usia > 18, urut nama desc
```

```
117 •  SELECT nama_member, alamat, usia FROM member WHERE usia > 18 ORDER BY nama_member DESC;
```

	nama_member	alamat	usia
▶	Rino	Jl. Mangga	22
▶	Rere Kurnia	Jl. Mawar	19
▶	Erni Susanti	Jl. Melati	20
▶	Andi Pratama	Jl. Kenanga	21

```
119      -- 17. LIMIT 4
120 •  SELECT * FROM member LIMIT 4;
121
```

	id_member	nama_member	semester	usia	alamat
▶	1	Erni Susanti	3	20	Jl. Melati
	2	Rere Kurnia	2	19	Jl. Mawar
	3	Andi Pratama	4	21	Jl. Kenanga
*	4	Rani	1	18	Jl. Rambutan
	NULL	NULL	NULL	NULL	NULL

```
122      -- 18. LIMIT 5 urut semester desc
123 •  SELECT * FROM member ORDER BY semester DESC LIMIT 5;
124
```

	id_member	nama_member	semester	usia	alamat	Export/Import:
▶	5	Rino	5	22	Jl. Mangga	
	3	Andi Pratama	4	21	Jl. Kenanga	
	1	Erni Susanti	3	20	Jl. Melati	
	2	Rere Kurnia	2	19	Jl. Mawar	
*	4	Rani	1	18	Jl. Rambutan	
	NULL	NULL	NULL	NULL	NULL	

```
125      -- 19. Record ke-2 sampai ke-4
126 •  SELECT * FROM member LIMIT 1,3;
127
```

	id_member	nama_member	semester	usia	alamat
▶	2	Rere Kurnia	2	19	Jl. Mawar
	3	Andi Pratama	4	21	Jl. Kenanga
	4	Rani	1	18	Jl. Rambutan
*	NULL	NULL	NULL	NULL	NULL

```

128      -- 20. Record ke-1 sampai ke-4 urut id_member
129 •   SELECT * FROM member ORDER BY id_member LIMIT 0,4;
130

```

Result Grid | Filter Rows: \_\_\_\_\_ | Edit: | Export

	id_member	nama_member	semester	usia	alamat
▶	1	Erni Susanti	3	20	Jl. Melati
	2	Rere Kurnia	2	19	Jl. Mawar
	3	Andi Pratama	4	21	Jl. Kenanga
*	4	Rani	1	18	Jl. Rambutan
	NULL	NULL	NULL	NULL	NULL

```

137      -- Buat database
138 •   CREATE DATABASE db_toko;
139 •   USE db_toko;
140
141      -- Buat tabel brg
142 •   CREATE TABLE IF NOT EXISTS brg (
143         kode_brg VARCHAR(6) PRIMARY KEY,
144         nama_brg VARCHAR(100),
145         stok INT,
146         harga_brg INT,
147         thn_pembuatan INT,
148         warna VARCHAR(30)
149     );
150

```

Action Output

#	Time	Action	Message
35	16:27:52	SELECT * FROM member ORDER BY semester DESC LIMIT 5	5 row(s) returned
36	16:28:12	SELECT * FROM member LIMIT 1,3	3 row(s) returned
37	16:28:32	SELECT * FROM member ORDER BY id_member LIMIT 0,4	4 row(s) returned
38	16:29:04	CREATE DATABASE db_toko	1 row(s) affected
39	16:29:06	USE db_toko	0 row(s) affected
40	16:29:08	CREATE TABLE IF NOT EXISTS brg ( kode_brg VARCHAR(6) PRIMARY KEY, ... )	0 row(s) affected

```

156     ('BR04','Kipas',90,90000,2000,'Biru'),
157     ('BR05','Monitor',50,450000,1999,'Hitam');
158
159      -- 6. Tampilkan semua
160 •   SELECT * FROM brg;

```

Result Grid | Filter Rows: \_\_\_\_\_ | Edit: | Export/Import: | W

	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna
▶	BR01	Terminal	150	120000	2003	Hitam
	BR02	Charger	300	25000	2005	Putih
	BR03	Cable	180	15000	2002	Merah
	BR04	Kipas	90	90000	2000	Biru
*	BR05	Monitor	50	450000	1999	Hitam
	NULL	NULL	NULL	NULL	NULL	NULL

```
162      -- 7. kode_brg, nama_brg, stok urut nama
163 •  SELECT kode_brg, nama_brg, stok FROM brg ORDER BY nama_brg ASC;
```

Result Grid			Filter Rows:	Edit:	Export/Import:
	kode_brg	nama_brg	stok		
▶	BR03	Cable	180		
▶	BR02	Charger	300		
▶	BR04	Kipas	90		
▶	BR05	Monitor	50		
▶	BR01	Terminal	150		
*	NULL	NULL	NULL		

```
165      -- 8. nama_barang = Terminal
166 •  SELECT * FROM brg WHERE nama_brg = 'Terminal';
```

Result Grid						Filter Rows:	Edit:
	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna	
▶	BR01	Terminal	150	120000	2003	Hitam	
*	NULL	NULL	NULL	NULL	NULL	NULL	

```
168      -- 9. nama_barang awalan C
169 •  SELECT * FROM brg WHERE nama_brg LIKE 'C%';
170
171      -- 10. stok < 200
172 •  SELECT kode_brg, nama_brg, harga_brg, stok FROM
```

Result Grid						Filter Rows:	Edit:
	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna	
▶	BR02	Charger	300	25000	2005	Putih	
▶	BR03	Cable	180	15000	2002	Merah	
*	NULL	NULL	NULL	NULL	NULL	NULL	

```
171      -- 10. stok < 200
172 •  SELECT kode_brg, nama_brg, harga_brg, stok FROM brg WHERE stok < 200;
```

Result Grid				Filter Rows:	Edit:	Export/Import:	Wr
	kode_brg	nama_brg	harga_brg	stok			
▶	BR01	Terminal	120000	150			
▶	BR03	Cable	15000	180			
▶	BR04	Kipas	90000	90			
▶	BR05	Monitor	450000	50			
*	NULL	NULL	NULL	NULL			

```
174      -- 11. Tahun 2002-2006
```

```
175 •  SELECT * FROM brg WHERE thn_pembuatan BETWEEN 2002 AND 2006;
```

	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna
▶	BR01	Terminal	150	120000	2003	Hitam
	BR02	Charger	300	25000	2005	Putih
*	BR03	Cable	180	15000	2002	Merah
*	HULL	NULL	NULL	NULL	NULL	NULL

```
177      -- 12. LIMIT 3
```

```
178 •  SELECT * FROM brg LIMIT 3;
```

	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna
▶	BR01	Terminal	150	120000	2003	Hitam
	BR02	Charger	300	25000	2005	Putih
*	BR03	Cable	180	15000	2002	Merah
*	HULL	NULL	NULL	NULL	NULL	NULL

```
180      -- 13. stok < 200 AND tahun 2000
```

```
181 •  SELECT * FROM brg WHERE stok < 200 AND thn_pembuatan = 2000;
```

	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna
▶	BR04	Kipas	90	90000	2000	Biru
*	HULL	NULL	NULL	NULL	NULL	NULL

```
183      -- 14. Record 2-4
```

```
184 •  SELECT * FROM brg ORDER BY kode_brg LIMIT 1,3;
```

	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna
▶	BR02	Charger	300	25000	2005	Putih
	BR03	Cable	180	15000	2002	Merah
*	BR04	Kipas	90	90000	2000	Biru
*	HULL	NULL	NULL	NULL	NULL	NULL

```
186      -- 15. kecuali BR05
187 •   SELECT * FROM brg WHERE kode_brg <> 'BR05';
```

	kode_brg	nama_brg	stok	harga_brg	thn_pembuatan	warna
▶	BR01	Terminal	150	120000	2003	Hitam
	BR02	Charger	300	25000	2005	Putih
	BR03	Cable	180	15000	2002	Merah
	BR04	Kipas	90	90000	2000	Biru
●	NULL	NULL	NULL	NULL	NULL	NULL

```
193      -- PRAKTIKUM 17 - AGREGASI
194
195 •   CREATE DATABASE db_dml_operator;
196 •   USE db_dml_operator;
197
198      -- Tabel pengajar
199 •   CREATE TABLE IF NOT EXISTS pengajar (
200          id_pengajar INT AUTO_INCREMENT PRIMARY KEY,
201          nama VARCHAR(100),
202          sks INT,
203          gaji INT,
204          kota_asal VARCHAR(50)
205      );
206
```

Action Output		
#	Time	Action
49	16:32:24	SELECT * FROM brg WHERE stok < 200 AND thn_pembuatan = 2000 LIMIT 0, 10...
50	16:32:42	SELECT * FROM brg ORDER BY kode_brg LIMIT 1,3
51	16:32:59	SELECT * FROM brg WHERE kode_brg <> 'BR05' LIMIT 0, 1000
52	16:33:28	CREATE DATABASE db_dml_operator
53	16:33:31	USE db_dml_operator
54	16:33:38	CREATE TABLE IF NOT EXISTS pengajar ( id_pengajar INT AUTO_INCREMEN...

```
211      ('Guru C',5,3500000,'Denpasar'),  
212      ('Guru D',2,2000000,'Gianyar');  
213  
214      -- 7. Semua record urut desc gaji  
215 •  SELECT * FROM pengajar ORDER BY gaji DESC;
```

Result Grid | Filter Rows: | Edit: |

	id_pengajar	nama	skls	gaji	kota_asal
▶	3	Guru C	5	3500000	Denpasar
	1	Guru A	4	3000000	Denpasar
	2	Guru B	3	2500000	Singaraja
*	4	Guru D	2	2000000	Gianyar
	HULL	HULL	HULL	HULL	HULL

```
217      -- 8. rata gaji  
218 •  SELECT AVG(gaji) AS rata_gaji FROM pengajar;
```

Result Grid | Filter Rows: | Export: | Wrap

rata_gaji
2750000.0000

```
220      -- 9. gaji terendah  
221 •  SELECT MIN(gaji) AS gaji_terendah FROM pengajar;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Co

gaji_terendah
2000000

```
223      -- 10. total gaji  
224 •  SELECT SUM(gaji) AS total_gaji FROM pengajar;
```

Result Grid | Filter Rows: | Export: | Wrap

total_gaji
11000000

```
226      -- 11. total gaji pengajar sks > 3
227 •   SELECT SUM(gaji) AS total_gaji_sks_lebih3 FROM pengajar WHERE sks > 3;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
total_gaji_sks_lebih3				

▶ 6500000

```
229      -- 12. distinct kota_asal
230 •   SELECT DISTINCT kota_asal FROM pengajar ORDER BY kota_asal;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
kota_asal				

▶ Denpasar  
Gianyar  
Singaraja

```
232      -- 13. alias rata_gaji
```

```
233 •   SELECT AVG(gaji) AS rata_gaji FROM pengajar;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
rata_gaji				

▶ 2750000.0000

```
235      -- 14. alias tabel tp
```

```
236 •   SELECT tp.nama, tp.sks, tp.gaji FROM pengajar AS tp;
```

Result Grid		Filter Rows:	Export:	Wrap Cell Content:
nama	sks	gaji		
Guru A	4	3000000		
Guru B	3	2500000		
Guru C	5	3500000		
Guru D	2	2000000		

```
238      -- 17. hasil = sks * gaji
239 •   SELECT nama, sks, gaji, (sks * gaji) AS hasil FROM pengajar;
```

	nama	sks	gaji	hasil
▶	Guru A	4	3000000	12000000
	Guru B	3	2500000	7500000
	Guru C	5	3500000	17500000
	Guru D	2	2000000	4000000

```
241      -- 18. bonus = sks * 100000
242 •   SELECT nama, sks, (sks * 100000) AS bonus FROM pengajar;
```

	nama	sks	bonus
▶	Guru A	4	400000
	Guru B	3	300000
	Guru C	5	500000
	Guru D	2	200000

```
244      -- 19. gaji terbesar
```

```
245 •   SELECT MAX(gaji) AS gaji_terbesar FROM pengajar;
```

	gaji_terbesar
▶	3500000

```
247      -- 20. tunjangan = sks * 250000
248 •   SELECT nama, sks, (sks * 250000) AS tunjangan FROM pengajar ORDER BY tunjangan DESC;
```

	nama	sks	tunjangan
▶	Guru C	5	1250000
	Guru A	4	1000000
	Guru B	3	750000
	Guru D	2	500000