

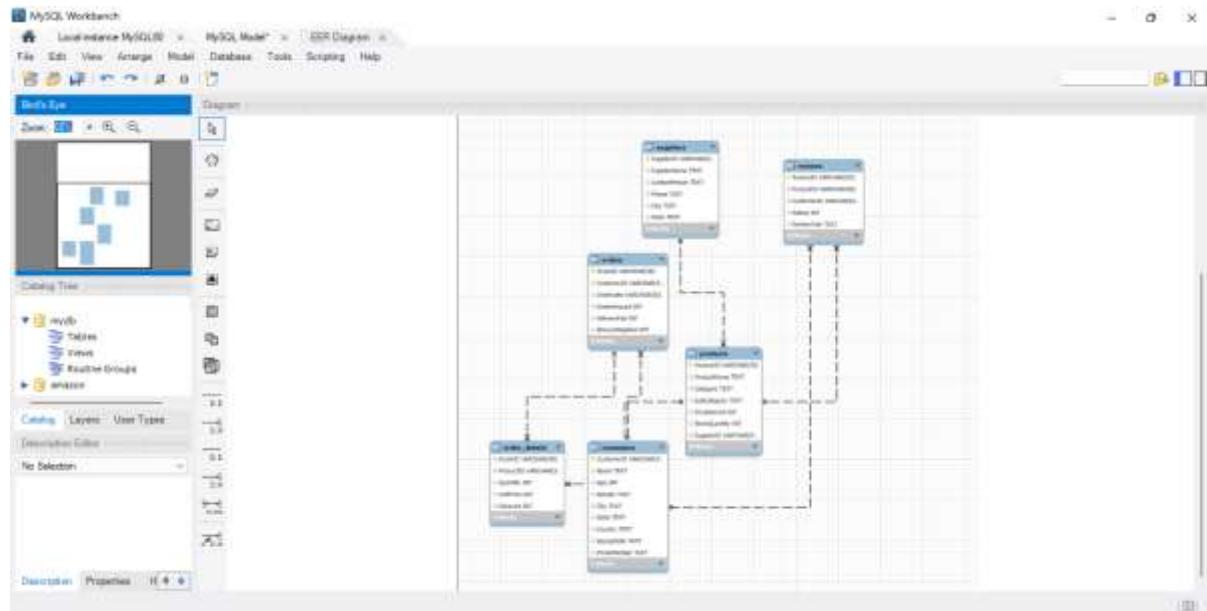
Amazon Fresh Analytics

Problem Statement:

Amazon Fresh is an e-commerce platform specializing in groceries and daily essentials. As the platform expands, its management seeks to optimize operations, enhance customer satisfaction, and boost revenue. The challenge lies in effectively managing large volumes of data, including customer profiles, product inventories, supplier information, orders, and customer reviews. The problem is to design a relational database for Amazon Fresh that organizes this data efficiently, allows for meaningful analysis, and supports critical business decisions. This project aims to solve key business problems, such as identifying top-performing products, analyzing customer purchasing patterns, and ensuring inventory management aligns with demand trends.

Task 1: ER Diagram Creation

The ER diagram was created in MySQL Workbench to represent the Amazon Fresh database structure. It includes entities such as Customers, Products, Orders, Suppliers, Order_Details, and Reviews, showing their relationships through primary and foreign keys.



Task 2: Identifying Keys and Relationships

In this step, primary keys and foreign keys were identified for each table. Relationships were defined — for example, CustomerID links Customers and Orders, while ProductID connects Products and Order_Details.

Task 3: Basic Select Queries

SQL queries were written to retrieve all customers from a specific city and to fetch all products under the 'Fruits' category. The results confirm correct filtering based on WHERE conditions.

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
File Test 1 Test 2
Schemas
Tables
order_details
Columns
OrderID int(10)
ProductID varchar(50)
Quantity int
UnitPrice float
Discount int
Table: order_details
Columns:
OrderID int(10)
ProductID varchar(50)
Quantity int
UnitPrice float
Discount int
Result Grid
Name City
John Smith West Elm
Heather Bennett West Elm

```

```

MySQL Workbench
File Edit View Query Database Server Tools Scripting Help
File Test 1 Test 2
Schemas
Tables
order_details
Columns
OrderID int(10)
ProductID varchar(50)
Quantity int
UnitPrice float
Discount int
Table: order_details
Columns:
OrderID int(10)
ProductID varchar(50)
Quantity int
UnitPrice float
Discount int
Result Grid
ProductID ProductName Category SubCategory PricePerUnit StockQuantity SupplierID
102460033 104-F-4669-a71e-(2)3528653069 Wood Fruit Fruits Sub-Fruit-4 130 396 11236290-0e3e-4bc7-8e04-c0830e13a46
09452037-40a7-4f59-b267-a3137285d548 Apple Fruit Fruits Sub-Fruit-3 887 33 75595ed0-9d2b-4793-a323-0886605affc3
61592794-434f-4636-8445-20838a5b2687 Community Fruit Fruits Sub-Fruit-3 459 461 12940219-9e79-4c36-a402-2a8a1893e49d
6c439cc-9b58-4251-8efc-39aa45b771b Card Fruit Fruits Sub-Fruit-4 515 309 70861a9-4ed9-4eca-859a-76b857423e03
6cf444d7-4207-434c-a61e-f4e2019340e6 Can Fruit Fruits Sub-Fruit-2 459 467 7126aa7-9b30-4c2b-b84b-29934034a4
1d85eb025-1030-4b2c-3-3780-ea5530567748 Pine Fruit Fruits Sub-Fruit-2 376 303 47053e1d-9117-4e1b-2259e35711349
07583666-6d3e-4449-8c75-49baa42632e8 Important Fruit Fruits Sub-Fruit-3 492 332 46499415-3801-4c75-b5d5-d03f5a4e4472
132ed774-0708-420f-a45-765ed53c198 Blue Fruit Fruits Sub-Fruit-2 8 230 076837B-0592-4ed4-a26c-87c330779d7
145239a4-4259-4511-633f-ea5030db7 Four Fruit Fruits Sub-Fruit-4 391 443 1294131-9114-4255-9-899826c5346
1755fe36-6d42-4513-bd46-078ef203c9e White Fruit Fruits Sub-Fruit-1 830 231 891523041-f994-4ef9-a29e-2b1530e7ae7
1c20386e-6d3e-4603-807-4631a1d1ac2 Yellow Fruit Fruits Sub-Fruit-4 152 29 1297c1f7-0230-4632-9914-2663344675

```

Task 4: Creating Customers Table with Constraints (DDL)

The Customers table was created using DDL statements, applying constraints such as PRIMARY KEY (CustomerID), CHECK constraint (Age > 18), NOT NULL (Age), and UNIQUE constraint (Name).

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Schemas: Local instance MySQL80 MySQL Model EER Diagram

Navigation: Day 7 worksheet SQL day 8 worksheet SQL excel-test 2 Amazon.sql project

Filter objects: Schemas Administrations Information

Table: order_details

Columns:

OrderID	varchar(50)
ProductID	varchar(50)
Quantity	int
UnitPrice	int
Discount	int

Action Output:

- Time Action
- 56 19:12:30 select * from amazon.Products LIMIT 0, 1000
- 57 19:12:58 select * from amazon.Products where Category='Fruits' LIMIT 0, 1000
- 58 19:15:07 AlterTable amazon.customers, CustomerID VARCHAR(50) (Primary Key)
- 59 19:21:32 alterTable amazon.customers add primary key(CustomerID)
- 60 19:22:08 select * from amazon.customers LIMIT 0, 1000
- 61 19:22:28 desc amazon.customers

Result Grid: Filter Rows Export Wrap Cell Content

Field	Type	Null	Key	Default	Extra
CustomerID	varchar(50)	NO	PRI	_____	
Name	text	YES	_____	_____	
Age	int	YES	_____	_____	
Gender	text	YES	_____	_____	
City	text	YES	_____	_____	

Object Info Session Query Completed

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Schemas: Local instance MySQL80 MySQL Model EER Diagram

Navigation: Day 7 worksheet SQL day 8 worksheet SQL excel-test 2 Amazon.sql project customers - Table

Filter objects: Schemas Administrations Information

Table: customers

Columns:

CustomerID	varchar(50) PK
Name	text
Age	int
Gender	text
City	text
State	text
Country	text
SignupDate	text
PrimeNumber	text

Action Output:

- Time Action Message Duration / Perf.
- 74 19:51:26 desc amazon.customers 3 rows(s) returned 0.018 sec / 0.000 sec
- 75 19:52:06 Apply changes to customers No changes detected
- 76 19:52:06 Apply changes to customers No changes detected
- 77 19:52:25 Apply changes to customers Changes applied
- 78 19:52:36 alterTable amazon.customers modify Age INT Not Null 3 rows(s) affected, 3 columns(s) updated, 0 warnings 0.002 sec
- 79 19:52:45 desc amazon.customers 3 rows(s) returned 0.008 sec / 0.000 sec

Result Grid: Filter Rows Export Wrap Cell Content

Field	Type	Null	Key	Default	Extra
CustomerID	varchar(50)	NO	PRI	_____	
Name	text	YES	_____	_____	
Age	int	NO	_____	_____	
Gender	text	YES	_____	_____	
City	text	YES	_____	_____	

Object Info Session Query Completed

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the tabs "MySQL Workbench", "Local instance MySQL 8.0", "MySQL Model", and "Edit Diagram" are visible. Below the tabs, there's a toolbar with various icons for database management. The main area is a query editor titled "customers - Table". The SQL code in the editor is:

```
17 • add primary key(CustomerID);
18
19 • desc amazon.customers;
20 • alter table amazon.customers
21 modify Age INT Not Null;
22
23 • select * from amazon.customers
24 where age<10;
```

Below the editor, there's a "Result Grid" tab and a "History" tab. The "History" tab shows the execution log:

Action	Time	Action	Message	Duration / Epoch
75	19-52:06	Apply changes to customers	No changes detected.	0.000 sec / 0.000 sec
76	19-52:08	Apply changes to customers	No changes detected.	0.000 sec / 0.000 sec
77	19-52:25	Apply changes to customers	Changes applied	0.000 sec / 0.000 sec
78	19-52:26	alter table amazon.customers modify Age INT Not Null	0 rows(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.002 sec
79	19-52:45	desc amazon.customers	3 rows(s) returned	0.000 sec / 0.000 sec
80	19-53:09	Select Name,age from amazon.customers where age<10 LIMIT 0,1000	9 rows(s) returned	0.000 sec / 0.000 sec

Task 5: Inserting Records (DML)

Three new records were inserted into the Products table using INSERT statements. Each record includes ProductID, ProductName, Category, PricePerUnit, and StockQuantity.

The screenshot shows the MySQL Workbench interface. In the top navigation bar, the tabs "MySQL Workbench", "Local instance MySQL 8.0", "MySQL Model", and "Edit Diagram" are visible. Below the tabs, there's a toolbar with various icons for database management. The main area is a query editor titled "customers - Table". The SQL code in the editor is:

```
17 • add primary key(CustomerID);
18
19 • desc amazon.customers;
20 • alter table amazon.customers
21 modify Age INT Not Null;
22
23 • select * from amazon.customers
24 where age<10;
25
26 • select * from amazon.products;
27
28 insert into amazon.products(ProductID,ProductName,Category,SubCategory,PricePerUnit,StockQuantity,SupplierID)
29 values(101, 'Keto Kicker', 'Bakery', 'Savory Bakery', 17, 100, 100);
30 (102, 'Organic Lattes', 'Bakery', 'Savory Bakery', 15, 100, 100),
31 (103, 'Organic Milkshakes', 'Drinks', 'Savory Drinks', 14, 100, 100);
```

Below the editor, there's a "Result Grid" tab and a "History" tab. The "History" tab shows the execution log:

Action	Time	Action	Message	Duration / Epoch
77	19-52:25	Apply changes to customers	Changes applied	0.002 sec
78	19-52:36	alter table amazon.customers modify Age INT Not Null	0 rows(s) affected Records: 0 Duplicates: 0 Warnings: 0	0.002 sec
79	19-52:45	desc amazon.customers	3 rows(s) returned	0.000 sec / 0.000 sec
80	19-53:09	Select Name,age from amazon.customers where age<10 LIMIT 0,1000	9 rows(s) returned	0.000 sec / 0.000 sec
81	19-56:26	select * from amazon.products LIMIT 0,1000	500 rows(s) returned	0.000 sec / 0.000 sec
82	19-56:27	view into amazon.products(ProductID,ProductName,Category,SubCategory,PricePerUnit,St...)	3 rows(s) affected Records: 3 Duplicates: 0 Warnings: 0	0.031 sec

Task 6: Updating Product Stock

An UPDATE statement was executed to modify the stock quantity of a specific product using the ProductID condition. The result shows the successful update of the product's inventory.

The screenshot shows the MySQL Workbench interface with the title bar "MySQL Workbench" and "Local instance MySQL 8.0". The main area displays a query editor with the following SQL code:

```

100 LIMIT 1000 rows
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
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
579
580
581
582
583
584
585
586
587
588
589
589
590
591
592
593
594
595
596
597
598
599
599
600
601
602
603
604
605
606
607
608
609
609
610
611
612
613
614
615
616
617
618
619
619
620
621
622
623
624
625
626
627
628
629
629
630
631
632
633
634
635
636
637
638
639
639
640
641
642
643
644
645
646
647
648
649
649
650
651
652
653
654
655
656
657
658
659
659
660
661
662
663
664
665
666
667
668
669
669
670
671
672
673
674
675
676
677
678
679
679
680
681
682
683
684
685
686
687
687
688
689
689
690
691
692
693
694
695
696
697
697
698
699
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
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
818
819
819
820
821
822
823
824
825
826
827
828
829
829
830
831
832
833
834
835
836
837
838
839
839
840
841
842
843
844
845
846
847
848
849
849
850
851
852
853
854
855
856
857
858
859
859
860
861
862
863
864
865
866
867
868
869
869
870
871
872
873
874
875
876
877
878
879
879
880
881
882
883
884
885
886
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
916
917
918
919
919
920
921
922
923
924
925
926
927
928
929
929
930
931
932
933
934
935
936
937
938
939
939
940
941
942
943
944
945
946
947
948
949
949
950
951
952
953
954
955
956
957
958
959
959
960
961
962
963
964
965
966
967
968
969
969
970
971
972
973
974
975
976
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
1017
1018
1019
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1089
1090
1091
1092
1093
1094
1095
1096
1097
1097
1098
1099
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1189
1190
1191
1192
1193
1194
1195
1196
1197
1197
1198
1199
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1209
1210
1211
1212
1213
1214
1215
1216
1217
1217
1218
1219
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1289
1290
1291
1292
1293
1294
1295
1296
1297
1297
1298
1299
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1309
1310
1311
1312
1313
1314
1315
1316
1317
1317
1318
1319
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1389
1390
1391
1392
1393
1394
1395
1396
1397
1397
1398
1399
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1409
1410
1411
1412
1413
1414
1415
1416
1417
1417
1418
1419
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1489
1490
1491
1492
1493
1494
1495
1496
1497
1497
1498
1499
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1509
1510
1511
1512
1513
1514
1515
1516
1517
1517
1518
1519
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1589
1590
1591
1592
1593
1594
1595
1596
1597
1597
1598
1599
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1609
1610
1611
1612
1613
1614
1615
1616
1617
1617
1618
1619
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1689
1690
1691
1692
1693
1694
1695
1696
1697
1697
1698
1699
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1709
1710
1711
1712
1713
1714
1715
1716
1717
1717
1718
1719
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1789
1790
1791
1792
1793
1794
1795
1796
1797
1797
1798
1799
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1809
1810
1811
1812
1813
1814
1815
1816
1817
1817
1818
1819
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1889
1890
1891
1892
1893
1894
1895
1896
1897
1897
1898
1899
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1909
1910
1911
1912
1913
1914
1915
1916
1917
1917
1918
1919
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2098
2099
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2198
2199
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2298
2299
2299
2300
2301
2302
2303
2304
23
```

MySQL Workbench

File Edit View Query Database Server Tools Reporting Help

Navigation: Day 7 worksheet SQL day 8 worksheet SQL exec test 2+ Decomposed project customers - Table

SCHEMAS

Tables

```

41 * set sql_safe_updates=0;
42 *
43 * deleted first row of suppliers table
44 * delete from amazon.suppliers
45 * where City='South Korea';
46 *
47 * Task 8 check constraints
48 *
49 * select * from amazon.reviews;
50 *
51 * alter table amazon.reviews
52 * add constraint Check_constraint CHECK Rating >= 1 AND Rating <= 5;

```

Table: order_details

Columns:

- OrderID: varchar(10)
- ProductID: varchar(10)
- Quantity: int
- UnitPrice: float
- Discount: float

Result Grid | Filter Rows | Auto | Export: | Hide Cell Context: | Refresh rows

ReviewID	ProductID	CustomerID	Rating	ReviewText
00004e5ab2c4-00d4-a5d1-2fe8265555ed	a10073e5-1c00-4153-9e26-48d59e9e9023	508c745f-1c27-4117-88d2-29383404e21	1	Life this hour system at reveal.
00168983-19de-4640-8053-8bf3af79e0f9	013389f5-2e8-41ff-83ff-7f946752958	4f01403f-1a75-463a-9347-0e98aef1769e	1	These recent densities.
01220-01-0573-0275-e4c4-408a-54a4c07	017268a1-4137-4237-9ec3-74480295675	ca7a376e-ae75-4079-a4ef-1ba2a1d273e	5	Layover professor test pack.
00965e75-8468-4524-a077-7176567068c	0e9c2b7f-1883-9e12-9039-0f992233a7e2	1fe4f590-20e1-9074-9226-0e96f5f03c39	2	Better another everyday enough open.
00815200-1524-4174-a17f-2705a0193201	4365a6f2-2040-48b7-4e28-21f95a3c	54637224-4e27-4e4d-933-446067024a65	5	Per firm some safe think.
00985930-2289-4f75-93b-302368d8201	40320711-9894-9999-8758-0e632612139	5c27f399-0369-4033-8538-61618866	1	Akt institution reach.
00e7a5-0003-4c39-1924-1a0559	e3aa512b-104-4c21-9524-2f4e452bad	8c3d25a-e27b-4046-8d44-45386983d51	1	Check number carry soon describe star.
00946059-76c3-41c1-95f9-7d98d707e0c	3fb6127b-2e8f-42b7-9e83-20e62918029	3199fcf-0342-46f1-9170-69993240411	3	book wif successful.
009c6c1-2417-4ebd-8ca4-8b144902323	0a173a63-8152-4259-aef2-2703a8fa6d24	c2a772b-e403-4c54-9e21-d757a8fa447	4	Source before church however.
01266e83-4459-4739-6008-1d507027699	0a9933-9192-4970-9403-0001ca123099	503f703-fc5-5291-652-492328166ef	3	Interesting north insert standard TV.

Drop Info Session

Query Completed

MySQL Workbench

File Edit View Query Database Server Tools Reporting Help

Navigation: Day 7 worksheet SQL day 8 worksheet SQL exec test 2+ Decomposed project customers - Table

SCHEMAS

Tables

```

99 * addDefault constraints
100 * REINC amazon.customers;
101 * select * from amazon.customers;
102 *
103 * alter table amazon.customers
104 * modify PrimeMember VARCHAR(3) DEFAULT 'NO';
105 *

```

Table: customers

Columns:

- CustomerID: int(10)
- Name: varchar(255)
- Age: int
- Gender: bool
- City: text
- State: text
- Country: text
- SignUpDate: date
- PrimeMember: varchar(30)

Result Grid | Filter Rows | Auto | Export: | Hide Cell Context: | Refresh rows

Field	Type	Null	Key	Default	Extra
City	text	YES			
State	text	YES			
Country	text	YES			
SignUpDate	text	YES			
PrimeMember	varchar(30)	YES		NO	

Result 13 v Output

Action Details

- 117. 21-49-05 altertable amazon.customers modify PrimeMember VARCHAR(3) DEFAULT 'NO'; 1000 row(s) affected Records: 1000 Duplicates: 0 Warnings: 0 Duration / Fetch: 0.234 sec
- 118. 21-49-06 altertable amazon.customers modify PrimeMember VARCHAR(3) DEFAULT 'NO'; 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 Duration / Fetch: 0.078 sec
- 119. 21-49-16 select * from amazon.customers LIMIT 0, 1000 1000 row(s) returned Duration / Fetch: 0.000 sec / 0.000 sec
- 120. 21-49-25 altertable amazon.customers modify PrimeMember VARCHAR(3) DEFAULT 'NO'; 0 row(s) affected Records: 0 Duplicates: 0 Warnings: 0 Duration / Fetch: 0.002 sec
- 121. 21-49-30 select * from amazon.customers LIMIT 0, 1000 1000 row(s) returned Duration / Fetch: 0.000 sec / 0.076 sec
- 122. 21-49-41 DESC amazon.customers 0 rows(s) returned Duration / Fetch: 0.000 sec / 0.000 sec

Drop Info Session

Query Completed

Task 9: Using WHERE, GROUP BY, HAVING, ORDER BY Clauses

Queries were written to:

- Retrieve orders placed after 2024-01-01 (WHERE).
- List products with an average rating above 4 (HAVING).
- Rank products by total sales (GROUP BY and ORDER BY).

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigation: schemas SQL project_warehouse SQL AMAZON PROJECT Import sql project products - Table subcategories - Table categories - Table products - Table

Limit to 1000 rows

Table: products

Columns:

- ProductID: int
- ProductName: text
- Category: text
- Subcategory: text
- SupplierID: int
- Unit: text
- UnitPrice: decimal(10,2)
- StockQuantity: int
- SupplierID: varchar(50)
- SubCategoryID: int

```

12   group by ProductName
13   order by OrderAmount DESC
14
15 • select pr.ProductID,
16   pr.ProductName, r.Rating, o.OrderDate, od.Quantity + od.UnitsPrice as TotalSales
17
18   from
19   amazon.products as pr
20   join amazon.order_details as od on pr.ProductID = od.ProductID
21   join amazon.orders as o on od.OrderID=o.OrderID
22   join amazon.reviews as r on pr.ProductID=r.ProductID
23   where o.OrderDate = '2023-01-01'
24   group by pr.ProductID, pr.ProductName, od.Quantity, od.UnitsPrice, r.Rating, o.OrderDate
25   having avg(r.Rating)>4
26   order by TotalSales DESC
27
28

```

Result 34 rows

Result Row	ProductID	ProductName	Rating	OrderDate	TotalSales
1	1753026-d9a7-4cc0-aed0-7930d8f2058	Able Max	5	2023-01-01	9958
2	f721a367-7795-4c70-94c0-e797e7374629	Bright Snack	5	2023-01-01	9933
3	rac0292a-1348-4346-1030-52d45c366083	Cool Max	5	2023-01-01	9840
4	94224889-e971-4334-9233-4229e49994f1	Great Max	5	2023-01-01	9820
5	285001c4-4111-41a0-90c9-553e33fb888	Continue Fruit	5	2023-01-01	9740
6	32094629-94d3-4873-9207-4038801720ec	Smiley Max	5	2023-01-01	9712
7	ec880e5-3211-4aa4-a604-c128ab9a0d12	Movie Day	5	2023-01-01	9360
8	5809474-c0e5-46f5-8936-c854a7a00012	Year Day	5	2023-01-01	9350
9	e76bb0ac-3b53-4622-a037-8792705b0ab1	Thrill Day	5	2023-01-01	8540
10	162a299-9398-4a0a-9a0a-10a0a0a0a0a0	Rainbow Max	5	2023-01-01	6419

Read Only

Task 10: Identifying High-Value Customers

Calculations were performed to determine total spending per customer. Customers were ranked using ORDER BY and filtered to find those who spent more than ₹5,000.

MySQL Workbench

Local instance MySQL80

File Edit View Query Database Server Tools Scripting Help

Navigation: schemas SQL project_warehouse SQL AMAZON PROJECT Import sql project

Table: customers

Columns:

- CustomerID: int
- Name: text
- Age: int
- Gender: text
- City: text
- State: text
- Country: text
- SignupDate: text
- PrimaryNumber: varchar(13)

```

12   Order by
13   TotalSales DESC
14
15 • use amazon;
16
17 • select c.Name,c.CustomerID, odv.Quantity*od.UnitsPrice) as Total_spend,
18   rank() over (order by odv.Quantity*od.UnitsPrice) desc) as Total_rank
19   from amazon.customers as c
20   join amazon.order_details as od on c.CustomerID = o.CustomerID
21   join amazon.orders as o on o.CustomerID = c.CustomerID
22   join amazon.Order_Details as odv on o.OrderID=od.OrderID
23   group by c.Name,c.CustomerID
24   having sum(od.Quantity*od.UnitsPrice) > 5000
25   order by TOTAL_spend desc
26
27

```

Result 5 rows

Result Row	Name	CustomerID	Total_spend	Total_rank
1	Christopher Douglas	b3053a75-9ca9-40b9-a276-c41576a6a128	94434	1
2	Steve White	c2f1136e-1321-4273-9249-11b5f5d29f9	83859	2
3	Sean Allen	25d897ed-1117-4522-9e43-c2295431f112	81215	3
4	Allison Carter	666f42c-fc79-46f9-876e-a013bed3416	79910	4
5	Ryan Lee	47c246c-3446-402c-a733-037fe32d1ac	75344	5

Read Only

Task 11: Complex Joins and Aggregations

JOIN operations were used between Orders and Order_Details tables to compute total revenue per order. Additional queries identified top customers and suppliers with maximum stock.

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Project: sql_project_amazon

Task: 11

```

17 * select p.ProductName,c.Name ,o.OrderDate,s.StockQuantity,(od.Quantity * od.UnitPrice * ((1-c.Discount)/100)) as Total_revenue
18 from amazon.customers as c
19 join amazon.orders as o on o.CustomerID=c.CustomerID
20 join amazon.Order_Details as od on o.OrderID=od.OrderID
21 join amazon.products as p on p.ProductID = od.ProductID
22 group by p.ProductName,c.Name,o.OrderDate,s.StockQuantity
23 order by Total_revenue desc
24
25 *
26 select p.ProductName,s.SupplierName from
27 amazon.suppliers as s
28 right join amazon.products as p on s.SupplierID = p.SupplierID
29 order by p.ProductName,s.SupplierName
30
31 limit 10

```

Result Set 1:

ProductID	ProductName	SupplierName
1	Banana	Supplie A
2	Banana	Supplie B
3	Banana	Supplie C
4	Banana	Supplie D
5	Banana	Supplie E
6	Banana	Supplie F
7	Banana	Supplie G
8	Banana	Supplie H
9	Banana	Supplie I
10	Banana	Supplie J

Result Set 2:

CustomerID	Name	OrderDate	StockQuantity	Total_revenue
1	Sarah Copeland	2020-01-01	338	12313.5000
2	Del-Del	2020-01-01	246	9981.2800
3	Inside Snack	2020-01-01	384	9721.6000
4	Drip Drip	2020-01-01	169	9663.2500
5	Ella Dar	2020-01-01	282	9623.0000
6	Center Star	2020-01-01	490	9594.3000
7	Ale Max	2020-01-01	425	9552.0000
8	As Puff	2020-01-01	279	9186.0000
9	Data Profit	2020-01-01	276	9152.6200
10	Karen Fitzgerald	2020-01-01	164	8019.9900

Task 12: Normalization to 3NF

The Products table was normalized by separating category and subcategory details into a new table. Foreign keys were added to maintain referential integrity, ensuring a well-structured 3NF database.

MySQL Workbench

File Edit View Query Database Server Tools Scripting Help

Project: SQL_AMAZON_PROJECT

Task: 12

```

174 * select * from amazon.products
175
176 * select * from amazon.categories;
177 * select * from amazon.subCategories;
178
179 #Task 13 b Customers who doesn't placed an order
180
181 * select c.Name,c.CustomerID,o.OrderID
182 From amazon.customers as c
183 left join amazon.orders as o on c.CustomerID = o.CustomerID
184 where o.OrderID is null;
185
186 #task 14 world analysis
187
188 * select c.City,c.PhoneNumber

```

Result Set 1:

CategoryID	CategoryName
1	Fruit
2	Sheds
3	Vegetables
4	Baby
5	Dairy
6	Milk
7	Meat

The screenshot shows the MySQL Workbench interface. The title bar says "MySQL Workbench - Local instance MySQL8.0". The menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help. The left sidebar shows "Schemas" with "amazon" selected, containing tables like "Categories", "Products", "SubCategories", and "Suppliers". The main area has tabs for "SQL", "Amazon Project", "Amazon Ed Project", "products - Table", "SubCategories - Table", "Categories - Table", and "products - Table". The SQL tab contains a script:

```
188 * select * from amazon.products;
189 *
190 * select * from amazon.categories;
191 *
192 * select * from amazon.SubCategories;
193 *
194 * Task 13 is Customers who doesn't placed an order
195 *
196 *
197 * select c.Name, c.CustomerID, o.OrderID
198 * from amazon.customers as c
199 * left join amazon.orders as o on c.CustomerID = o.CustomerID
200 * where o.OrderID is null;
```

The "products" table editor is open, showing columns: ProductID (int), ProductName (varchar(255)), Category (text), SubCategory (text), PriceQuantity (int), StockQuantity (int), SupplierID (varchar(255)), and SubCategoryId (varchar(255)). The table has 13 rows. The "SubCategories" table editor is also visible at the top.

The screenshot shows the MySQL Workbench interface with the following details:

- MySQL Workbench** window title.
- Local instance MySQL80** connection selected.
- File Edit View Gears Database Server Tools Scripting Help** menu bar.
- Schemas** pane on the left showing **amazon** schema with objects like **indexes**, **foreign keys**, **triggers**, **views**, and **subcategories**.
- Table: products** selected in the Schemas pane.
- Columns** pane showing columns: **ProductID** (varchar(25)), **ProductName** (text), **Category** (text), **SubCategory** (text), **PricePerUnit** (int), **StockQuantity** (int), **SupplierID** (varchar(25)), and **SubCategoryID** (varchar(25)).
- SQL AMAZON PROJECT** tab active.
- Amazon sql project** dropdown.
- products - Table**, **subcategories - Table**, **categories - Table**, **products - Table** buttons.
- Limit to 1000 rows** dropdown.
- Query Editor** content:

```
188 * select * from amazon.products;
189
190 * select * from amazon.categories;
191 * select * from amazon.subCategories;
192
193 * desc amazon.subCategories;
194 * desc amazon.products;
195
196 * task 13 b Customers who doesn't placed an order
197
198 * select c.Name,c.CustomerID,o.OrderID
199   from amazon.customers as c
200   left join amazon.orders as o on c.CustomerID = o.CustomerID
201   where o.OrderID is null;
```
- Result Grid** pane below the editor showing the results of the last query.

Task 13: Subqueries and Nested Queries

Subqueries were used to identify:

- Top 3 products based on total sales revenue.
 - Customers who have not placed any orders.

This demonstrates advanced SQL techniques for data analysis.

The screenshot shows the MySQL Workbench interface. The top navigation bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help, and a search bar. On the left, there's a tree view for 'Schemas' and 'Tables'. The main area has tabs for 'sql_amazon_main' (selected), 'SQL AMAZON PROJECT', and 'Joinwise ed project'. A query editor window displays the following SQL code:

```
137 * select p.ProductName, sum(od.Quantity * od.UnitPrice * (1 - od.Discount/100.0)) as Total_Sales
138 From
139     amazon.order_details as od
140 join amazon.products as p on od.ProductID=p.ProductID
141 group by p.ProductName
142 order by Total_Sales desc
143
144 End;
```

Below the editor is a results grid titled 'ResultsGrid' with columns 'ProductName' and 'Total_Sales'. It contains three rows of data:

ProductName	Total_Sales
Fal Shaki	87195.000
Do Dari	75835.000
Baby Shaki	67893.990

At the bottom, there's an 'Action Output' section with a timeline showing execution steps and a 'Duration / Fetch' table.

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help, and a search bar. The left sidebar displays the 'Schemas' tree, which includes a 'Tables' section with 'orders' and its columns: OrderID, CustomerID, OrderDate, OrderQuantity, and DeliveryFee. Below this is an 'Administrators' section with a 'Dinners' entry. The main workspace contains a query editor with the following SQL code:

```
137 * select p.ProductName, sum(od.Quantity * od.UnitPrice) * ((od.Discount/100)) as Total_Sales
138 from
139    amazon.order_details as od
140    join amazon.products as p on od.ProductID=p.ProductID
141 group by p.ProductName
142 order by Total_Sales desc
143 limit 10;
144
145
146 #Task 13.0 Customers who doesn't placed an order
147
148 * select c.Name,c.CustomerID,c.OrderID
149 from session.customers as c
150 left join session.orders as o on c.CustomerID = o.CustomerID
151 where o.OrderID is null;
```

Below the query editor is a results grid titled 'Results Grid'. It has columns for Result Set, Name, File Name, Insert, and Map Cell Content. The data grid shows 15 rows of customer information, each with a small icon to the right.

Task 14: Real-World Analysis

Analytical queries were written to find:

- Cities with the highest number of Prime Members.
 - The top 3 most frequently ordered product categories.
These insights help understand customer trends and business performance.

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help, and a user icon. The left sidebar displays the 'Schemas' tree, which includes 'Local instance MySQL 8.0', 'Tables', 'Views', 'Procedures', 'Functions', 'Triggers', 'Indexes', 'Foreign keys', and 'Materialized views'. Below this is the 'Administration' section with 'Databases' and 'Information'. A 'Table: supplies' section is shown with columns: SupplierID (number(10)), SupplierName (text), ContactPerson (text), Phone (text), City (text), and State (text). The main workspace contains a query editor with the following code:

```
146 * select c.CustomerID, c.OrderID
147   from amazon.customers as c
148   left join amazon.orders as o on c.CustomerID = o.CustomerID
149   where o.OrderID is null;
150
151      -- need world analysis
152
153 *
154
155 * Select c.City,c.PrimeSeller
156   from amazon.customers as c
157   order by c.PrimeSeller desc
158
159 *
160 * select * from amazon.customers;
```

Below the code is a results grid titled 'Result grid' with columns 'City' and 'PrimeSeller'. The data is as follows:

City	PrimeSeller
Oremont	Yes
Megafit	No
Roberbury	Yes
South Georgieshield	Yes
Cashified	Yes
Bethsport	Yes
Gakerson	Yes
East Potida	Yes
Massyprod	Yes
Island Lizard	Yes

The screenshot shows the MySQL Workbench interface. The top menu bar includes File, Edit, View, Query, Database, Server, Tools, Scripting, Help, and a toolbar with various icons. The left sidebar displays the 'Schemas' tree, which includes the 'suppliers' schema. Below the tree, sections for Administration, Structure, and Information are visible. The main area contains two tabs: 'SQL (amazon PROJECT)' and 'Amazon sql project'. The 'SQL (amazon PROJECT)' tab contains two queries:

```
131 * select c.City,c.PrimeMember
  from amazon.customers as c
  order by c.PrimeMember desc;
132 *
133 * select * from amazon.customers;
134 *
135 * a top 3 most Frequently ordered category
136 *
137 * select distinct p.Category
  from amazon.products as p
  order by Category
  limit 3;
```

Below the queries is a results grid titled 'products 51 x'. The grid has columns for 'Category' (with a dropdown menu showing 'Electronics', 'Books', 'Movies', and 'Toys'), 'Edit', 'Delete', and 'Trunc'. The bottom status bar shows 'Direct Edit' and 'Session'.