

# SQL QUERIES

## 1. Display snum,sname,city and comm of all salespeople.

The screenshot shows a database interface with a query editor and a results grid. The query is:

```
49 • SELECT snum, sname, city, comm  
50 FROM Salespeople;
```

The results grid displays the following data:

snum	sname	city	comm
1001	Peel	London	0.12
1002	Serres	San Jose	0.13
1003	Axelrod	New York	0.10
1004	Motika	London	0.11
1007	Rafkin	Barcelona	0.15
*	NULL	NULL	NULL

The interface includes a sidebar with icons for Result Grid, Form Editor, and Field Types. Below the results grid is an output pane titled "Salespeople 5" showing the following log entries:

#	Time	Action	Message
14	11:47:44	CREATE TABLE Orders (	0 row(s) affected
15	11:48:40	INSERT INTO Orders (onum, amt, odate, cnum, snum) VALUES (3001, 18.69, '1994-10-03', 2008, 1007), (300...	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('persons'.
16	11:52:14	INSERT INTO Cust (cnum, cname, city, rating, snum) VALUES (2008, 'Unknown', 'Unknown', 100, 1007)	1 row(s) affected
17	11:52:35	INSERT INTO Orders (onum, amt, odate, cnum, snum) VALUES (3001, 18.69, '1994-10-03', 2008, 1007), (300...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
18	11:53:11	select * from Orders LIMIT 0, 1000	10 row(s) returned
19	11:55:07	SELECT snum, sname, city, comm FROM Salespeople LIMIT 0, 1000	5 row(s) returned

## 2. Display all snum without duplicates from all orders.

The screenshot shows a database interface with a query editor and a results grid. The query is:

```
56  
57 • SELECT DISTINCT snum  
58 FROM Orders;
```

The results grid displays the following data:

snum
1001
1002
1003
1004
1007

The interface includes a sidebar with icons for Result Grid, Form Editor, and Field Types. Below the results grid is an output pane titled "Orders 6" showing the following log entries:

#	Time	Action	Message
15	11:48:40	INSERT INTO Orders (onum, amt, odate, cnum, snum) VALUES (3001, 18.69, '1994-10-03', 2008, 1007), (300...	Error Code: 1452. Cannot add or update a child row: a foreign key constraint fails ('persons'.
16	11:52:14	INSERT INTO Cust (cnum, cname, city, rating, snum) VALUES (2008, 'Unknown', 'Unknown', 100, 1007)	1 row(s) affected
17	11:52:35	INSERT INTO Orders (onum, amt, odate, cnum, snum) VALUES (3001, 18.69, '1994-10-03', 2008, 1007), (300...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
18	11:53:11	select * from Orders LIMIT 0, 1000	10 row(s) returned
19	11:55:07	SELECT snum, sname, city, comm FROM Salespeople LIMIT 0, 1000	5 row(s) returned
20	12:04:33	SELECT DISTINCT snum FROM Orders LIMIT 0, 1000	5 row(s) returned

### 3. Display names and commissions of all salespeople in london.

```

52
53
54 •   SELECT sname, comm
55     FROM Salespeople
56    WHERE city = 'London';
57

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


| sname  | comm |
|--------|------|
| Peel   | 0.12 |
| Motika | 0.11 |


```

Salespeople 7 x

Output

Action Output

#	Time	Action	Message
16	11:52:14	INSERT INTO Cust (cnum, cname, city, rating, snum) VALUES (2008, 'Unknown', 'Unknown', 100, 1007)	1 row(s) affected
17	11:52:35	INSERT INTO Orders (onum, amt, odate, cnum, snum) VALUES (3001, 18.69, '1994-10-03', 2008, 1007), (300...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
18	11:53:11	select * from Orders LIMIT 0, 1000	10 row(s) returned
19	11:55:07	SELECT snum, sname, city, comm FROM Salespeople LIMIT 0, 1000	5 row(s) returned
20	12:04:33	SELECT DISTINCT snum FROM Orders LIMIT 0, 1000	5 row(s) returned
21	12:09:26	SELECT sname, comm FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	2 row(s) returned

### 4. All customers with rating of 100.

```

58
59 •   SELECT * FROM Cust WHERE rating = 100;
60

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |


| cnum | cname   | city    | rating | snum |
|------|---------|---------|--------|------|
| 2001 | Hoffman | London  | 100    | 1001 |
| 2004 | Grass   | Brelin  | 100    | 1002 |
| 2007 | Pereira | Rome    | 100    | 1004 |
| 2008 | Unknown | Unknown | 100    | 1007 |
| *    | NULL    | NULL    | NULL   | NULL |


```

Cust 8 x

Output

Action Output

#	Time	Action	Message
17	11:52:35	INSERT INTO Orders (onum, amt, odate, cnum, snum) VALUES (3001, 18.69, '1994-10-03', 2008, 1007), (300...	10 row(s) affected Records: 10 Duplicates: 0 Warnings: 0
18	11:53:11	select * from Orders LIMIT 0, 1000	10 row(s) returned
19	11:55:07	SELECT snum, sname, city, comm FROM Salespeople LIMIT 0, 1000	5 row(s) returned
20	12:04:33	SELECT DISTINCT snum FROM Orders LIMIT 0, 1000	5 row(s) returned
21	12:09:26	SELECT sname, comm FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	2 row(s) returned
22	12:12:30	SELECT * FROM Cust WHERE rating = 100 LIMIT 0, 1000	4 row(s) returned

## 5. Produce orderno, amount and date form all rows in the order table.

59 • `SELECT onum, amt, odate FROM Orders;`

onum	amt	odate
3001	18.69	1994-10-03
3002	1900.10	1994-10-03
3003	767.19	1994-10-03
3005	5160.45	1994-10-03
3006	1098.16	1994-10-04
3007	75.75	1994-10-05
3008	4723.00	1994-10-05
3009	1713.23	1994-10-04
3010	1309.95	1994-10-06
3011	9891.88	1994-10-06
*	NULL	NULL

Orders 9 ×

Output

Action Output

#	Time	Action	Message
18	11:53:11	select * from Orders LIMIT 0, 1000	10 row(s) returned
19	11:55:07	SELECT snum, sname, city, comm FROM Salespeople LIMIT 0, 1000	5 row(s) returned
20	12:04:33	SELECT DISTINCT snum FROM Orders LIMIT 0, 1000	5 row(s) returned
21	12:09:26	SELECT sname, comm FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	2 row(s) returned
22	12:12:30	SELECT * FROM Cust WHERE rating = 100 LIMIT 0, 1000	4 row(s) returned
23	12:14:37	SELECT onum, amt, odate FROM Orders LIMIT 0, 1000	10 row(s) returned

## 6. All customers in San Jose, who have rating more than 200.

58 • `SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200;`

onum	cname	city	rating	snum
2003	Liu	San Jose	300	1002
*	NULL	NULL	NULL	NULL

Cust 10 ×

Output

Action Output

#	Time	Action	Message
20	12:04:33	SELECT DISTINCT snum FROM Orders LIMIT 0, 1000	5 row(s) returned
21	12:09:26	SELECT sname, comm FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	2 row(s) returned
22	12:12:30	SELECT * FROM Cust WHERE rating = 100 LIMIT 0, 1000	4 row(s) returned
23	12:14:37	SELECT onum, amt, odate FROM Orders LIMIT 0, 1000	10 row(s) returned
24	12:16:14	CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE...	Error Code: 1050. Table 'salespeople' already exists
25	12:16:27	SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200 LIMIT 0, 1000	1 row(s) returned

**7. All customers who were either located in San Jose or had a rating above 200.**

The screenshot shows the MySQL Workbench interface. At the top, a query window displays the following SQL code:

```
59
60 •  SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200;
```

Below the query window is the "Result Grid" pane, which shows the following data:

cnum	cname	city	rating	snum
2003	Liu	San Jose	300	1002
2006	Clemens	London	300	1007
NULL	NULL	NULL	NULL	NULL

On the right side of the interface, there is a vertical toolbar with icons for "Result Grid", "Form Editor", and "Field Types". Below the toolbar, there are "Apply" and "Revert" buttons.

At the bottom, the "Cust 11" tab is active, showing the following action history:

#	Time	Action	Message
21	12:09:26	SELECT sname, comm FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	2 row(s) returned
22	12:12:30	SELECT * FROM Cust WHERE rating = 100 LIMIT 0, 1000	4 row(s) returned
23	12:14:37	SELECT onum, amt, odate FROM Orders LIMIT 0, 1000	10 row(s) returned
24	12:16:14	CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE... Error Code: 1050. Table 'salespeople' already exists	Error Code: 1050. Table 'salespeople' already exists
25	12:16:27	SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200 LIMIT 0, 1000	1 row(s) returned
26	12:17:58	SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200 LIMIT 0, 1000	2 row(s) returned

**8. All orders for more than \$1000.**

The screenshot shows the MySQL Workbench interface. At the top, a query window displays the following SQL code:

```
59
60 •  SELECT * FROM Orders WHERE amt > 1000;
```

Below the query window is the "Result Grid" pane, which shows the following data:

onum	amt	odate	cnum	snum
3002	1900.10	1994-10-03	2007	1004
3005	5160.45	1994-10-03	2003	1002
3006	1098.16	1994-10-04	2008	1007
3008	4723.00	1994-10-05	2006	1001
3009	1713.23	1994-10-04	2002	1003
3010	1309.95	1994-10-06	2004	1002
3011	9891.88	1994-10-06	2006	1001
NULL	NULL	NULL	NULL	NULL

On the right side of the interface, there is a vertical toolbar with icons for "Result Grid", "Form Editor", and "Field Types". Below the toolbar, there are "Apply" and "Revert" buttons.

At the bottom, the "Orders 12" tab is active, showing the following action history:

#	Time	Action	Message
22	12:12:30	SELECT * FROM Cust WHERE rating = 100 LIMIT 0, 1000	4 row(s) returned
23	12:14:37	SELECT onum, amt, odate FROM Orders LIMIT 0, 1000	10 row(s) returned
24	12:16:14	CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE... Error Code: 1050. Table 'salespeople' already exists	Error Code: 1050. Table 'salespeople' already exists
25	12:16:27	SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200 LIMIT 0, 1000	1 row(s) returned
26	12:17:58	SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200 LIMIT 0, 1000	2 row(s) returned
27	12:19:12	SELECT * FROM Orders WHERE amt > 1000 LIMIT 0, 1000	7 row(s) returned

## 9. Names and cities of all salespeople in london with commission above 0.10.

```

59
60 •  SELECT sname, city FROM Salespeople WHERE city = 'London' AND comm > 0.10;
~~

Result Grid | Filter Rows: _____ | Export: | Wrap Cell Content: 
sname      city
Peel       London
Motika    London

Salespeople 13 x
Output
Action Output
# Time Action
1 23 12:14:37 SELECT onum, amt, odate FROM Orders LIMIT 0, 1000
2 24 12:16:14 CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE... Error Code: 1050. Table 'salespeople' already exists
3 25 12:16:27 SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200 LIMIT 0, 1000
4 26 12:17:58 SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200 LIMIT 0, 1000
5 27 12:19:12 SELECT * FROM Orders WHERE amt > 1000 LIMIT 0, 1000
6 28 12:32:14 SELECT sname, city FROM Salespeople WHERE city = 'London' AND comm > 0.10 LIMIT 0, 1000

```

## 10. All customers excluding those with rating <= 100 unless they are located in Rome.

```

58
59
60 •  SELECT * FROM Cust WHERE (rating > 100) OR city = 'Rome';
~~

Result Grid | Filter Rows: _____ | Edit: | Export/Import: | Wrap Cell Content: 
cnum      cname   city    rating  snum
2002    Giovanne Rome    200    1003
2003     Liu      San Jose 300    1002
2006    Clemens  London  300    1007
2007    Pereira  Rome    100    1004
*      NULL     NULL    NULL    NULL

Cust 14 x
Output
Action Output
# Time Action
1 24 12:16:14 CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE... Error Code: 1050. Table 'salespeople' already exists
2 25 12:16:27 SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200 LIMIT 0, 1000
3 26 12:17:58 SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200 LIMIT 0, 1000
4 27 12:19:12 SELECT * FROM Orders WHERE amt > 1000 LIMIT 0, 1000
5 28 12:32:14 SELECT sname, city FROM Salespeople WHERE city = 'London' AND comm > 0.10 LIMIT 0, 1000
6 29 12:33:59 SELECT * FROM Cust WHERE (rating > 100) OR city = 'Rome' LIMIT 0, 1000

```

## 11. All salespeople either in Barcelona or in london.

The screenshot shows the Oracle SQL Developer interface. The SQL tab contains the following query:

```
60 •  SELECT * FROM Salespeople WHERE city IN ('Barcelona', 'London');
```

The Result Grid shows the following data:

snum	sname	city	comm
1001	Peel	London	0.12
1004	Motika	London	0.11
1007	Rafkin	Barcelona	0.15

The Action Output pane shows the history of actions taken:

#	Time	Action	Message
25	12:16:27	SELECT * FROM Cust WHERE city = 'San Jose' AND rating > 200 LIMIT 0, 1000	1 row(s) returned
26	12:17:58	SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200 LIMIT 0, 1000	2 row(s) returned
27	12:19:12	SELECT * FROM Orders WHERE amt > 1000 LIMIT 0, 1000	7 row(s) returned
28	12:32:14	SELECT sname, city FROM Salespeople WHERE city = 'London' AND comm > 0.10 LIMIT 0, 1000	2 row(s) returned
29	12:33:59	SELECT * FROM Cust WHERE (rating > 100) OR city = 'Rome' LIMIT 0, 1000	4 row(s) returned
30	12:37:36	SELECT * FROM Salespeople WHERE city IN ('Barcelona', 'London') LIMIT 0, 1000	3 row(s) returned

## 12. All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded)

The screenshot shows the Oracle SQL Developer interface. The SQL tab contains the following query:

```
59  
60 •  SELECT * FROM Salespeople WHERE comm > 0.10 AND comm < 0.12;
```

The Result Grid shows the following data:

snum	sname	city	comm
1004	Motika	London	0.11

The Action Output pane shows the history of actions taken:

#	Time	Action	Message
26	12:17:58	SELECT * FROM Cust WHERE city = 'San Jose' OR rating > 200 LIMIT 0, 1000	2 row(s) returned
27	12:19:12	SELECT * FROM Orders WHERE amt > 1000 LIMIT 0, 1000	7 row(s) returned
28	12:32:14	SELECT sname, city FROM Salespeople WHERE city = 'London' AND comm > 0.10 LIMIT 0, 1000	2 row(s) returned
29	12:33:59	SELECT * FROM Cust WHERE (rating > 100) OR city = 'Rome' LIMIT 0, 1000	4 row(s) returned
30	12:37:36	SELECT * FROM Salespeople WHERE city IN ('Barcelona', 'London') LIMIT 0, 1000	3 row(s) returned
31	12:39:03	SELECT * FROM Salespeople WHERE comm > 0.10 AND comm < 0.12 LIMIT 0, 1000	1 row(s) returned

### 13. All customers with NULL values in city column.

60 • `SELECT * FROM Cust WHERE city IS NULL;`

cnum	cname	city	rating	snum
NULL	NULL	NULL	NULL	NULL

Cust 17 x Apply Revert

Output:

#	Time	Action	Message
27	12:19:12	SELECT * FROM Orders WHERE amt > 1000 LIMIT 0, 1000	7 row(s) returned
28	12:32:14	SELECT sname, city FROM Salespeople WHERE city = 'London' AND comm > 0.10 LIMIT 0, 1000	2 row(s) returned
29	12:33:59	SELECT * FROM Cust WHERE (rating > 100) OR city = 'Rome' LIMIT 0, 1000	4 row(s) returned
30	12:37:36	SELECT * FROM Salespeople WHERE city IN ('Barcelona', 'London') LIMIT 0, 1000	3 row(s) returned
31	12:39:03	SELECT * FROM Salespeople WHERE comm > 0.10 AND comm < 0.12 LIMIT 0, 1000	1 row(s) returned
32	12:40:43	SELECT * FROM Cust WHERE city IS NULL LIMIT 0, 1000	0 row(s) returned

### 14. All orders taken on Oct 3<sup>rd</sup> and Oct 4<sup>th</sup> 1994.

60 • `SELECT * FROM Orders WHERE odate IN ('1994-10-03', '1994-10-04');`

onum	amt	odate	cnum	snum
3001	18.69	1994-10-03	2008	1007
3002	1900.10	1994-10-03	2007	1004
3003	767.19	1994-10-03	2001	1001
3005	5160.45	1994-10-03	2003	1002
3006	1098.16	1994-10-04	2008	1007
3009	1713.23	1994-10-04	2002	1003
NULL	NULL	NULL	NULL	NULL

Orders 18 x Apply Revert

Output:

#	Time	Action	Message
29	12:33:59	SELECT * FROM Cust WHERE (rating > 100) OR city = 'Rome' LIMIT 0, 1000	4 row(s) returned
30	12:37:36	SELECT * FROM Salespeople WHERE city IN ('Barcelona', 'London') LIMIT 0, 1000	3 row(s) returned
31	12:39:03	SELECT * FROM Salespeople WHERE comm > 0.10 AND comm < 0.12 LIMIT 0, 1000	1 row(s) returned
32	12:40:43	SELECT * FROM Cust WHERE city IS NULL LIMIT 0, 1000	0 row(s) returned
33	12:41:45	CREATE TABLE Salespeople (onum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE...	Error Code: 1050. Table 'salespeople' already exists
34	12:41:51	SELECT * FROM Orders WHERE odate IN ('1994-10-03', '1994-10-04') LIMIT 0, 1000	6 row(s) returned

## 15. All customers serviced by Peel or Motika.

60 • `SELECT * FROM Cust WHERE snum IN (SELECT snum FROM Salespeople WHERE sname IN ('Peel', 'Motika'));`

cnum	cname	city	rating	snum
2001	Hoffman	London	100	1001
2007	Pereira	Rome	100	1004
*	NULL	NULL	NULL	NULL

**Cust 19** x

Output

Action Output

#	Time	Action	Message
30	12:37:36	SELECT * FROM Salespeople WHERE city IN ('Barcelona', 'London') LIMIT 0, 1000	3 row(s) returned
31	12:39:03	SELECT * FROM Salespeople WHERE comm > 0.10 AND comm < 0.12 LIMIT 0, 1000	1 row(s) returned
32	12:40:43	SELECT * FROM Cust WHERE city IS NULL LIMIT 0, 1000	0 row(s) returned
33	12:41:45	CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE...	Error Code: 1050. Table 'salespeople' already exists
34	12:41:51	SELECT * FROM Orders WHERE odate IN ('1994-10-03', '1994-10-04') LIMIT 0, 1000	6 row(s) returned
35	12:42:54	SELECT * FROM Cust WHERE snum IN (SELECT snum FROM Salespeople WHERE sname IN ('Peel', 'Motik...'))	2 row(s) returned

## 16. All customers whose names begin with a letter from A to B.

59

60 • `SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%';`

cnum	cname	city	rating	snum
*	NULL	NULL	NULL	NULL

**Cust 21** x

Output

Action Output

#	Time	Action	Message
32	12:40:43	SELECT * FROM Cust WHERE city IS NULL LIMIT 0, 1000	0 row(s) returned
33	12:41:45	CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE...	Error Code: 1050. Table 'salespeople' already exists
34	12:41:51	SELECT * FROM Orders WHERE odate IN ('1994-10-03', '1994-10-04') LIMIT 0, 1000	6 row(s) returned
35	12:42:54	SELECT * FROM Cust WHERE snum IN (SELECT snum FROM Salespeople WHERE sname IN ('Peel', 'Motik...'))	2 row(s) returned
36	12:43:58	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
37	12:44:06	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned

## 17. All orders except those with 0 or NULL value in amt field.

The screenshot shows a database query interface with the following details:

Query:

```
59
60 • SELECT * FROM Orders WHERE amt IS NOT NULL AND amt <> 0;
```

Result Grid:

onum	amt	odate	cnum	snum
3001	18.69	1994-10-03	2008	1007
3002	1900.10	1994-10-03	2007	1004
3003	767.19	1994-10-03	2001	1001
3005	5160.45	1994-10-03	2003	1002
3006	1098.16	1994-10-04	2008	1007
3007	75.75	1994-10-05	2004	1002
3008	4723.00	1994-10-05	2006	1001
3009	1713.23	1994-10-04	2002	1003
3010	1309.95	1994-10-06	2004	1002
3011	9891.88	1994-10-06	2006	1001
NULL	NULL	NULL	NULL	NULL

Output:

#	Time	Action	Message
33	12:41:45	CREATE TABLE Salespeople (snum INT PRIMARY KEY,sname VARCHAR(50),city VARCHAR(50),comm DE...	Error Code: 1050. Table 'salespeople' already exists
34	12:41:51	SELECT * FROM Orders WHERE odate IN ('1994-10-03', '1994-10-04') LIMIT 0, 1000	6 row(s) returned
35	12:42:54	SELECT * FROM Cust WHERE snum IN (SELECT snum FROM Salespeople WHERE sname IN ('Peel', 'Motik...'))	2 row(s) returned
36	12:43:58	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
37	12:44:06	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
38	12:45:44	SELECT * FROM Orders WHERE amt IS NOT NULL AND amt <> 0 LIMIT 0, 1000	10 row(s) returned

## 18. Count the number of salespeople currently listing orders in the order table.

The screenshot shows a database query interface with the following details:

Query:

```
58
59
60 • SELECT COUNT(DISTINCT snum) AS num_salespeople_with_orders FROM Orders;
```

Result Grid:

num_salespeople_with_orders
5

Output:

#	Time	Action	Message
34	12:41:51	SELECT * FROM Orders WHERE odate IN ('1994-10-03', '1994-10-04') LIMIT 0, 1000	6 row(s) returned
35	12:42:54	SELECT * FROM Cust WHERE snum IN (SELECT snum FROM Salespeople WHERE sname IN ('Peel', 'Motik...'))	2 row(s) returned
36	12:43:58	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
37	12:44:06	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
38	12:45:44	SELECT COUNT(DISTINCT snum) AS num_salespeople_with_orders FROM Orders LIMIT 0, 1000	1 row(s) returned
39	12:46:49	SELECT COUNT(DISTINCT snum) AS num_salespeople_with_orders FROM Orders LIMIT 0, 1000	1 row(s) returned

## 19. Largest order taken by each salesperson, datewise.

The screenshot shows a database query interface with the following details:

SQL Query:

```
59
60 • SELECT snum, odate, MAX(amt) AS max_order FROM Orders GROUP BY snum, odate;
```

Result Grid:

snum	odate	max_order
1007	1994-10-03	18.69
1004	1994-10-03	1900.10
1001	1994-10-03	767.19
1002	1994-10-03	5160.45
1007	1994-10-04	1098.16
1002	1994-10-05	75.75
1001	1994-10-05	4723.00
1003	1994-10-04	1713.23
1002	1994-10-06	1309.95
1001	1994-10-06	9891.88

Action Output:

#	Time	Action	Message
35	12:42:54	SELECT * FROM Cust WHERE snum IN (SELECT snum FROM Salespeople WHERE sname IN ('Peel', 'Motik...'))	2 row(s) returned
36	12:43:58	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
37	12:44:06	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
38	12:45:44	SELECT * FROM Orders WHERE amt IS NOT NULL AND amt < 0 LIMIT 0, 1000	10 row(s) returned
39	12:46:49	SELECT COUNT(DISTINCT snum) AS num_salespeople_with_orders FROM Orders LIMIT 0, 1000	1 row(s) returned
40	12:48:00	SELECT snum, odate, MAX(amt) AS max_order FROM Orders GROUP BY snum, odate LIMIT 0, 1000	10 row(s) returned

## 20. Largest order taken by each salesperson with order value more than \$3000.

The screenshot shows a database query interface with the following details:

SQL Query:

```
59
60 • SELECT snum, MAX(amt) AS max_order FROM Orders WHERE amt > 3000 GROUP BY snum;
```

Result Grid:

snum	max_order
1001	9891.88
1002	5160.45

Action Output:

#	Time	Action	Message
36	12:43:58	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
37	12:44:06	SELECT * FROM Cust WHERE cname LIKE 'A%' OR cname LIKE 'B%' LIMIT 0, 1000	0 row(s) returned
38	12:45:44	SELECT * FROM Orders WHERE amt IS NOT NULL AND amt < 0 LIMIT 0, 1000	10 row(s) returned
39	12:46:49	SELECT COUNT(DISTINCT snum) AS num_salespeople_with_orders FROM Orders LIMIT 0, 1000	1 row(s) returned
40	12:48:00	SELECT snum, odate, MAX(amt) AS max_order FROM Orders GROUP BY snum, odate LIMIT 0, 1000	10 row(s) returned
41	12:49:22	SELECT snum, MAX(amt) AS max_order FROM Orders WHERE amt > 3000 GROUP BY snum LIMIT 0, 1000	2 row(s) returned

## 21. Which day had the highest total amount ordered.

The screenshot shows a database query results window. The SQL query is:

```
59
60 • SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 1;
```

The result grid displays one row:

odate	total_amt
1994-10-06	11201.83

The sidebar on the right includes tabs for Result Grid, Form Editor, and Field Types. The status bar at the bottom right indicates "Read Only".

## 22. Count all orders for Oct 3<sup>rd</sup>.

The screenshot shows a database query results window. The SQL query is:

```
59
60 • SELECT COUNT(*) AS num_orders FROM Orders WHERE odate = '1994-10-03';
```

The result grid displays one row:

num_orders
4

The sidebar on the right includes tabs for Result Grid, Form Editor, and Field Types. The status bar at the bottom right indicates "Read Only".

### 23. Count the number of different non NULL city values in customers table.

```
58  
59  
60 •  SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL;  
--  
  
Result Grid | Filter Rows: _____ | Export: _____ | Wrap Cell Content:   
unique_cities  
5  
  
Result 28 x Read Only  
Output  
Action Output  
# Time Action Message  
39 12:46:49 SELECT COUNT(DISTINCT snum) AS num_salespeople_with_orders FROM Orders LIMIT 0, 1000 1 row(s) returned  
40 12:48:00 SELECT snum, odate, MAX(amt) AS max_order FROM Orders GROUP BY snum, odate LIMIT 0, 1000 10 row(s) returned  
41 12:49:22 SELECT snum, MAX(amt) AS max_order FROM Orders WHERE amt > 3000 GROUP BY snum LIMIT 0, 1000 2 row(s) returned  
42 12:50:43 SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 1 1 row(s) returned  
43 12:51:59 SELECT COUNT(*) AS num_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000 1 row(s) returned  
44 12:53:23 SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL LIMIT 0, 1000 1 row(s) returned
```

### 24. Select each customer's smallest order.

```
59  
60 •  SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum;  
--  
  
Result Grid | Filter Rows: _____ | Export: _____ | Wrap Cell Content:   
cnum smallest_order  
2001 767.19  
2002 1713.23  
2003 5160.45  
2004 75.75  
2006 4723.00  
2007 1900.10  
2008 18.69  
  
Result 29 x Read Only  
Output  
Action Output  
# Time Action Message  
40 12:48:00 SELECT snum, odate, MAX(amt) AS max_order FROM Orders GROUP BY snum, odate LIMIT 0, 1000 10 row(s) returned  
41 12:49:22 SELECT snum, MAX(amt) AS max_order FROM Orders WHERE amt > 3000 GROUP BY snum LIMIT 0, 1000 2 row(s) returned  
42 12:50:43 SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 1 1 row(s) returned  
43 12:51:59 SELECT COUNT(*) AS num_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000 1 row(s) returned  
44 12:53:23 SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL LIMIT 0, 1000 1 row(s) returned  
45 12:54:46 SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum LIMIT 0, 1000 7 row(s) returned
```

## 25. First customer in alphabetical order whose name begins with G.

59  
60 • `SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1;`

**Result Grid** | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Fetch rows: | **Result Grid**  

cnum	cname	city	rating	snum
2002	Giovanne	Rome	200	1003
NULL	NULL	NULL	NULL	NULL

  
**Cust 30** × **Output**: Action Output

#	Time	Action	Message
41	12:49:22	SELECT snum, MAX(amt) AS max_order FROM Orders WHERE amt > 3000 GROUP BY snum LIMIT 0, 1000	2 row(s) returned
42	12:50:43	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 1	1 row(s) returned
43	12:51:59	SELECT COUNT(*) AS num_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000	1 row(s) returned
44	12:53:23	SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL LIMIT 0, 1000	1 row(s) returned
45	12:54:46	SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum LIMIT 0, 1000	7 row(s) returned
46	13:23:07	SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1	1 row(s) returned

## 26. Get the output like “ For dd/mm/yy there are \_\_\_ orders.

59  
60 • `SELECT DATE_FORMAT(odate, 'For %d/%m/%y there are') AS message,COUNT(*) AS total_orders FROM Orders GROUP BY odate;`

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

message	total_orders
For 03/10/94 there are	4
For 04/10/94 there are	2
For 05/10/94 there are	2
For 06/10/94 there are	2

  
**Result 31** × **Output**: Action Output

#	Time	Action	Message
42	12:50:43	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 1	1 row(s) returned
43	12:51:59	SELECT COUNT(*) AS num_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000	1 row(s) returned
44	12:53:23	SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL LIMIT 0, 1000	1 row(s) returned
45	12:54:46	SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum LIMIT 0, 1000	7 row(s) returned
46	13:23:07	SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1	1 row(s) returned
47	13:24:28	SELECT DATE_FORMAT(odate, 'For %d/%m/%y there are') AS message,COUNT(*) AS total_orders FROM Or...	4 row(s) returned

**27. Assume that each salesperson has a 12% commission. Produce order no., salesperson no., and amount of salesperson's commission for that order.**

```

59
60 •   SELECT onum, snum, amt * 0.12 AS commission FROM Orders;
--
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid Form Editor Field Types

onum	snum	commission
3001	1007	2.2428
3002	1004	228.0120
3003	1001	92.0628
3005	1002	619.2540
3006	1007	131.7792
3007	1002	9.0900
3008	1001	566.7600
3009	1003	205.5876
3010	1002	157.1940
3011	1001	1187.0256

Result 32 x Read Only

Output:

Action Output

#	Time	Action	Message
43	12:51:59	SELECT COUNT(*) AS num_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000	1 row(s) returned
44	12:53:23	SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL LIMIT 0, 1000	1 row(s) returned
45	12:54:46	SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum LIMIT 0, 1000	7 row(s) returned
46	13:23:07	SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1	1 row(s) returned
47	13:24:28	SELECT DATE_FORMAT(odate, 'For %d/%m/%y there are') AS message,COUNT(*) AS total_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000	4 row(s) returned
48	13:25:50	SELECT onum, snum, amt * 0.12 AS commission FROM Orders LIMIT 0, 1000	10 row(s) returned

**28. Find highest rating in each city. Put the output in this form. For the city (city), the highest rating is : (rating).**

```

59
60 •   SELECT city, MAX(rating) AS highest_rating,CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) AS message
61     FROM Cust
62   GROUP BY city;
--
```

Result Grid | Filter Rows: Export: Wrap Cell Content: Result Grid Form Editor Field Types

city	highest_rating	message
London	300	For the city London, the highest rating is: 300
Rome	200	For the city Rome, the highest rating is: 200
San Jose	300	For the city San Jose, the highest rating is: 300
Brelin	100	For the city Brelin, the highest rating is: 100
Unknown	100	For the city Unknown, the highest rating is: 100

Result 33 x Read Only

Output:

Action Output

#	Time	Action	Message
44	12:53:23	SELECT COUNT(DISTINCT city) AS unique_cities FROM Cust WHERE city IS NOT NULL LIMIT 0, 1000	1 row(s) returned
45	12:54:46	SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum LIMIT 0, 1000	7 row(s) returned
46	13:23:07	SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1	1 row(s) returned
47	13:24:28	SELECT DATE_FORMAT(odate, 'For %d/%m/%y there are') AS message,COUNT(*) AS total_orders FROM Orders WHERE odate = '1994-10-03' LIMIT 0, 1000	4 row(s) returned
48	13:25:50	SELECT onum, snum, amt * 0.12 AS commission FROM Orders LIMIT 0, 1000	10 row(s) returned
49	13:27:09	SELECT city, MAX(rating) AS highest_rating,CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) ...	5 row(s) returned

**29. Display the totals of orders for each day and place the results in descending order.**

```

59
60 •   SELECT odate, SUM(amt) AS total_amt FROM Orders
61     GROUP BY odate
62     ORDER BY total_amt DESC;
~~

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


| odate      | total_amt |
|------------|-----------|
| 1994-10-06 | 11201.83  |
| 1994-10-03 | 7846.43   |
| 1994-10-05 | 4798.75   |
| 1994-10-04 | 2811.39   |



Result 34 x
Output:
Action Output
# Time Action Message
45 12:54:46 SELECT cnum, MIN(amt) AS smallest_order FROM Orders GROUP BY cnum LIMIT 0, 1000 7 row(s) returned
46 13:23:07 SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1 1 row(s) returned
47 13:24:28 SELECT DATE_FORMAT(odate, For %d/%m/%y there are') AS message,COUNT(*) AS total_orders FROM Or... 4 row(s) returned
48 13:25:50 SELECT onum, snum, amt * 0.12 AS commission FROM Orders LIMIT 0, 1000 10 row(s) returned
49 13:27:09 SELECT city, MAX(rating) AS highest_rating,CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) ... 5 row(s) returned
50 13:28:36 SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 0... 4 row(s) returned

Read Only

```

**30. All combinations of salespeople and customers who shared a city. (ie same city).**

```

59
60 •   SELECT S.sname, C.cname, S.city
61     FROM Salespeople S
62     JOIN Cust C ON S.city = C.city;
~~

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


| sname  | cname   | city     |
|--------|---------|----------|
| Motika | Hoffman | London   |
| Peel   | Hoffman | London   |
| Serres | Liu     | San Jose |
| Motika | Clemens | London   |
| Peel   | Clemens | London   |



Result 35 x
Output:
Action Output
# Time Action Message
46 13:23:07 SELECT * FROM Cust WHERE cname LIKE 'G%' ORDER BY cname ASC LIMIT 1 1 row(s) returned
47 13:24:28 SELECT DATE_FORMAT(odate, For %d/%m/%y there are') AS message,COUNT(*) AS total_orders FROM Or... 4 row(s) returned
48 13:25:50 SELECT onum, snum, amt * 0.12 AS commission FROM Orders LIMIT 0, 1000 10 row(s) returned
49 13:27:09 SELECT city, MAX(rating) AS highest_rating,CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) ... 5 row(s) returned
50 13:28:36 SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 0... 4 row(s) returned
51 13:29:45 SELECT S.sname, C.cname, S.city FROM Salespeople S JOIN Cust C ON S.city = C.city LIMIT 0, 1000 5 row(s) returned

Read Only

```

### 31. Name of all customers matched with the salespeople serving them.

```
59
60 •   SELECT C.cname, S.sname
61   FROM Cust C
62   JOIN Salespeople S ON C.snum = S.snum
```

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

cname	sname
Hoffman	Peel
Liu	Serres
Grass	Serres
Giovanne	Axelrod
Pereira	Motika
Clemens	Rafkin
Unknown	Rafkin

Result 36 x Read Only

Action Output

#	Time	Action	Message
47	13:24:28	SELECT DATE_FORMAT(odate, 'For %d %m/%y there are') AS message,COUNT() AS total_orders FROM Orders	4 row(s) returned
48	13:25:50	SELECT onum, snum, amt * 0.12 AS commission FROM Orders LIMIT 0, 1000	10 row(s) returned
49	13:27:09	SELECT city, MAX(rating) AS highest_rating,CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) ...	5 row(s) returned
50	13:28:36	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 0...	4 row(s) returned
51	13:29:45	SELECT S.sname, C cname, S.city FROM Salespeople S JOIN Cust C ON S.city = C.city LIMIT 0, 1000	5 row(s) returned
52	13:32:23	SELECT C.cname, S.sname FROM Cust C JOIN Salespeople S ON C.snum = S.snum LIMIT 0, 1000	7 row(s) returned

### 32. List each order number followed by the name of the customer who made the order.

```
58
59
60 •   SELECT O.onum, C cname
61   FROM Orders O
62   JOIN Cust C ON O.cnum = C.cnum
```

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

onum	cname
3003	Hoffman
3009	Giovanne
3005	Liu
3007	Grass
3010	Grass
3008	Clemens
3011	Clemens
3002	Pereira
3001	Unknown
3006	Unknown

Result 37 x Read Only

Action Output

#	Time	Action	Message
48	13:25:50	SELECT onum, snum, amt * 0.12 AS commission FROM Orders LIMIT 0, 1000	10 row(s) returned
49	13:27:09	SELECT city, MAX(rating) AS highest_rating,CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) ...	5 row(s) returned
50	13:28:36	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 0...	4 row(s) returned
51	13:29:45	SELECT S.sname, C cname, S.city FROM Salespeople S JOIN Cust C ON S.city = C.city LIMIT 0, 1000	5 row(s) returned
52	13:32:23	SELECT C.cname, S.sname FROM Cust C JOIN Salespeople S ON C.snum = S.snum LIMIT 0, 1000	7 row(s) returned
53	13:33:25	SELECT O.onum, C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum LIMIT 0, 1000	10 row(s) returned

### 33. Names of salesperson and customer for each order after the order number.

```
59
60 •   SELECT O.onum, S.sname, C cname
61   FROM Orders O
62   JOIN Salespeople S ON O.snum = S.snum
63   JOIN Cust C ON O.cnum = C.cnum
```

Result Grid | Filter Rows: Export: Wrap Cell Content:  Result Grid | Form Editor | Field Types

onum	sname	cname
3003	Peel	Hoffman
3008	Peel	Clemens
3011	Peel	Clemens
3005	Serres	Liu
3007	Serres	Grass
3010	Serres	Grass
3009	Axelrod	Giovanne
3002	Motika	Pereira
3001	Rafkin	Unknown
3006	Rafkin	Unknown

Result 38 × Read Only

Action Output

#	Time	Action	Message
49	13:27:09	SELECT city, MAX(rating) AS highest_rating, CONCAT('For the city ', city, ', the highest rating is: ', MAX(rating)) ...	5 row(s) returned
50	13:28:36	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 0...	4 row(s) returned
51	13:29:45	SELECT S.sname, C cname, S.city FROM Salespeople S JOIN Cust C ON S.city = C.city LIMIT 0, 1000	5 row(s) returned
52	13:32:23	SELECT C cname, S.sname FROM Cust C JOIN Salespeople S ON C.snum = S.snum LIMIT 0, 1000	7 row(s) returned
53	13:33:25	SELECT O.onum, C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum LIMIT 0, 1000	10 row(s) returned
54	13:34:40	SELECT O.onum, S.sname, C cname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ...	10 row(s) returned

### 34. Produce all customer serviced by salespeople with a commission above 12%.

```
59
60 •   SELECT C cname
61   FROM Cust C
62   JOIN Salespeople S ON C.snum = S.snum
63   WHERE S.comm > 0.12;
```

Result Grid | Filter Rows: Export: Wrap Cell Content:  Result Grid | Form Editor | Field Types

cname
Liu
Grass
Clemens
Unknown

Result 39 × Read Only

Action Output

#	Time	Action	Message
50	13:28:36	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate ORDER BY total_amt DESC LIMIT 0...	4 row(s) returned
51	13:29:45	SELECT S.sname, C cname, S.city FROM Salespeople S JOIN Cust C ON S.city = C.city LIMIT 0, 1000	5 row(s) returned
52	13:32:23	SELECT C cname, S.sname FROM Cust C JOIN Salespeople S ON C.snum = S.snum LIMIT 0, 1000	7 row(s) returned
53	13:33:25	SELECT O.onum, C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum LIMIT 0, 1000	10 row(s) returned
54	13:34:40	SELECT O.onum, S.sname, C cname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ...	10 row(s) returned
55	13:36:10	SELECT C cname FROM Cust C JOIN Salespeople S ON C.snum = S.snum WHERE S.comm > 0.12 LIMIT 0, ...	4 row(s) returned

**35. Calculate the amount of the salesperson's commission on each order with a rating above 100.**

```

52
53 •   SELECT O.onum, O.amt * S.comm AS commission
54     FROM Orders O
55     JOIN Salespeople S ON O.snum = S.snum
56     JOIN Cust C ON O.cnum = C.cnum
57     WHERE C.rating > 100;
58
59

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result 40 | Read Only

onum	commission
3009	171.3230
3005	670.8585
3008	566.7600
3011	1187.0256

Action Output

#	Time	Action	Message
51	13:29:45	SELECT S.sname, C cname, S.city FROM Salespeople S JOIN Cust C ON S.city = C.city LIMIT 0, 1000	5 row(s) returned
52	13:32:23	SELECT C cname, S.sname FROM Cust C JOIN Salespeople S ON C.snum = S.snum LIMIT 0, 1000	7 row(s) returned
53	13:33:25	SELECT O.onum, C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum LIMIT 0, 1000	10 row(s) returned
54	13:34:40	SELECT O.onum, S.sname, C cname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ...	10 row(s) returned
55	13:36:10	SELECT C cname FROM Cust C JOIN Salespeople S ON C.snum = S.snum WHERE S.comm > 0.12 LIMIT 0, ...	4 row(s) returned
56	13:41:06	SELECT O.onum, O.amt * S.comm AS commission FROM Orders O JOIN Salespeople S ON O.snum = S.snum ...	4 row(s) returned

**36. Find all pairs of customers having the same rating.**

```

52
53 •   SELECT A cname AS customer1, B cname AS customer2
54     FROM Cust A, Cust B
55     WHERE A.rating = B.rating AND A.cnum <> B.cnum;
56
57

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result 41 | Read Only

customer1	customer2
Unknown	Hoffman
Pereira	Hoffman
Grass	Hoffman
Clemens	Liu
Unknown	Grass
Pereira	Grass
Hoffman	Grass
Liu	Clemens
Unknown	Pereira
Grass	Pereira
Hoffman	Pereira
Pereira	Unknown
Grass	Unknown
Hoffman	Unknown

Action Output

#	Time	Action	Message
52	13:32:23	SELECT C cname, S.sname FROM Cust C JOIN Salespeople S ON C.snum = S.snum LIMIT 0, 1000	7 row(s) returned
53	13:33:25	SELECT O.onum, C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum LIMIT 0, 1000	10 row(s) returned
54	13:34:40	SELECT O.onum, S.sname, C cname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ...	10 row(s) returned
55	13:36:10	SELECT C cname FROM Cust C JOIN Salespeople S ON C.snum = S.snum WHERE S.comm > 0.12 LIMIT 0, ...	4 row(s) returned
56	13:41:06	SELECT O.onum, O.amt * S.comm AS commission FROM Orders O JOIN Salespeople S ON O.snum = S.snum ...	4 row(s) returned
57	13:42:58	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...	14 row(s) returned

**37. Find all pairs of customers having the same rating, each pair coming once only.**

```

58 •   SELECT A.cname AS customer1, B.cname AS customer2
59     FROM Cust A, Cust B
60    WHERE A.rating = B.rating AND A.cnum < B.cnum;
A1

```

customer1	customer2
Hoffman	Grass
Hoffman	Pereira
Hoffman	Unknown
Liu	Clemens
Grass	Pereira
Grass	Unknown
Pereira	Unknown

Result 42 ×

Output:

Action Output	#	Time	Action	Message
53	13:33:25	SELECT O.onum, C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum LIMIT 0, 1000		10 row(s) returned
54	13:34:40	SELECT O.onum, S.sname, C cname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ...		10 row(s) returned
55	13:36:10	SELECT C cname FROM Cust C JOIN Salespeople S ON C.snum = S.snum WHERE S.comm > 0.12 LIMIT 0, ...		4 row(s) returned
56	13:41:06	SELECT O.onum, O.amt * S.comm AS commission FROM Orders O JOIN Salespeople S ON O.snum = S.snum ...		4 row(s) returned
57	13:42:58	SELECT A.cname AS customer1, B.cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...		14 row(s) returned
58	13:45:01	SELECT A.cname AS customer1, B.cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...		7 row(s) returned

**38. Policy is to assign three salesperson to each customers. Display all such combinations.**

```

56
57
58 •   SELECT C cname, S1.sname AS salesperson1, S2.sname AS salesperson2, S3.sname AS salesperson3
59     FROM Cust C
60     JOIN Salespeople S1 ON C.snum = S1.snum
61     JOIN Salespeople S2 ON C.snum != S2.snum
62     JOIN Salespeople S3 ON C.snum != S3.snum;

```

cname	salesperson1	salesperson2	salesperson3
Liu	Serres	Rafkin	Peel
Grass	Serres	Rafkin	Peel
Giovanne	Axelrod	Rafkin	Peel
Pereira	Motika	Rafkin	Peel
Liu	Serres	Motika	Peel
Grass	Serres	Motika	Peel
Giovanne	Axelrod	Motika	Peel
Clemens	Rafkin	Motika	Peel
Unknown	Rafkin	Motika	Peel
Liu	Serres	Axelrod	Peel
Grass	Serres	Axelrod	Peel
Pereira	Motika	Axelrod	Peel
Clemens	Rafkin	Axelrod	Peel
Unknown	Rafkin	Axelrod	Peel
Giovanne	Axelrod	Serres	Peel
Pereira	Motika	Serres	Peel
Clemens	Rafkin	Serres	Peel
Unknown	Rafkin	Serres	Peel
Liu	Serres	Peel	Peel
Grass	Serres	Peel	Peel
Giovanne	Axelrod	Peel	Peel
Pereira	Motika	Peel	Peel
Clemens	Rafkin	Peel	Peel

Result 43 ×

Output:

### 39. Display all customers located in cities where salesman serres has customer.

```
56
57
58 •  SELECT C cname, C city
59   FROM Cust C
60   WHERE C city IN (SELECT city FROM Salespeople WHERE sname = 'Serres');
61
62
```

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

cname	city
Liu	San Jose

Result Grid | Form Editor | Read Only

Cust 44 x

Action Output

#	Time	Action	Message
55	13:36:10	SELECT C cname FROM Cust C JOIN Salespeople S ON C.snum = S.snum WHERE S.comm > 0.12 LIMIT 0, ...	4 row(s) returned
56	13:41:06	SELECT O.onum, O.amt * S.comm AS commission FROM Orders O JOIN Salespeople S ON O.snum = S.snum ...	4 row(s) returned
57	13:42:58	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...	14 row(s) returned
58	13:45:01	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...	7 row(s) returned
59	13:49:52	SELECT C cname, S1.sname AS salesperson1, S2.sname AS salesperson2, S3.sname AS salesperson3 FRO...	112 row(s) returned
60	13:52:00	SELECT C cname, C city FROM Cust C WHERE C city IN (SELECT city FROM Salespeople WHERE sname =...)	1 row(s) returned

### 40. Find all pairs of customers served by single salesperson.

```
56
57
58 •  SELECT A cname AS customer1, B cname AS customer2, S sname AS salesperson
59   FROM Cust A, Cust B, Salespeople S
60   WHERE A.snum = S.snum AND B.snum = S.snum AND A.cnum <> B.cnum;
61
```

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

customer1	customer2	salesperson
Liu	Grass	Serres
Grass	Liu	Serres
Clemens	Unknown	Rafkin
Unknown	Clemens	Rafkin

Result Grid | Form Editor | Read Only

Result 45 x

Action Output

#	Time	Action	Message
56	13:41:06	SELECT O.onum, O.amt * S.comm AS commission FROM Orders O JOIN Salespeople S ON O.snum = S.snum ...	4 row(s) returned
57	13:42:58	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...	14 row(s) returned
58	13:45:01	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A.rating = B.rating A...	7 row(s) returned
59	13:49:52	SELECT C cname, S1.sname AS salesperson1, S2.sname AS salesperson2, S3.sname AS salesperson3 FRO...	112 row(s) returned
60	13:52:00	SELECT C cname, C city FROM Cust C WHERE C city IN (SELECT city FROM Salespeople WHERE sname =...)	1 row(s) returned
61	13:53:25	SELECT A cname AS customer1, B cname AS customer2, S sname AS salesperson FROM Cust A, Cust B, Sales...	4 row(s) returned

**41. Produce all pairs of salespeople which are living in the same city. Exclude combinations of salespeople with themselves as well as duplicates with the order reversed.**

```

53
54
55 •   SELECT S1.sname AS salesperson1, S2.sname AS salesperson2
56     FROM Salespeople S1, Salespeople S2
57    WHERE S1.city = S2.city AND S1.snum < S2.snum;
58
59
60

```

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

salesperson1	salesperson2
Peel	Motika

Result 46 x Read Only

Output:

Action Output

#	Time	Action	Message
57	13:42:58	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A rating = B rating A...	14 row(s) returned
58	13:45:01	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A rating = B rating A...	7 row(s) returned
59	13:49:52	SELECT C cname, S1 sname AS salesperson1, S2 sname AS salesperson2, S3 sname AS salesperson3 FRO...	112 row(s) returned
60	13:52:00	SELECT C cname, C city FROM Cust C WHERE C city IN (SELECT city FROM Salespeople WHERE sname =...)	1 row(s) returned
61	13:53:25	SELECT A cname AS customer1, B cname AS customer2, S sname AS salesperson FROM Cust A, Cust B, Sal...	4 row(s) returned
62	13:54:43	SELECT S1 sname AS salesperson1, S2 sname AS salesperson2 FROM Salespeople S1, Salespeople S2 WH...	1 row(s) returned

**42. Produce all pairs of orders by given customer, names that customers and eliminates duplicates.**

```

54
55 •   SELECT O1.onum AS order1, O2.onum AS order2, C cname
56     FROM Orders O1, Orders O2, Cust C
57    WHERE O1.cnum = O2.cnum AND O1.cnum = C.cnum AND O1.onum <> O2.onum
58    GROUP BY O1.onum, O2.onum, C cname;
59
60

```

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

order1	order2	cname
3007	3010	Grass
3010	3007	Grass
3008	3011	Clemens
3011	3008	Clemens
3001	3006	Unknown
3006	3001	Unknown

Result 47 x Read Only

Output:

Action Output

#	Time	Action	Message
58	13:45:01	SELECT A cname AS customer1, B cname AS customer2 FROM Cust A, Cust B WHERE A rating = B rating A...	7 row(s) returned
59	13:49:52	SELECT C cname, S1 sname AS salesperson1, S2 sname AS salesperson2, S3 sname AS salesperson3 FRO...	112 row(s) returned
60	13:52:00	SELECT C cname, C city FROM Cust C WHERE C city IN (SELECT city FROM Salespeople WHERE sname =...)	1 row(s) returned
61	13:53:25	SELECT A cname AS customer1, B cname AS customer2, S sname AS salesperson FROM Cust A, Cust B, Sal...	4 row(s) returned
62	13:54:43	SELECT S1 sname AS salesperson1, S2 sname AS salesperson2 FROM Salespeople S1, Salespeople S2 WH...	1 row(s) returned
63	13:55:35	SELECT O1.onum AS order1, O2.onum AS order2, C cname FROM Orders O1, Orders O2, Cust C WHERE O1...	6 row(s) returned

### 43. Produce names and cities of all customers with the same rating as Hoffman.

```

53
54
55 •  SELECT C.cname, C.city
56   FROM Cust C
57 WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hoffman')
58
59
60

```

**Result Grid**

cname	city
Hoffman	London
Grass	Brelin
Pereira	Rome
Unknown	Unknown

**Cust 48**

**Output**

Action Output

#	Time	Action	Message
59	13:49:52	SELECT C cname, S1.sname AS salesperson1, S2.sname AS salesperson2, S3.sname AS salesperson3 FRO...	112 row(s) returned
60	13:52:00	SELECT C cname, C city FROM Cust C WHERE C.city IN (SELECT city FROM Salespeople WHERE sname =...	1 row(s) returned
61	13:53:25	SELECT A cname AS customer1, B cname AS customer2, S sname AS salesperson FROM Cust A, Cust B, Sal...	4 row(s) returned
62	13:54:43	SELECT S1 sname AS salesperson1, S2 sname AS salesperson2 FROM Salespeople S1, Salespeople S2 WH...	1 row(s) returned
63	13:55:35	SELECT O1.onum AS order1, O2.onum AS order2, C cname FROM Orders O1, Orders O2, Cust C WHERE O1...	6 row(s) returned
64	13:56:37	SELECT C cname, C city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hof...	4 row(s) returned

### 44. Extract all the orders of Motika.

```

55 •  SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika')
56
57

```

**Result Grid**

onum	amt	odate	cnum	snum
3002	1900.10	1994-10-03	2007	1004
*	NULL	NULL	NULL	NULL

**Orders 49**

**Output**

Action Output

#	Time	Action	Message
60	13:52:00	SELECT C cname, C city FROM Cust C WHERE C.city IN (SELECT city FROM Salespeople WHERE sname =...	1 row(s) returned
61	13:53:25	SELECT A cname AS customer1, B cname AS customer2, S sname AS salesperson FROM Cust A, Cust B, Sal...	4 row(s) returned
62	13:54:43	SELECT S1 sname AS salesperson1, S2 sname AS salesperson2 FROM Salespeople S1, Salespeople S2 WH...	1 row(s) returned
63	13:55:35	SELECT O1.onum AS order1, O2.onum AS order2, C cname FROM Orders O1, Orders O2, Cust C WHERE O1...	6 row(s) returned
64	13:56:37	SELECT C cname, C city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Motika') LIMIT...	4 row(s) returned
65	13:57:47	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMIT...	1 row(s) returned

#### 45. All orders credited to the same salesperson who services Hoffman.

```

55
56
57 • SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman');
58

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: 


| onum | amt     | odate      | cnum | snum |
|------|---------|------------|------|------|
| 3003 | 767.19  | 1994-10-03 | 2001 | 1001 |
| 3008 | 4723.00 | 1994-10-05 | 2006 | 1001 |
| 3011 | 9891.88 | 1994-10-06 | 2006 | 1001 |
| *    | NULL    | NULL       | NULL | NULL |

  

Orders 50 x
Output:
Action Output
# Time Action Message
61 13:53:25 SELECT A.cname AS customer1, B.cname AS customer2, S.sname AS salesperson FROM Cust A, Cust B, Salespeople S WHERE A.cname = 'Motika' AND B.cname = 'Hoffman' AND S.sname = 'Hoffman' 4 row(s) returned
62 13:54:43 SELECT S1.sname AS salesperson1, S2.sname AS salesperson2 FROM Salespeople S1, Salespeople S2 WHERE S1.sname = 'Motika' AND S2.sname = 'Hoffman' 1 row(s) returned
63 13:55:35 SELECT O1.onum AS order1, O2.onum AS order2, C cname FROM Orders O1, Orders O2, Cust C WHERE O1.cnum = O2.cnum AND C.cname = 'Hoffman' 6 row(s) returned
64 13:56:37 SELECT C cname, C.city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hoffman') 4 row(s) returned
65 13:57:47 SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMIT 0, 1... 1 row(s) returned
66 13:59:01 SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1... 3 row(s) returned

```

#### 46. All orders that are greater than the average for Oct 4.

```

55
56 • SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04');
57
58

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: 


| onum | amt     | odate      | cnum | snum |
|------|---------|------------|------|------|
| 3002 | 1900.10 | 1994-10-03 | 2007 | 1004 |
| 3005 | 5160.45 | 1994-10-03 | 2003 | 1002 |
| 3008 | 4723.00 | 1994-10-05 | 2006 | 1001 |
| 3009 | 1713.23 | 1994-10-04 | 2002 | 1003 |
| 3011 | 9891.88 | 1994-10-06 | 2006 | 1001 |
| *    | NULL    | NULL       | NULL | NULL |

  

Orders 51 x
Output:
Action Output
# Time Action Message
62 13:54:43 SELECT S1.sname AS salesperson1, S2.sname AS salesperson2 FROM Salespeople S1, Salespeople S2 WHERE S1.sname = 'Motika' AND S2.sname = 'Hoffman' 1 row(s) returned
63 13:55:35 SELECT O1.onum AS order1, O2.onum AS order2, C cname FROM Orders O1, Orders O2, Cust C WHERE O1.cnum = O2.cnum AND C.cname = 'Hoffman' 6 row(s) returned
64 13:56:37 SELECT C cname, C.city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hoffman') 4 row(s) returned
65 13:57:47 SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMIT 0, 1... 1 row(s) returned
66 13:59:01 SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1... 3 row(s) returned
67 14:00:22 SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') LIMIT 0, 1... 5 row(s) returned

```

#### 47. Find average commission of salespeople in london.

```

54
55
56 •   SELECT AVG(comm)
57     FROM Salespeople
58     WHERE city = 'London';
59

```

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

	AVG(comm)
▶	0.115000

Result 52 ×

Output:

Action Output

#	Time	Action	Message
63	13:55:35	SELECT O1.onum AS order1, O2.onum AS order2, C cname FROM Orders O1, Orders O2, Cust C WHERE O1... 6 row(s) returned	
64	13:56:37	SELECT C cname, C.city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hof... 4 row(s) returned	
65	13:57:47	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMI... 1 row(s) returned	
66	13:59:01	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1... 3 row(s) returned	
67	14:00:22	SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') LI... 5 row(s) returned	
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000 1 row(s) returned	

Result 53 ×

Output:

Action Output

#	Time	Action	Message
64	13:56:37	SELECT C cname, C.city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hof... 4 row(s) returned	
65	13:57:47	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMI... 1 row(s) returned	
66	13:59:01	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1... 3 row(s) returned	
67	14:00:22	SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') LI... 5 row(s) returned	
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000 1 row(s) returned	
69	14:02:27	SELECT * FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnum = C.cnum WH... 3 row(s) returned	

#### 48. Find all orders attributed to salespeople servicing customers in london.

```

55
56 •   SELECT * FROM Orders O
57     JOIN Salespeople S ON O.snum = S.snum
58     JOIN Cust C ON O.cnum = C.cnum
59     WHERE C.city = 'London';
60
61

```

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

onum	amt	odate	cnum	snum	snum	sname	city	comm	cnum	cname	city	rating	snum
3003	767.19	1994-10-03	2001	1001	1001	Peel	London	0.12	2001	Hoffman	London	100	1001
3008	4723.00	1994-10-05	2006	1001	1001	Peel	London	0.12	2006	Clemens	London	300	1007
3011	9891.88	1994-10-06	2006	1001	1001	Peel	London	0.12	2006	Clemens	London	300	1007

Result 53 ×

Output:

Action Output

#	Time	Action	Message
64	13:56:37	SELECT C cname, C.city FROM Cust C WHERE C.rating = (SELECT rating FROM Cust WHERE cname = 'Hof... 4 row(s) returned	
65	13:57:47	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMI... 1 row(s) returned	
66	13:59:01	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1... 3 row(s) returned	
67	14:00:22	SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') LI... 5 row(s) returned	
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000 1 row(s) returned	
69	14:02:27	SELECT * FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnum = C.cnum WH... 3 row(s) returned	

#### 49. Extract commissions of all salespeople servicing customers in London.

```

55
56 •   SELECT S.sname, S.comm
57     FROM Salespeople S
58     JOIN Cust C ON S.snum = C.snum
59     WHERE C.city = 'London'
60
61

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

sname	comm
Peel	0.12
Rafkin	0.15

Output

Action Output

#	Time	Action	Message
65	13:57:47	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Salespeople WHERE sname = 'Motika') LIMIT 0, 1...	1 row(s) returned
66	13:59:01	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1...	3 row(s) returned
67	14:00:22	SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') Li...	5 row(s) returned
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	1 row(s) returned
69	14:02:27	SELECT * FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnum = C.cnum WH...	3 row(s) returned
70	14:05:26	SELECT S.sname, S.comm FROM Salespeople S JOIN Cust C ON S.snum = C.snum WHERE C.city = 'London'...	2 row(s) returned

#### 50. Find all customers whose cnum is 1000 above the snum of serres.

```

55
56 •   SELECT * FROM Cust WHERE cnum > (SELECT snum + 1000 FROM Salespeople WHERE sname = 'Serres')
57
58

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Apply

cnum	cname	city	rating	snum
2003	Liu	San Jose	300	1002
2004	Grass	Brelin	100	1002
2006	Clemens	London	300	1007
2007	Pereira	Rome	100	1004
2008	Unknown	Unknown	100	1007
*	HULL	HULL	HULL	HULL

Cust 55 x

Output

Action Output

#	Time	Action	Message
66	13:59:01	SELECT * FROM Orders WHERE snum = (SELECT snum FROM Cust WHERE cname = 'Hoffman') LIMIT 0, 1...	3 row(s) returned
67	14:00:22	SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') Li...	5 row(s) returned
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	1 row(s) returned
69	14:02:27	SELECT * FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnum = C.cnum WH...	3 row(s) returned
70	14:05:26	SELECT S.sname, S.comm FROM Salespeople S JOIN Cust C ON S.snum = C.snum WHERE C.city = 'London'...	2 row(s) returned
71	14:07:33	SELECT * FROM Cust WHERE cnum > (SELECT snum + 1000 FROM Salespeople WHERE sname = 'Serres')...	5 row(s) returned

## 51. Count the customers with rating above San Jose's average.

The screenshot shows a database interface with a query editor and a results grid. The query is:

```
55  
56 • SELECT COUNT(*) FROM Cust  
57 WHERE rating > (SELECT AVG(rating) FROM Cust WHERE city = 'San Jose')  
58  
59
```

The results grid shows:

COUNT(*)
0

The output pane shows the following log:

#	Time	Action	Message
67	14:00:22	SELECT * FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE odate = '1994-10-04') LIMIT 0, 1000	5 row(s) returned
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	1 row(s) returned
69	14:02:27	SELECT * FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnum = C.cnum WHERE C.city = 'London' LIMIT 0, 1000	3 row(s) returned
70	14:05:26	SELECT S.sname, S.comm FROM Salespeople S JOIN Cust C ON S.snum = C.snum WHERE C.city = 'London' LIMIT 0, 1000	2 row(s) returned
71	14:07:33	SELECT * FROM Cust WHERE crum > (SELECT snum + 1000 FROM Salespeople WHERE sname = 'Serres') LIMIT 0, 1000	5 row(s) returned
72	14:09:09	SELECT COUNT(*) FROM Cust WHERE rating > (SELECT AVG(rating) FROM Cust WHERE city = 'San Jose') LIMIT 0, 1000	1 row(s) returned

## 52. Obtain all orders for the customer named Cisnerous. (Assume you don't know his customer no. (cnum)).

The screenshot shows a database interface with a query editor and a results grid. The query is:

```
55  
56 • SELECT * FROM Orders  
57 WHERE cnum = (SELECT cnum FROM Cust WHERE cname = 'Cisnerous')  
58  
59
```

The results grid shows:

onum	amt	odate	cnum	snum
NULL	NULL	HULL	NULL	NULL

The output pane shows the following log:

#	Time	Action	Message
68	14:01:31	SELECT AVG(comm) FROM Salespeople WHERE city = 'London' LIMIT 0, 1000	1 row(s) returned
69	14:02:27	SELECT * FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnum = C.cnum WHERE C.city = 'London' LIMIT 0, 1000	3 row(s) returned
70	14:05:26	SELECT S.sname, S.comm FROM Salespeople S JOIN Cust C ON S.snum = C.snum WHERE C.city = 'London' LIMIT 0, 1000	2 row(s) returned
71	14:07:33	SELECT * FROM Cust WHERE crum > (SELECT snum + 1000 FROM Salespeople WHERE sname = 'Serres') LIMIT 0, 1000	5 row(s) returned
72	14:09:09	SELECT COUNT(*) FROM Cust WHERE rating > (SELECT AVG(rating) FROM Cust WHERE city = 'San Jose') LIMIT 0, 1000	1 row(s) returned
73	14:10:55	SELECT * FROM Orders WHERE cnum = (SELECT cnum FROM Cust WHERE cname = 'Cisnerous') LIMIT 0, 1000	0 row(s) returned

**53. Produce the names and rating of all customers who have above average orders.**

```

55
56 •   SELECT C.cname, C.rating
57   FROM Cust C
58 WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders));
59

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:						
<table border="1"> <thead> <tr> <th>cname</th> <th>rating</th> </tr> </thead> <tbody> <tr> <td>Liu</td> <td>300</td> </tr> <tr> <td>Clemens</td> <td>300</td> </tr> </tbody> </table>	cname	rating	Liu	300	Clemens	300			
cname	rating								
Liu	300								
Clemens	300								

Result Grid
Form Editor
Read Only

**Cust 59 x**

**Output**

Action Output	#	Time	Action	Message
	70	14:05:26	SELECT S.sname, S.comm FROM Salespeople S JOIN Cust C ON S.snum = C.snum WHERE C.city = 'London'...	2 row(s) returned
	71	14:07:33	SELECT * FROM Cust WHERE cnum > (SELECT snum + 1000 FROM Salespeople WHERE sname = 'Serres')...	5 row(s) returned
	72	14:09:09	SELECT COUNT(*) FROM Cust WHERE rating > (SELECT AVG(rating) FROM Cust WHERE city = 'San Jose')...	1 row(s) returned
	73	14:10:55	SELECT * FROM Orders WHERE cnum = (SELECT cnum FROM Cust WHERE cname = 'Cinereous') LIMIT 0...	0 row(s) returned
	74	14:14:52	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (...	2 row(s) returned
	75	14:15:54	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (...	2 row(s) returned

**54. Find total amount in orders for each salesperson for whom this total is greater than the amount of the largest order in the table.**

```

33
34
35 •   SELECT S.sname, SUM(O.amt) AS total_amt
36   FROM Orders O
37   JOIN Salespeople S ON O.snum = S.snum
38   GROUP BY S.snum
39   HAVING SUM(O.amt) > (SELECT MAX(amt) FROM Orders);
40
41
42

```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:				
<table border="1"> <thead> <tr> <th>sname</th> <th>total_amt</th> </tr> </thead> <tbody> <tr> <td>Peel</td> <td>15382.07</td> </tr> </tbody> </table>	sname	total_amt	Peel	15382.07			
sname	total_amt						
Peel	15382.07						

Result Grid
Form Editor
Read Only

**Result 60 x**

**Output**

Action Output	#	Time	Action	Message
	71	14:07:33	SELECT * FROM Cust WHERE cnum > (SELECT snum + 1000 FROM Salespeople WHERE sname = 'Serres')...	5 row(s) returned
	72	14:09:09	SELECT COUNT(*) FROM Cust WHERE rating > (SELECT AVG(rating) FROM Cust WHERE city = 'San Jose')...	1 row(s) returned
	73	14:10:55	SELECT * FROM Orders WHERE cnum = (SELECT cnum FROM Cust WHERE cname = 'Cinereous') LIMIT 0...	0 row(s) returned
	74	14:14:52	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (...	2 row(s) returned
	75	14:15:54	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (...	2 row(s) returned
	76	14:17:55	SELECT S.sname, SUM(O.amt) AS total_amt FROM Orders O JOIN Salespeople S ON O.snum = S.snum GRO...	1 row(s) returned

## 55. Find all customers with order on 3<sup>rd</sup> Oct.

```
55
56 •   SELECT DISTINCT C.cname
57   FROM Orders O
58   JOIN Cust C ON O.cnum = C.cnum
59   WHERE O.odate = '1994-10-03';
60
61
62
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

name
Unknown
Pereira
Hoffman
Liu

Action Output

#	Time	Action	Message
72	14:09:09	SELECT COUNT(*) FROM Cust WHERE rating > (SELECT AVG(rating) FROM Cust WHERE city = 'San Jose')...	1 row(s) returned
73	14:10:55	SELECT * FROM Orders WHERE crum = (SELECT crum FROM Cust WHERE cname = 'Cisnerous') LIMIT 0...	0 row(s) returned
74	14:14:52	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT crum FROM Orders WHERE amt > (...	2 row(s) returned
75	14:15:54	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT crum FROM Orders WHERE amt > (...	2 row(s) returned
76	14:17:55	SELECT S.sname, SUM(O.amt) AS total_amt FROM Orders O JOIN Salespeople S ON O.snum = S.snum GRO...	1 row(s) returned
77	14:19:16	SELECT DISTINCT C.cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.odate = '1994-1...	4 row(s) returned

## 56. Find names and numbers of all salesperson who have more than one customer.

```
55
56 •   SELECT S.sname, S.snum
57   FROM Salespeople S
58   JOIN Cust C ON S.snum = C.snum
59   GROUP BY S.snum
60   HAVING COUNT(C.cnum) > 1;
61
62
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

sname	snum
Serres	1002
Rafkin	1007

Action Output

#	Time	Action	Message
73	14:10:55	SELECT * FROM Orders WHERE crum = (SELECT crum FROM Cust WHERE cname = 'Cisnerous') LIMIT 0...	0 row(s) returned
74	14:14:52	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT crum FROM Orders WHERE amt > (...	2 row(s) returned
75	14:15:54	SELECT C.cname, C.rating FROM Cust C WHERE C.cnum IN (SELECT crum FROM Orders WHERE amt > (...	2 row(s) returned
76	14:17:55	SELECT S.sname, SUM(O.amt) AS total_amt FROM Orders O JOIN Salespeople S ON O.snum = S.snum GRO...	1 row(s) returned
77	14:19:16	SELECT DISTINCT C.cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.odate = '1994-1...	4 row(s) returned
78	14:22:10	SELECT S.sname, S.snum FROM Salespeople S JOIN Cust C ON S.snum = C.snum GROUP BY S.snum HAVI...	2 row(s) returned

## 57. Check if the correct salesperson was credited with each sale.

```
55
56 • SELECT O.onum, S.sname
57   FROM Orders O
58   JOIN Salespeople S ON O.snum = S.snum
59   JOIN Cust C ON O.cnum = C.cnum
60   WHERE O.snum = C.snum;
61
62
```

Result Grid | Filter Rows: Export: Wrap Cell Content:  Result Grid | Form Editor | Read Only

onum	sname
3003	Peel
3005	Serres
3007	Serres
3010	Serres
3009	Axelrod
3002	Motika
3001	Rafkin
3006	Rafkin

Result 63 ×

Action Output

#	Time	Action	Message
74	14:14:52	SELECT C cname, C rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE cnum = C.cnum))	2 row(s) returned
75	14:15:54	SELECT C cname, C rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE cnum = C.cnum))	2 row(s) returned
76	14:17:55	SELECT S.sname, SUM(O.amt) AS total_amt FROM Orders O JOIN Salespeople S ON O.snum = S.snum GROUP BY S.snum	1 row(s) returned
77	14:19:16	SELECT DISTINCT C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.odate = '1994-1-1'	4 row(s) returned
78	14:22:10	SELECT S.sname, S.snum FROM Salespeople S JOIN Cust C ON S.snum = C.cnum GROUP BY S.snum HAVING COUNT(C.cnum) > 1	2 row(s) returned
79	14:24:06	SELECT O.onum, S.sname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnu...	8 row(s) returned
80	14:25:39	SELECT O.onum, O.amt FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.amt > (SELECT AVG(amt) FROM Orders WHERE cnum = C.cnum)	3 row(s) returned

## 58. Find all orders with above average amounts for their customers.

```
55
56 • SELECT O.onum, O.amt
57   FROM Orders O
58   JOIN Cust C ON O.cnum = C.cnum
59   WHERE O.amt > (SELECT AVG(amt) FROM Orders WHERE cnum = O.cnum);
60
61
62
```

Result Grid | Filter Rows: Export: Wrap Cell Content:  Result Grid | Form Editor | Read Only

onum	amt
3010	1309.95
3011	9891.88
3006	1098.16

Result 64 ×

Action Output

#	Time	Action	Message
75	14:15:54	SELECT C cname, C rating FROM Cust C WHERE C.cnum IN (SELECT cnum FROM Orders WHERE amt > (SELECT AVG(amt) FROM Orders WHERE cnum = C.cnum))	2 row(s) returned
76	14:17:55	SELECT S.sname, SUM(O.amt) AS total_amt FROM Orders O JOIN Salespeople S ON O.snum = S.snum GROUP BY S.snum	1 row(s) returned
77	14:19:16	SELECT DISTINCT C cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.odate = '1994-1-1'	4 row(s) returned
78	14:22:10	SELECT S.sname, S.snum FROM Salespeople S JOIN Cust C ON S.snum = C.cnum GROUP BY S.snum HAVING COUNT(C.cnum) > 1	2 row(s) returned
79	14:24:06	SELECT O.onum, S.sname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnu...	8 row(s) returned
80	14:25:39	SELECT O.onum, O.amt FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.amt > (SELECT AVG(amt) FROM Orders WHERE cnum = C.cnum)	3 row(s) returned

**59. Find the sums of the amounts from order table grouped by date, eliminating all those dates where the sum was not at least 2000 above the maximum amount.**

```

54
55
56 •   SELECT odate, SUM(amt) AS total_amt
57   FROM Orders
58   GROUP BY odate
59   HAVING SUM(amt) > (SELECT MAX(amt) + 2000 FROM Orders);
60
61
62

```

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

odate	total_amt

Result 65 x

Output ::

Action Output

#	Time	Action	Message
76	14:17:55	SELECT S.sname, SUM(O.amt) AS total_amt FROM Orders O JOIN Salespeople S ON O.snum = S.snum GROUP BY O.odate	1 row(s) returned
77	14:19:16	SELECT DISTINCT C.cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.odate = '1994-1...' 4 row(s) returned	
78	14:22:10	SELECT S.sname, S.snum FROM Salespeople S JOIN Cust C ON S.snum = C.cnum GROUP BY S.snum HAVING S.sname = 'John' 2 row(s) returned	
79	14:24:06	SELECT O.onum, O.sname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnu...	8 row(s) returned
80	14:25:39	SELECT O.onum, O.amt FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.amt > (SELECT AVG(...)	3 row(s) returned
81	14:26:57	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate HAVING SUM(amt) > (SELECT MAX(...)	0 row(s) returned

Result Only

**60. Find names and numbers of all customers with ratings equal to the maximum for their city.**

```

55
56 •   SELECT C.cname, C.cnum
57   FROM Cust C
58   WHERE C.rating = (SELECT MAX(rating) FROM Cust WHERE city = C.city);
59
60
61
62

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: |

cname	cnum
Giovanne	2002
Liu	2003
Grass	2004
Clemens	2006
Unknown	2008
*	*

Cust 66 x

Output ::

Action Output

#	Time	Action	Message
77	14:19:16	SELECT DISTINCT C.cname FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.odate = '1994-1...' 4 row(s) returned	
78	14:22:10	SELECT S.sname, S.snum FROM Salespeople S JOIN Cust C ON S.snum = C.cnum GROUP BY S.snum HAVING S.sname = 'John' 2 row(s) returned	
79	14:24:06	SELECT O.onum, O.sname FROM Orders O JOIN Salespeople S ON O.snum = S.snum JOIN Cust C ON O.cnu...	8 row(s) returned
80	14:25:39	SELECT O.onum, O.amt FROM Orders O JOIN Cust C ON O.cnum = C.cnum WHERE O.amt > (SELECT AVG(...)	3 row(s) returned
81	14:26:57	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate HAVING SUM(amt) > (SELECT MAX(...)	0 row(s) returned
82	14:29:11	SELECT C.cname, C.cnum FROM Cust C WHERE C.rating = (SELECT MAX(rating) FROM Cust WHERE city ... 5 row(s) returned	

Apply

**61. Find all salespeople who have customers in their cities who they don't service. ( Both way using Join and Correlated subquery.)**

```

55
56  -- Using JOIN
57 •   SELECT DISTINCT S.sname FROM Salespeople S JOIN Cust C ON S.city = C.city
58 WHERE S.snum != C.snum;
59
60  -- Using Correlated Subquery
61 •   SELECT S.sname FROM Salespeople S WHERE EXISTS (SELECT 1 FROM Cust C
62     WHERE C.city = S.city AND C.snum != S.snum
63 );
64
65

```

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

sname
Peel
Motika

Result 69 Salespeople70 x Read Only

Action Output

#	Time	Action	Message
81	14:26:57	SELECT odate, SUM(amt) AS total_amt FROM Orders GROUP BY odate HAVING SUM(amt) > (SELECT MAX(...)	0 row(s) returned
82	14:29:11	SELECT C.cname, C.cnum FROM Cust C WHERE C.rating = (SELECT MAX(rating) FROM Cust WHERE city ...)	5 row(s) returned
83	14:31:58	SELECT DISTINCT S.sname FROM Salespeople S JOIN Cust C ON S.city = C.city WHERE S.snum != C.snum...	2 row(s) returned
84	14:31:58	SELECT S.sname FROM Salespeople S WHERE EXISTS ( SELECT 1 FROM Cust C WHERE C.city = ...)	2 row(s) returned
85	14:33:08	SELECT DISTINCT S.sname FROM Salespeople S JOIN Cust C ON S.city = C.city WHERE S.snum != C.snum...	2 row(s) returned
86	14:33:08	SELECT S.sname FROM Salespeople S WHERE EXISTS (SELECT 1 FROM Cust C WHERE C.city = S.city...)	2 row(s) returned

**62. Extract cnum,cname and city from customer table if and only if one or more of the customers in the table are located in San Jose.**

```

17
18 •   SELECT cnum, cname, city
19   FROM cust
20  WHERE EXISTS (
21    SELECT 1 FROM cust WHERE city = 'San Jose'
22 )

```

Result Grid | Filter Rows: \_\_\_\_\_ | Edit: | Export/Import: | Wrap Cell Content:

cnum	cname	city
2001	Hoffman	London
2002	Giovanne	Rome
2003	Liu	San Jose
2004	Grass	Brelin
2006	Clemens	London
2007	Pereira	Rome
2008	Unknown	Unknown
*	NULL	NULL

cust 10 x

Action Output

#	Time	Action	Message
16	14:53:11	select * from cust LIMIT 0, 1000	7 row(s) returned
17	14:53:33	SELECT crnum, cname, city FROM customer WHERE EXISTS ( SELECT 1 FROM customer WHERE city = ...)	Error Code: 1146. Table 'persons.customer' doesn't exist
18	14:55:15	SELECT crnum, cname, city FROM customer WHERE EXISTS ( SELECT 1 FROM customer WHERE ...)	Error Code: 1146. Table 'persons.customer' doesn't exist
19	14:56:21	SELECT crnum, cname, city FROM customer WHERE city = 'San Jose' LIMIT 0, 1000	Error Code: 1146. Table 'persons.customer' doesn't exist
20	14:57:47	SELECT crnum, cname, city FROM cust WHERE EXISTS ( SELECT 1 FROM cust WHERE city = 'San Jose...')	7 row(s) returned
21	14:58:03	SELECT crnum, cname, city FROM cust WHERE EXISTS ( SELECT 1 FROM cust WHERE city = 'San Jose...')	7 row(s) returned

### 63. Find salespeople no. who have multiple customers.

```
17
18
19 •   SELECT snum
20   FROM cust
21   GROUP BY snum
22   HAVING COUNT(*) > 1;
```

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

snum
1002
1007

cust 12 x

Output:

#	Time	Action	Message
19	14:56:21	SELECT cnum, cname, city FROM customer WHERE city = 'San Jose' LIMIT 0, 1000	Error Code: 1146. Table 'persons.customer' doesn't exist
20	14:57:47	SELECT cnum, cname, city FROM cust WHERE EXISTS ( SELECT 1 FROM cust WHERE city = 'San Jose...' )	7 row(s) returned
21	14:58:03	SELECT cnum, cname, city FROM cust WHERE EXISTS ( SELECT 1 FROM cust WHERE city = 'San Jose...' )	7 row(s) returned
22	14:59:39	SELECT snum FROM customer GROUP BY snum HAVING COUNT(*) > 1 LIMIT 0, 1000	Error Code: 1146. Table 'persons.customer' doesn't exist
23	15:00:04	SELECT snum FROM cust GROUP BY snum HAVING COUNT(*) > 1 LIMIT 0, 1000	2 row(s) returned
24	15:00:19	SELECT snum FROM cust GROUP BY snum HAVING COUNT(*) > 1 LIMIT 0, 1000	2 row(s) returned

Result Only

### 64. Find salespeople number, name and city who have multiple customers.

```
26
27 •   SELECT s.snum, s.sname, s.city
28   FROM salespeople s
29   JOIN cust c ON s.snum = c.snum
30   GROUP BY s.snum, s.sname, s.city
31   HAVING COUNT(c.cnum) > 1;
```

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

snum	sname	city
1002	Serres	San Jose
1007	Rafkin	Barcelona

Result 17 x

Output:

#	Time	Action	Message
31	15:05:22	SELECT cnum, cname, city FROM customer WHERE EXISTS ( SELECT 1 FROM customer WHERE ... )	Error Code: 1146. Table 'persons.customer' doesn't exist
32	15:05:35	show tables	3 row(s) returned
33	15:06:19	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN customer c ON s.snum = c.snum GROUP BY s.snum, s...	Error Code: 1146. Table 'persons.salesman' doesn't exist
34	15:07:00	SELECT s.snum, s.sname, s.city FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	Error Code: 1146. Table 'persons.saleperson' doesn't exist
35	15:07:13	show tables	3 row(s) returned
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN customer c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned

Result Only

## 65. Find salespeople who serve only one customer.

```

26
27 •   SELECT s.snum, s.sname
28   FROM salespeople s
29   JOIN cust c ON s.snum = c.snum
30   GROUP BY s.snum, s.sname
31   HAVING COUNT(*) = 1;

```

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

snum	sname
1001	Peel
1003	Axelrod
1004	Motika

**Action Output**

#	Time	Action	Message
31	15:05:22	SELECT crnum, cname, city FROM customer WHERE EXISTS ( SELECT 1 FROM customer WHERE ... )	Error Code: 1146. Table 'persons.customer' doesn't exist
32	15:05:35	show tables	3 row(s) returned
33	15:06:19	SELECT s.snum, s.sname, s.city FROM salesman s JOIN customer c ON s.snum = c.snum GROUP BY s.snum, s...	Error Code: 1146. Table 'persons.saleman' doesn't exist
34	15:07:00	SELECT s.snum, s.sname, s.city FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	Error Code: 1146. Table 'persons.salesperson' doesn't exist
35	15:07:13	show tables	3 row(s) returned
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
37	15:09:29	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
38	15:11:06	SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	Error Code: 1146. Table 'persons.saleperson' doesn't exist
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	3 row(s) returned

## 66. Extract rows of all salespeople with more than one current order.

```

25
26
27 •   SELECT s.*
28   FROM salespeople s
29   JOIN orders o ON s.snum = o.snum
30   GROUP BY s.snum, s.sname, s.city, s.comm
31   HAVING COUNT(*) > 1;

```

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

snum	sname	city	comm
1001	Peel	London	0.12
1002	Serres	San Jose	0.13
1007	Rafkin	Barcelona	0.15

**Action Output**

#	Time	Action	Message
32	15:05:35	show tables	3 row(s) returned
33	15:06:19	SELECT s.snum, s.sname, s.city FROM salesman s JOIN customer c ON s.snum = c.snum GROUP BY s.snum, s...	Error Code: 1146. Table 'persons.saleman' doesn't exist
34	15:07:00	SELECT s.snum, s.sname, s.city FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	Error Code: 1146. Table 'persons.salesperson' doesn't exist
35	15:07:13	show tables	3 row(s) returned
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
37	15:09:29	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
38	15:11:06	SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	Error Code: 1146. Table 'persons.saleperson' doesn't exist
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	3 row(s) returned
40	15:18:23	SELECT s.* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.city, s.co...	3 row(s) returned

## 67. Find all salespeople who have customers with a rating of 300. (use EXISTS)

```

26
27 •   SELECT s.*  

28     FROM salespeople s  

29   WHERE EXISTS (  

30       SELECT 1 FROM cust c WHERE c.snum = s.snum AND c.rating = 300  

31   );

```

**Result Grid** | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | **Result Grid** | **Form Editor**

snum	sname	city	comm
1002	Serres	San Jose	0.13
1007	Rafkin	Barcelona	0.15
NULL	NULL	NULL	NULL

**salespeople21** x

Output:

Action Output

#	Time	Action	Message
33	15:06:19	SELECT s.snum, s.sname, s.city FROM salesman s JOIN customer c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	Error Code: 1146. Table 'persons.salesman' doesn't exist
34	15:07:00	SELECT s.snum, s.sname, s.city FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	Error Code: 1146. Table 'persons.salesperson' doesn't exist
35	15:07:13	show tables	3 row(s) returned
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	2 row(s) returned
37	15:09:29	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	2 row(s) returned
38	15:11:06	SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname	Error Code: 1146. Table 'persons.salesperson' doesn't exist
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname	3 row(s) returned
40	15:18:23	SELECT s.* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.city, s.co...	3 row(s) returned
41	15:19:21	SELECT s.* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum A...	2 row(s) returned

## 68. Find all salespeople who have customers with a rating of 300. (use Join).

```

26
27 •   SELECT DISTINCT s.*  

28     FROM salespeople s  

29   JOIN cust c ON s.snum = c.snum  

30   WHERE c.rating = 300;

```

**Result Grid** | Filter Rows: | Export: | Wrap Cell Content: | **Result Grid** | **Form Editor**

snum	sname	city	comm
1002	Serres	San Jose	0.13
1007	Rafkin	Barcelona	0.15

**Result 22** x

Output:

Action Output

#	Time	Action	Message
34	15:07:00	SELECT s.snum, s.sname, s.city FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	Error Code: 1146. Table 'persons.salesperson' doesn't exist
35	15:07:13	show tables	3 row(s) returned
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	2 row(s) returned
37	15:09:29	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city	2 row(s) returned
38	15:11:06	SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname	Error Code: 1146. Table 'persons.salesperson' doesn't exist
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname	3 row(s) returned
40	15:18:23	SELECT s.* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.city, s.co...	3 row(s) returned
41	15:19:21	SELECT s.* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum A...	2 row(s) returned
42	15:20:28	SELECT DISTINCT s.* FROM salespeople s JOIN cust c ON s.snum = c.snum WHERE c.rating = 300 LIMIT 0...	2 row(s) returned

**69. Select all salespeople with customers located in their cities who are not assigned to them. (use EXISTS).**

```

26
27 •   SELECT s.*
28   FROM salespeople s
29 WHERE EXISTS (SELECT 1 FROM cust c WHERE c.city = s.city AND c.snum <> s.snum);

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Apply

snum	sname	city	comm
1001	Peel	London	0.12
1004	Motika	London	0.11
*	HULL	HULL	0.11

salespeople23 x

Output:

#	Time	Action	Message
35	15:07:13	show tables	3 row(s) returned
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
37	15:09:29	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
38	15:11:06	SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam... Error Code: 1146. Table 'persons.salesperson' doesn't exist	Error Code: 1146. Table 'persons.salesperson' doesn't exist
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	3 row(s) returned
40	15:18:23	SELECT s.* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.cit...	3 row(s) returned
41	15:19:21	SELECT s.* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum A...	2 row(s) returned
42	15:20:28	SELECT DISTINCT s.* FROM salespeople s JOIN cust c ON s.snum = c.snum WHERE c.rating = 300 LIMIT 0...	2 row(s) returned
43	15:21:39	SELECT s.* FROM salespeople s WHERE EXISTS (SELECT 1 FROM cust c WHERE c.city = s.city AND c.s...	2 row(s) returned

**70. Extract from customers table every customer assigned the a salesperson who currently has at least one other customer ( besides the customer being selected) with orders in order table.**

```

27 •   SELECT c1.*
28   FROM cust c1
29 WHERE EXISTS (
30   SELECT 1 FROM cust c2
31   JOIN orders o ON c2.cnum = o.cnum
32   WHERE c2.snum = c1.snum AND c2.cnum <> c1.cnum
33 );

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Apply

cnum	cname	city	rating	snum
2003	Liu	San Jose	300	1002
2004	Grass	Brelin	100	100
2006	Clemens	London	300	1007
2008	Unknown	Unknown	100	1007
*	HULL	HULL	HULL	HULL

cust 24 x

Output:

#	Time	Action	Message
36	15:07:38	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
37	15:09:29	SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s...	2 row(s) returned
38	15:11:06	SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam... Error Code: 1146. Table 'persons.salesperson' doesn't exist	Error Code: 1146. Table 'persons.salesperson' doesn't exist
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	3 row(s) returned
40	15:18:23	SELECT s.* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.cit...	3 row(s) returned
41	15:19:21	SELECT s.* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum A...	2 row(s) returned
42	15:20:28	SELECT DISTINCT s.* FROM salespeople s JOIN cust c ON s.snum = c.snum WHERE c.rating = 300 LIMIT 0...	2 row(s) returned
43	15:21:39	SELECT s.* FROM salespeople s WHERE EXISTS (SELECT 1 FROM cust c WHERE c.city = s.city AND c.s...	2 row(s) returned
44	15:22:46	SELECT c1.* FROM cust c1 WHERE EXISTS ( SELECT 1 FROM cust c2 JOIN orders o ON c2.cnum = ...	4 row(s) returned

## 71. Find salespeople with customers located in their cities ( using both ANY and IN).

```

26
27 •   SELECT *
28     FROM cust c1
29     WHERE c1.rating > ALL (
30         SELECT rating
31         FROM cust c2
32         WHERE c2.city = 'Rome'
33     )
34

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Apply

cnum	cname	city	rating	snum
2003	Liu	San Jose	300	1002
2006	Clemens	London	300	1007
•	NULL	NULL	NULL	NULL

cust 25 x

Output:

Action Output | Message

- 37 15:09:29 SELECT s.snum, s.sname, s.city FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname, s.city 2 row(s) returned
- 38 15:11:06 SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname Error Code: 1146. Table 'persons.salesperson' doesn't exist
- 39 15:16:44 SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname 3 row(s) returned
- 40 15:18:23 SELECT s.\* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.city, s.co... 3 row(s) returned
- 41 15:19:21 SELECT s.\* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum AND c.city = s.city ) 2 row(s) returned
- 42 15:20:28 SELECT DISTINCT s.\* FROM salespeople s JOIN cust c ON s.snum = c.snum WHERE c.rating = 300 LIMIT 0... 2 row(s) returned
- 43 15:21:39 SELECT s.\* FROM salespeople s WHERE EXISTS (SELECT 1 FROM cust c WHERE c.city = s.city AND c.snum = s.snum ) 2 row(s) returned
- 44 15:22:46 SELECT c1.\* FROM cust c1 WHERE EXISTS ( SELECT 1 FROM cust c2 JOIN orders o ON c2.cnum = o.cnum WHERE c2.city = s.city ) 4 row(s) returned
- 45 15:26:28 SELECT \* FROM cust c1 WHERE c1.rating > ALL ( SELECT rating FROM cust c2 WHERE c2.city = s.city ) 2 row(s) returned

## 72. Find all salespeople for whom there are customers that follow them in alphabetical order. (Using ANY and EXISTS)

```

27 •   SELECT s.snum, s.sname
28     FROM salespeople s
29     WHERE EXISTS (
30         SELECT 1
31         FROM cust c
32         WHERE c.snum = s.snum
33         AND c cname > s.sname
34     )
35

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Apply

snum	sname
1003	Axelrod
1004	Motika
1007	Rafkin
•	NULL

salespeople26 x

Output:

Action Output | Message

- 38 15:11:06 SELECT s.snum, s.sname FROM salesperson s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname Error Code: 1146. Table 'persons.salesperson' doesn't exist
- 39 15:16:44 SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.sname 3 row(s) returned
- 40 15:18:23 SELECT s.\* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.city, s.co... 3 row(s) returned
- 41 15:19:21 SELECT s.\* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum AND c.city = s.city ) 2 row(s) returned
- 42 15:20:28 SELECT DISTINCT s.\* FROM salespeople s JOIN cust c ON s.snum = c.snum WHERE c.rating = 300 LIMIT 0... 2 row(s) returned
- 43 15:21:39 SELECT s.\* FROM salespeople s WHERE EXISTS (SELECT 1 FROM cust c WHERE c.city = s.city AND c.snum = s.snum ) 2 row(s) returned
- 44 15:22:46 SELECT c1.\* FROM cust c1 WHERE EXISTS ( SELECT 1 FROM cust c2 JOIN orders o ON c2.cnum = o.cnum WHERE c2.city = s.city ) 4 row(s) returned
- 45 15:26:28 SELECT \* FROM cust c1 WHERE c1.rating > ALL ( SELECT rating FROM cust c2 WHERE c2.city = s.city ) 2 row(s) returned
- 46 15:28:43 SELECT s.snum, s.sname FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum ) 3 row(s) returned

### 73. Select customers who have a greater rating than any customer in rome.

```

27 •   SELECT *
28     FROM cust c1
29   WHERE c1.rating > ALL (
30       SELECT rating
31         FROM cust c2
32       WHERE c2.city = 'Rome'
33   );
34

```

**Result Grid**

cnum	cname	city	rating	snum
2003	Liu	San Jose	300	1002
2006	Clemens	London	300	300 7
*				

**cust 27**

**Action Output**

#	Time	Action	Message
39	15:16:44	SELECT s.snum, s.sname FROM salespeople s JOIN cust c ON s.snum = c.snum GROUP BY s.snum, s.snam...	3 row(s) returned
40	15:18:23	SELECT s.* FROM salespeople s JOIN orders o ON s.snum = o.snum GROUP BY s.snum, s.sname, s.city, s.co...	3 row(s) returned
41	15:19:21	SELECT s.* FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum A...	2 row(s) returned
42	15:20:28	SELECT DISTINCT s.* FROM salespeople s JOIN cust c ON s.snum = c.snum WHERE c.rating = 300 LIMIT 0...	2 row(s) returned
43	15:21:39	SELECT s.* FROM salespeople s WHERE EXISTS (SELECT 1 FROM cust c WHERE c.city = s.city AND c.s...	2 row(s) returned
44	15:22:46	SELECT c1.* FROM cust c1 WHERE EXISTS ( SELECT 1 FROM cust c2 JOIN orders o ON c2.cnum = ...)	4 row(s) returned
45	15:26:28	SELECT * FROM cust c1 WHERE c1.rating > ALL ( SELECT rating FROM cust c2 WHERE c2.city = '...')	2 row(s) returned
46	15:28:43	SELECT s.snum, s.sname FROM salespeople s WHERE EXISTS ( SELECT 1 FROM cust c WHERE c...	3 row(s) returned
47	15:29:57	SELECT * FROM cust c1 WHERE c1.rating > ALL ( SELECT rating FROM cust c2 WHERE c2.city = '...')	2 row(s) returned

### 74. Select all orders that had amounts that were greater than atleast one of the orders from Oct 6<sup>th</sup>.

```

26
27 •   SELECT *
28     FROM orders o1
29   WHERE o1.amt > ANY (
30       SELECT amt
31         FROM orders o2
32       WHERE STR_TO_DATE(o2.odate, '%d-%b-%y') = STR_TO_DATE('06-OCT-94', '%d-%b-%y')
33   );
34

```

**orders 30**

**Action Output**

#	Time	Action	Message
48	15:31:15	SELECT * FROM orders o1 WHERE o1.amt > ANY ( SELECT amt FROM orders o2 WHERE o2.odate ...)	Error Code: 1525. Incorrect DATE value: '06-OCT-94'
49	15:31:24	SELECT * FROM orders o1 WHERE o1.amt > ANY ( SELECT amt FROM orders o2 WHERE o2.odate ...)	Error Code: 1525. Incorrect DATE value: '06-OCT-94'
50	15:31:33	SELECT * FROM orders o1 WHERE o1.amt > ANY ( SELECT amt FROM orders o2 WHERE o2.odate ...)	Error Code: 1525. Incorrect DATE value: '06-OCT-94'
51	15:33:54	SELECT * FROM orders WHERE amount > ANY (SELECT amount FROM orders WHERE order_date = '2025...')	Error Code: 1054. Unknown column 'amount' in 'IN/ALL/ANY subquery'
52	15:35:07	SELECT * FROM orders o1 WHERE o1.amt > ANY ( SELECT amt FROM orders o2 WHERE o2.odate ...)	Error Code: 1525. Incorrect DATE value: '06-OCT-94'
53	15:36:22	SELECT * FROM orders o1 WHERE o1.amt > ANY ( SELECT amt FROM orders o2 WHERE o2.odate ...)	Error Code: 1525. Incorrect DATE value: '06-OCT-1994'
54	15:44:29	show tables	3 row(s) returned
55	15:44:52	select * from orders LIMIT 0, 1000	10 row(s) returned
56	15:45:53	SELECT * FROM orders o1 WHERE o1.amt > ANY ( SELECT amt FROM orders o2 WHERE STR_TO...	0 row(s) returned

**75. Find all orders with amounts smaller than any amount for a customer in San Jose.  
(Both using ANY and without ANY)**

```

28 • SELECT *
29   FROM orders o1
30   WHERE o1.amt < ANY (
31     SELECT amt
32     FROM orders o2
33     JOIN cust c ON o2.cnum = c.cnum
34     WHERE c.city = 'San Jose'
35   );
36
37 -- Without using ANY:
38 • SELECT *
39   FROM orders o1
40   WHERE o1.amt < (
41     SELECT MIN(amt)
42     FROM orders o2
43     JOIN cust c ON o2.cnum = c.cnum
44     WHERE c.city = 'San Jose'
45 );

```

The screenshot shows the MySQL Workbench interface with two SQL queries in the editor. The first query uses the ANY operator to find orders with amounts less than any amount for a customer in San Jose. The second query uses a subquery with MIN(amt) to achieve the same result without ANY.

**Result Grid:**

onum	amt	odate	cnum	snum
3001	18.69	1994-10-03	2008	1007
3002	1900.10	1994-10-03	2007	1004
3003	767.19	1994-10-03	2001	1001
3006	1098.16	1994-10-04	2008	1007
3007	75.75	1994-10-05	2004	1002
3008	4723.00	1994-10-05	2008	1001
3009	1713.23	1994-10-04	2002	1003
3010	1309.95	1994-10-06	2004	1002
NULL	NULL	NULL	NULL	NULL

**76. Select those customers whose ratings are higher than every customer in Paris. (Using both ALL and NOT EXISTS).**

```

29 • SELECT *
30   FROM cust c1
31   WHERE c1.rating > ALL (
32     SELECT rating
33     FROM cust c2
34     WHERE c2.city = 'Paris'
35   );
36
37 -- Using NOT EXISTS:
38 • SELECT *
39   FROM cust c1
40   WHERE NOT EXISTS (
41     SELECT 1
42     FROM cust c2
43     WHERE c2.city = 'Paris' AND c1.rating <= c2.rating
44   );
45

```

The screenshot shows the MySQL Workbench interface with two SQL queries for question 76. The first query uses the ALL operator to find customers with ratings higher than all customers in Paris. The second query uses NOT EXISTS to achieve the same result.

**Result Grid:**

cnum	name	city	rating	snum
2001	Hoffman	London	100	1001
2002	Giovanni	Rome	200	1003
2003	Lia	San Jose	300	1002
2004	Grass	Berlin	100	1002
2005	Clemens	London	300	1007
2007	Pereira	Rome	100	1004
2008	Unknown	Unknown	100	1007
NULL	NULL	NULL	NULL	NULL

**77. Select all customers whose ratings are equal to or greater than ANY of the Seeres.**

```

30
31 •   SELECT *
32     FROM cust c1
33     WHERE c1.rating >= ANY (
34         SELECT rating
35         FROM cust c2
36         WHERE c2.snum = 1007 -- Assuming Seeres' snum is 1007
37     );

```

**Result Grid**

cnum	cname	city	rating	snum
2001	Hoffman	London	100	1001
2002	Giovanne	Rome	200	1003
2003	Liu	San Jose	300	1002
2004	Grass	Brelin	100	1002
2006	Clemens	London	300	1007
2007	Pereira	Rom	100	1004
2008	Unknown	Unknown	100	1007

**Output**

- Action Output
 

#	Time	Action	Message
57	15:47:52	SELECT * FROM orders o1 WHERE o1.amt < ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.cnum = c.snum )	8 row(s) returned
58	15:47:52	SELECT * FROM orders o1 WHERE o1.amt < ( SELECT MIN(amt) FROM orders o2 JOIN cust c ON o2.cnum = c.snum )	8 row(s) returned
59	15:50:00	SELECT * FROM cust c1 WHERE c1.rating > ALL ( SELECT rating FROM cust c2 WHERE c2.city = 'Paris' )	7 row(s) returned
60	15:50:00	SELECT * FROM cust c1 WHERE NOT EXISTS ( SELECT 1 FROM cust c2 WHERE c2.city = 'Paris' )	7 row(s) returned
61	15:51:59	SELECT * FROM cust c1 WHERE c1.rating >= ANY ( SELECT rating FROM cust c2 WHERE c2.snum = 1007 )	7 row(s) returned

**78. Find all salespeople who have no customers located in their city. ( Both using ANY and ALL)**

```

14
15 •   SELECT *
16     FROM salespeople s
17     WHERE NOT EXISTS (
18         SELECT 1
19         FROM cust c
20         WHERE c.city = s.city AND c.snum = s.snum
21     );
22 •   -- Using ALL:
23     SELECT *
24     FROM salespeople s
25     WHERE NOT EXISTS (
26         SELECT 1
27         FROM cust c
28         WHERE c.snum = s.snum AND c.city = s.city
29     );
30

```

**Result Grid**

snum	sname	city	comm
1003	Axelrod	New york	0.10
1004	Motika	London	0.11
1007	Rafkin	Barcelona	0.15

**Output**

**79. Find all orders for amounts greater than any for the customers in London.**

```

12
13 •   SELECT *
14     FROM orders o
15     WHERE o.amt > ANY (
16         SELECT amt
17         FROM orders o2
18         JOIN cust c ON o2.cnum = c.cnum
19         WHERE c.city = 'London'
20     );

```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content: | Result Grid | Form Editor | Apply | Revert

onum	amt	odate	cnum	snum
3002	1900.10	1994-10-03	2007	1004
3005	5160.45	1994-10-03	2003	1002
3006	1098.16	1994-10-04	2008	1007
3008	4723.00	1994-10-05	2006	1001
3009	1713.23	1994-10-04	2002	1003
3010	1309.95	1994-10-06	2004	1002
3011	9891.88	1994-10-06	2006	1001
*	NULL	NULL	NULL	NULL

orders 39 ×

Output:

Action Output | # Time Action | Message

- 61 15:51:59 SELECT \* FROM cust c1 WHERE c1.rating >= ANY ( SELECT rating FROM cust c2 WHERE c2.snum... 7 row(s) returned
- 62 15:53:47 SELECT \* FROM salespeople s WHERE NOT EXISTS ( SELECT 1 FROM cust c WHERE c.city = s.ci... 3 row(s) returned
- 63 15:53:47 SELECT \* FROM salespeople s WHERE NOT EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.... 3 row(s) returned
- 64 15:56:33 SELECT \* FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.c... 7 row(s) returned
- 65 15:56:56 SELECT \* FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.c... 7 row(s) returned

**80. Find all salespeople and customers located in london.**

```

12
13 •   SELECT s.snum, s.sname, c.cnum, c cname
14     FROM salespeople s
15     JOIN cust c ON c.snum = s.snum
16     WHERE c.city = 'London';
17

```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: | Result Grid | Form Editor | Read Only

snum	sname	cnum	cname
1001	Peel	2001	Hoffman
1007	Rafkin	2006	Clemens

Result 40 ×

Output:

Action Output | # Time Action | Message

- 62 15:53:47 SELECT \* FROM salespeople s WHERE NOT EXISTS ( SELECT 1 FROM cust c WHERE c.city = s.ci... 3 row(s) returned
- 63 15:53:47 SELECT \* FROM salespeople s WHERE NOT EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.... 3 row(s) returned
- 64 15:56:33 SELECT \* FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.c... 7 row(s) returned
- 65 15:56:56 SELECT \* FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.c... 7 row(s) returned
- 66 15:58:46 SELECT s.snum, s.sname, c.cnum, c cname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE ... 2 row(s) returned

## 81. For every salesperson, dates on which highest and lowest orders were brought.

```

13 •   SELECT s.snum, MAX(o.odate) AS highest_order_date, MIN(o.odate) AS lowest_order_date
14     FROM salespeople s
15   JOIN orders o ON o.snum = s.snum
16 GROUP BY s.snum

```

**Result Grid**

snum	highest_order_date	lowest_order_date
1001	1994-10-06	1994-10-03
1002	1994-10-06	1994-10-03
1003	1994-10-04	1994-10-04
1004	1994-10-03	1994-10-03
1007	1994-10-04	1994-10-03

**Action Output**

#	Time	Action	Message
63	15:53:47	SELECT * FROM salespeople s WHERE NOT EXISTS ( SELECT 1 FROM cust c WHERE c.snum = s.snum )	3 row(s) returned
64	15:56:33	SELECT * FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.cust_id = c.cust_id )	7 row(s) returned
65	15:56:56	SELECT * FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.cust_id = c.cust_id )	7 row(s) returned
66	15:58:46	SELECT s.snum, s.sname, c.cnum, c cname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE ...	2 row(s) returned
67	15:59:51	SELECT s.snum, MAX(o.odate) AS highest_order_date, MIN(o.odate) AS lowest_order_date FROM salespeople s	5 row(s) returned

## 82. List all of the salespeople and indicate those who don't have customers in their cities as well as those who do have.

```

13 •   SELECT s.snum, s.sname,
14       CASE
15         WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.snum AND c.city = s.city) THEN 'Has customers in city'
16         ELSE 'Does not have customers in city'
17       END AS city_status
18   FROM salespeople s

```

**Result Grid**

snum	sname	city_status
1001	Peel	Has customers in city
1002	Serres	Has customers in city
1003	Axelrod	Does not have customers in city
1004	Motika	Does not have customers in city
1007	Rafkin	Does not have customers in city

**Action Output**

#	Time	Action	Message
64	15:56:33	SELECT * FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.cust_id = c.cust_id )	7 row(s) returned
65	15:56:56	SELECT * FROM orders o WHERE o.amt > ANY ( SELECT amt FROM orders o2 JOIN cust c ON o2.cust_id = c.cust_id )	7 row(s) returned
66	15:58:46	SELECT s.snum, s.sname, c.cnum, c cname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE ...	2 row(s) returned
67	15:59:51	SELECT s.snum, MAX(o.odate) AS highest_order_date, MIN(o.odate) AS lowest_order_date FROM salespeople s	5 row(s) returned
68	16:00:51	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.snum AND c.city = s.city) THEN 'Has customers in city' ELSE 'Does not have customers in city' END AS city_status FROM salespeople s	5 row(s) returned

**83. Append strings to the selected fields, indicating whether or not a given salesperson was matched to a customer in his city.**

```

12
13 •   SELECT s.snum, s.sname,
14     CASE
15       WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.snum AND c.city = s.city)
16         THEN CONCAT(s.sname, ' has customers in their city')
17       ELSE CONCAT(s.sname, ' does not have customers in their city')
18     END AS city_match_status
19   FROM salespeople s
  
```

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

snum	sname	city_match_status
1001	Peel	Peel has customers in their city
1002	Serres	Serres has customers in their city
1003	Axelrod	Axelrod does not have customers in their city
1004	Motika	Motika does not have customers in their city
1007	Rafkin	Rafkin does not have customers in their city

**Action Output**

#	Time	Action	Message
65	15:56:56	SELECT * FROM orders o WHERE o.ant > ANY ( SELECT ant FROM orders o2 JOIN cust c ON o2.c...	7 row(s) returned
66	15:58:46	SELECT s.snum, s.sname, c.cnum, c cname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE ...	2 row(s) returned
67	15:59:51	SELECT s.snum, MAX(o.odate) AS highest_order_date, MIN(o.odate) AS lowest_order_date FROM salespeopl...	5 row(s) returned
68	16:00:51	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.s...	5 row(s) returned
69	16:02:00	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.s...	5 row(s) returned

**84. Create a union of two queries that shows the names, cities and ratings of all customers. Those with a rating of 200 or greater will also have the words ‘High Rating’, while the others will have the words ‘Low Rating’.**

```

12
13 •   SELECT cname, city, rating, 'High Rating' AS rating_status
14   FROM cust
15   WHERE rating >= 200
16
17 UNION
18
19   SELECT cname, city, rating, 'Low Rating' AS rating_status
20   FROM cust
21   WHERE rating < 200;
  
```

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

cname	city	rating	rating_status
Giovannie	Rome	200	High Rating
Liu	San Jose	300	High Rating
Clemens	London	300	High Rating
Hoffman	London	100	Low Rating
Grass	Brelin	100	Low Rating
Pereira	Rome	100	Low Rating
Unknown	Unknown	100	Low Rating

**Action Output**

#	Time	Action	Message
66	15:58:46	SELECT s.snum, s.sname, c.cnum, c cname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE ...	2 row(s) returned
67	15:59:51	SELECT s.snum, MAX(o.odate) AS highest_order_date, MIN(o.odate) AS lowest_order_date FROM salespeopl...	5 row(s) returned
68	16:00:51	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.s...	5 row(s) returned
69	16:02:00	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.s...	5 row(s) returned
70	16:03:02	SELECT cname, city, rating, 'High Rating' AS rating_status FROM cust WHERE rating >= 200 UNION SELE...	7 row(s) returned

**85. Write command that produces the name and number of each salesperson and each customer with more than one current order. Put the result in alphabetical order.**

```

12
13 •   SELECT s.sname, s.snum, c cname, c.cnum
14   FROM salespeople s
15   JOIN orders o ON o.snum = s.snum
16   JOIN cust c ON o.cnum = c.cnum
17   GROUP BY s.sname, s.snum, c cname, c.cnum
18   HAVING COUNT(o.cnum) > 1
19   ORDER BY s.sname, c cname;

```

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

sname	snum	cname	cnum
Peel	1001	Clemens	2006
Rafkin	1007	Unknwon	2008
Serres	1002	Grass	2004

Result 45 x Read Only

Output:

Action Output

#	Time	Action	Message
67	15:59:51	SELECT s.snum, MAX(o.odate) AS highest_order_date, MIN(o.odate) AS lowest_order_date FROM salespeople s	5 row(s) returned
68	16:00:51	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.snum) THEN 1 ELSE 0 END AS has_order	5 row(s) returned
69	16:02:00	SELECT s.snum, s.sname, CASE WHEN EXISTS (SELECT 1 FROM cust c WHERE c.snum = s.snum) THEN 1 ELSE 0 END AS has_order	5 row(s) returned
70	16:03:02	SELECT cname, city, rating, 'High Rating' AS rating_status FROM cust WHERE rating >= 200 UNION SELECT cname, city, rating, 'Low Rating' AS rating_status FROM cust WHERE rating < 200	7 row(s) returned
71	16:04:03	SELECT s.sname, s.snum, c cname, c.cnum FROM salespeople s JOIN orders o ON o.snum = s.snum JOIN cust c ON o.cnum = c.cnum	3 row(s) returned

**86. Form a union of three queries. Have the first select the snums of all salespeople in San Jose, then second the cnums of all customers in San Jose and the third the onums of all orders on Oct. 3. Retain duplicates between the last two queries, but eliminates and redundancies between either of them and the first.**

```

12
13 •   SELECT snum FROM salespeople WHERE city = 'San Jose'
14   UNION
15   SELECT cnum FROM cust WHERE city = 'San Jose'
16   UNION ALL
17   SELECT onum FROM orders
18   WHERE STR_TO_DATE(odate, '%d-%b-%y') = STR_TO_DATE('03-OCT-94', '%d-%b-%y');

```

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

snum
1002
2003

Result 46 x Read Only

Output:

Action Output

#	Time	Action	Message
70	16:03:02	SELECT cname, city, rating, 'High Rating' AS rating_status FROM cust WHERE rating >= 200 UNION SELECT cname, city, rating, 'Low Rating' AS rating_status FROM cust WHERE rating < 200	7 row(s) returned
71	16:04:03	SELECT s.sname, s.snum, c cname, c.cnum FROM salespeople s JOIN orders o ON o.snum = s.snum JOIN cust c ON o.cnum = c.cnum	3 row(s) returned
72	16:05:23	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city = 'San Jose'	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
73	16:05:55	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city = 'San Jose'	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
74	16:07:19	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city = 'San Jose'	2 row(s) returned

**87. Produce all the salesperson in London who had at least one customer there.**

```
12
13 •   SELECT DISTINCT s.snum, s.sname
14   FROM salespeople s
15   JOIN cust c ON c.snum = s.snum
16   WHERE c.city = 'London';
```

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

snum	sname
1001	Peel
1007	Rafkin

Result 47 x Read Only

Action Output

#	Time	Action	Message
71	16:04:03	SELECT s.sname, s.snum, c cname, c.cnum FROM salespeople s JOIN orders o ON o.snum = s.snum JOIN cu...	3 row(s) returned
72	16:05:23	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
73	16:05:55	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
74	16:07:19	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	2 row(s) returned
75	16:08:46	SELECT DISTINCT s.snum, s.sname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE c.city = '...'	2 row(s) returned

**88. Produce all the salesperson in London who did not have customers there.**

```
12
13 •   SELECT s.snum, s.sname
14   FROM salespeople s
15   WHERE s.city = 'London'
16   AND NOT EXISTS (
17       SELECT 1
18       FROM cust c
19       WHERE c.snum = s.snum AND c.city = 'London'
20   );
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

snum	sname
1004	Motika
HULL	HULL

salespeople48 x Apply Revert

Action Output

#	Time	Action	Message
72	16:05:23	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
73	16:05:55	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
74	16:07:19	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	2 row(s) returned
75	16:08:46	SELECT DISTINCT s.snum, s.sname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE c.city = '...'	2 row(s) returned
76	16:09:50	SELECT s.snum, s.sname FROM salespeople s WHERE s.city = 'London' AND NOT EXISTS (	1 row(s) returned

**89. We want to see salespeople matched to their customers without excluding those salesperson who were not currently assigned to any customers. (User OUTER join and UNION)**

```

12
13 •   SELECT s.snum, s.sname, c.cnum, c cname
14   FROM salespeople s
15   LEFT OUTER JOIN cust c ON c.snum = s.snum
16 UNION
17   SELECT s.snum, s.sname, NULL AS cnum, NULL AS cname
18   FROM salespeople s
19 WHERE NOT EXISTS (
20     SELECT 1 FROM cust c WHERE c.snum = s.snum
21   );

```

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

snum	sname	cnum	cname
1001	Peel	2001	Hoffman
1002	Serres	2003	Liu
1002	Serres	2004	Grass
1003	Axelrod	2002	Giovanne
1004	Motika	2007	Pereira
1007	Rafkin	2006	Clemens
1007	Rafkin	2008	Unknown

Result: 49 × Read Only

Action Output

#	Time	Action	Message
73	16:05:55	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	Error Code: 1525. Incorrect DATE value: '03-OCT-94'
74	16:07:19	SELECT snum FROM salespeople WHERE city = 'San Jose' UNION SELECT cnum FROM cust WHERE city ...	2 row(s) returned
75	16:08:46	SELECT DISTINCT s.snum, s.sname FROM salespeople s JOIN cust c ON c.snum = s.snum WHERE c.city = ...	2 row(s) returned
76	16:09:50	SELECT s.snum, s.sname FROM salespeople s WHERE s.city = 'London' AND NOT EXISTS ( SELECT 1 ...	1 row(s) returned
77	16:11:29	SELECT s.snum, s.sname, c.cnum, c cname FROM salespeople s LEFT OUTER JOIN cust c ON c.snum = s.s...	7 row(s) returned