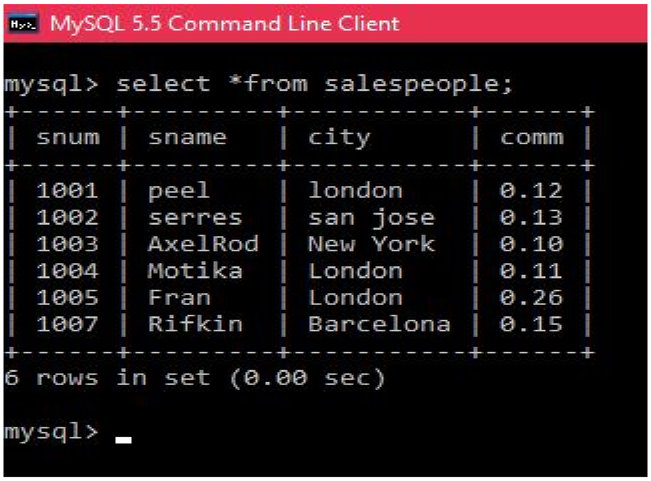
HADOOP PROJECT-1

MYSQL QUERIES WITH ANSWERS:

1. List all the columns of the Salespeople table

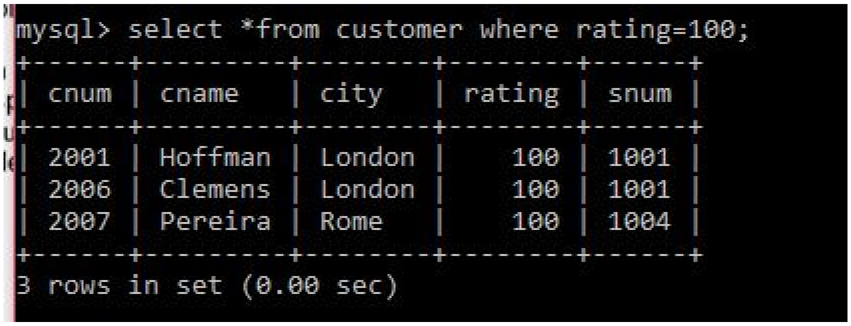
Ans:



```
mysql> select *from salespeople;
+------+---------+-----------+------+
| 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 |
| 1005 | Fran    | London    | 0.26 |
| 1007 | Rifkin  | Barcelona | 0.15 |
+------+---------+-----------+------+
6 rows in set (0.00 sec)

mysql> _
```

2. List all customers with a rating of 100.

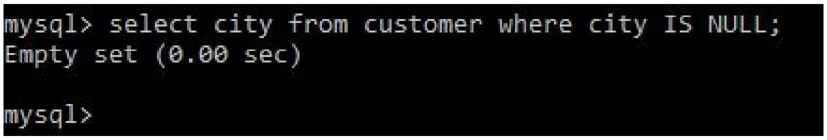
ans:



```
mysql> select *from customer where rating=100;
+------+---------+--------+--------+------+
| cnum | cname   | city   | rating | snum |
+------+---------+--------+--------+------+
| 2001 | Hoffman | London |    100 | 1001 |
| 2006 | Clemens | London |    100 | 1001 |
| 2007 | Pereira | Rome   |    100 | 1004 |
+------+---------+--------+--------+------+
3 rows in set (0.00 sec)
```

3. Find all records in the Customer table with NULL values in the city column.

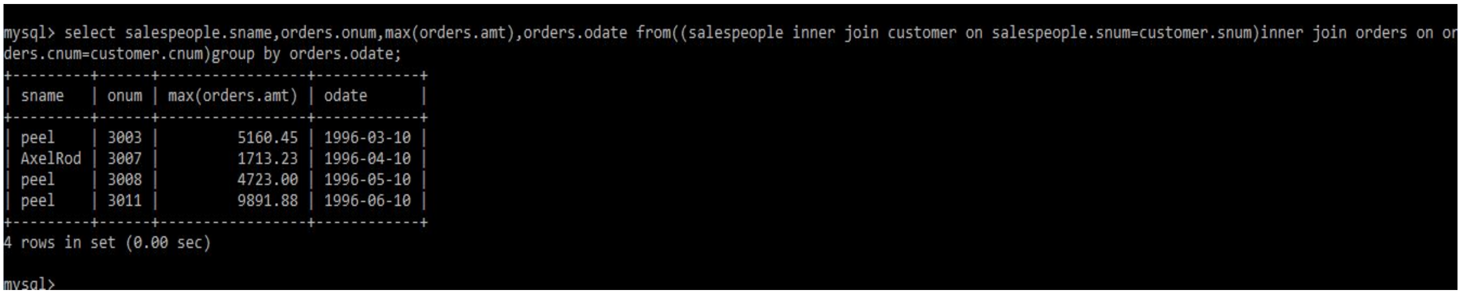
Ans:



```
mysql> select city from customer where city IS NULL;
Empty set (0.00 sec)

mysql>
```

4. Find the largest order taken by each salesperson on each date.

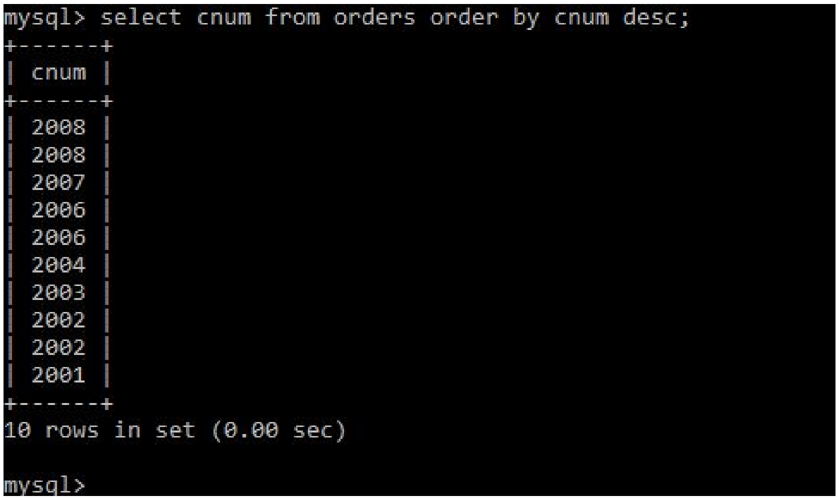
Ans:



```
mysql> select salespeople.sname,orders.onum,max(orders.amt),orders.odate from((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on or
ders.cnum=customer.cnum)group by orders.odate;
+---------+------+-----------------+------------+
| sname   | onum | max(orders.amt) | odate      |
+---------+------+-----------------+------------+
| peel    | 3003 |         5160.45 | 1996-03-10 |
| AxelRod | 3007 |         1713.23 | 1996-04-10 |
| peel    | 3008 |         4723.00 | 1996-05-10 |
| peel    | 3011 |         9891.88 | 1996-06-10 |
+---------+------+-----------------+------------+
4 rows in set (0.00 sec)

mysql>
```

5. Arrange the Orders table by descending customer number.

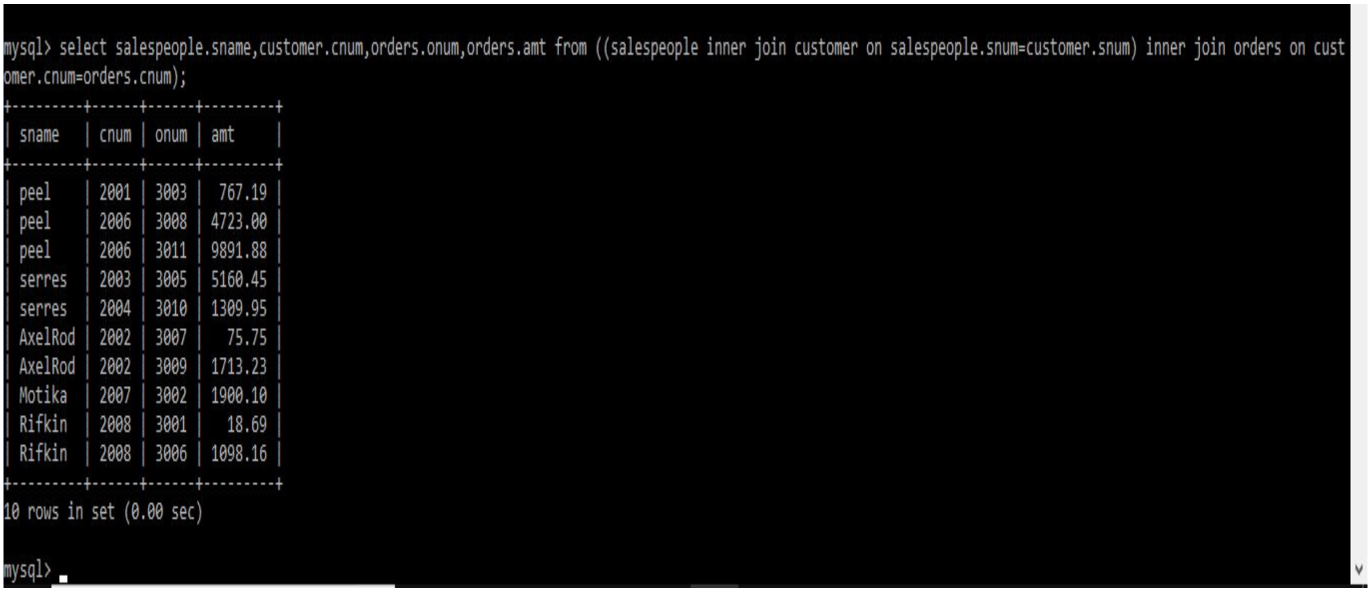
Ans:



```
mysql> select cnum from orders order by cnum desc;
+------+
| cnum |
+------+
| 2008 |
| 2008 |
| 2007 |
| 2006 |
| 2006 |
| 2004 |
| 2003 |
| 2002 |
| 2002 |
| 2001 |
+------+
10 rows in set (0.00 sec)

mysql>
```

6. Find which salespeople currently have orders in the Orders table.

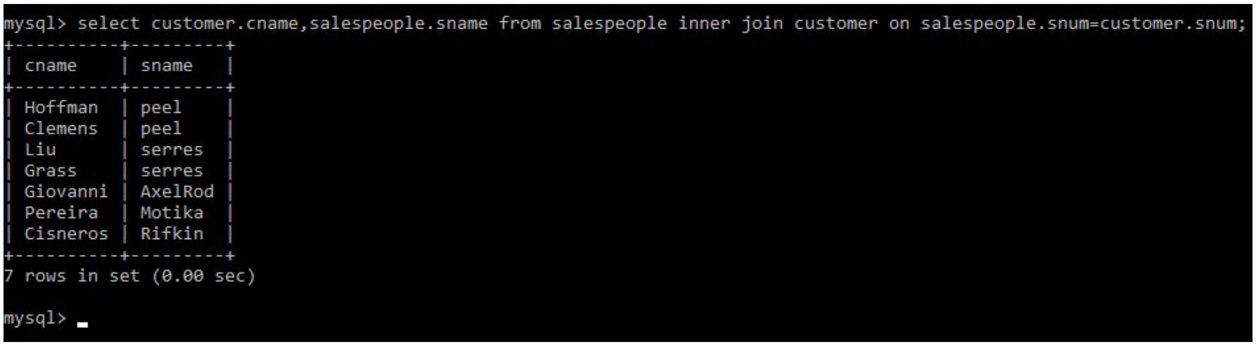
Ans:



```
mysql> select salespeople.sname,customer.cnum,orders.onum,orders.amt from ((salespeople inner join customer on salespeople.snum=customer.snum) inner join orders on cust
omer.cnum=orders.cnum);
+---------+------+------+---------+
| sname   | cnum | onum | amt     |
+---------+------+------+---------+
| peel    | 2001 | 3003 |  767.19 |
| peel    | 2006 | 3008 | 4723.00 |
| peel    | 2006 | 3011 | 9891.88 |
| serres  | 2003 | 3005 | 5160.45 |
| serres  | 2004 | 3010 | 1309.95 |
| AxelRod | 2002 | 3007 |   75.75 |
| AxelRod | 2002 | 3009 | 1713.23 |
| Motika  | 2007 | 3002 | 1900.10 |
| Rifkin  | 2008 | 3001 |   18.69 |
| Rifkin  | 2008 | 3006 | 1098.16 |
+---------+------+------+---------+
10 rows in set (0.00 sec)

mysql>
```

7. List names of all customers matched with the salespeople serving them.

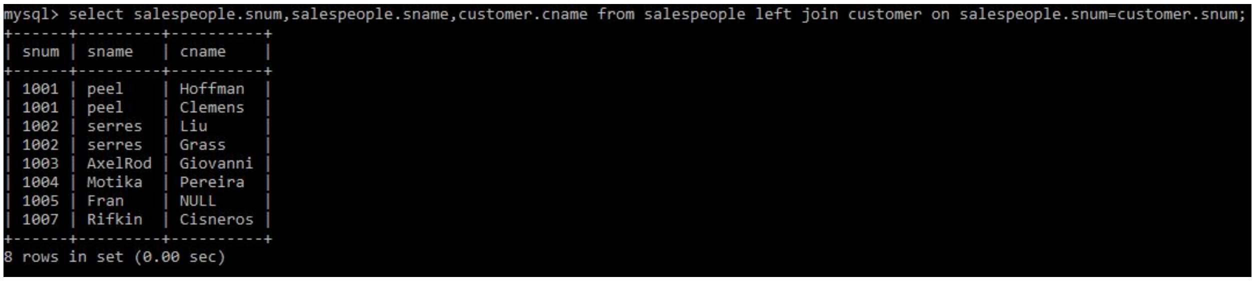
Ans:



```
mysql> select customer.cname,salespeople.sname from salespeople inner join customer on salespeople.snum=customer.snum;
+----------+---------+
| cname    | sname   |
+----------+---------+
| Hoffman  | peel    |
| Clemens  | peel    |
| Liu      | serres  |
| Grass    | serres  |
| Giovanni | AxelRod |
| Pereira  | Motika  |
| Cisneros | Rifkin  |
+----------+---------+
7 rows in set (0.00 sec)

mysql>
```

8. Find the names and numbers of all salespeople who had more than one customer.

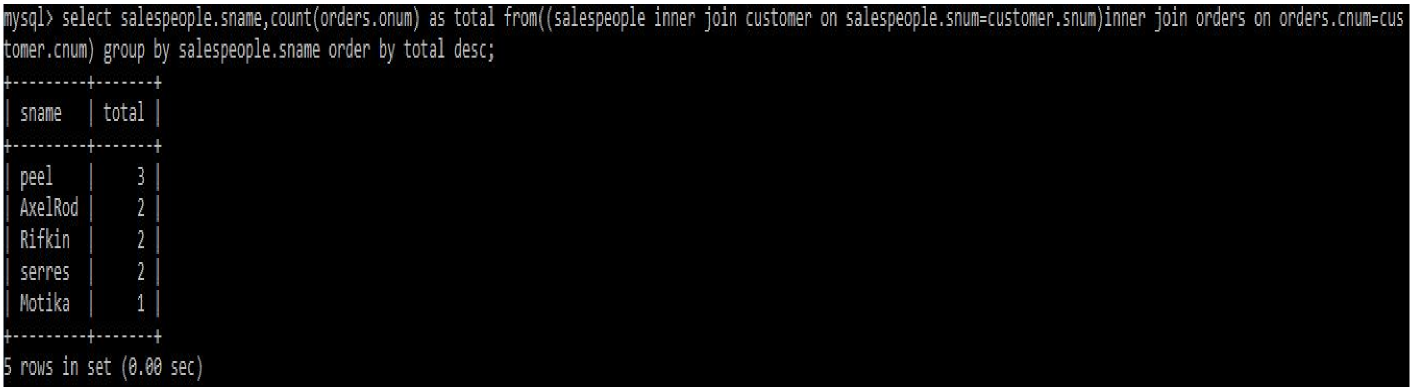
ans:



```
mysql> select salespeople.snum,salespeople.sname,customer.cname from salespeople left join customer on salespeople.snum=customer.snum;
+------+---------+----------+
| snum | sname   | cname    |
+------+---------+----------+
| 1001 | peel    | Hoffman  |
| 1001 | peel    | Clemens  |
| 1002 | serres  | Liu      |
| 1002 | serres  | Grass    |
| 1003 | AxelRod | Giovanni |
| 1004 | Motika  | Pereira  |
| 1005 | Fran    | NULL     |
| 1007 | Rifkin  | Cisneros |
+------+---------+----------+
8 rows in set (0.00 sec)
```

9. Count the orders of each of the salespeople and output the results in descending order.

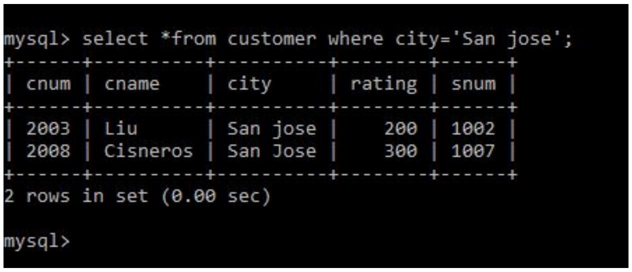
Ans:



```
mysql> select salespeople.sname,count(orders.onum) as total from((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on orders.cnum=cus
tomer.cnum) group by salespeople.sname order by total desc;
+---------+-------+
| sname   | total |
+---------+-------+
| peel    |     3 |
| AxelRod |     2 |
| Rifkin  |     2 |
| serres  |     2 |
| Motika  |     1 |
+---------+-------+
5 rows in set (0.00 sec)
```

10. List the Customer table if and only if one or more of the customers in the Customer table are located in San Jose.

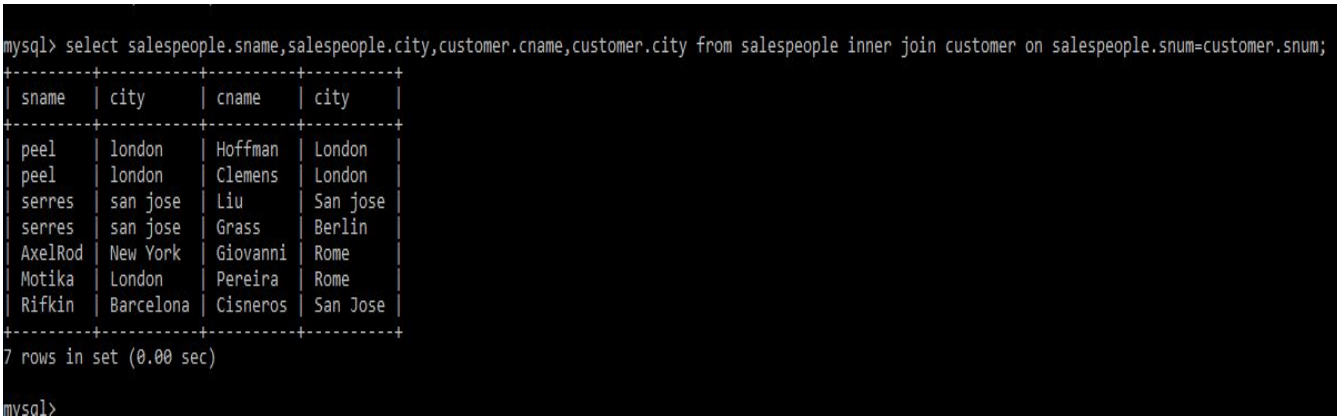
Ans:



```
mysql> select *from customer where city='San jose';
+------+----------+----------+--------+------+
| cnum | cname    | city     | rating | snum |
+------+----------+----------+--------+------+
| 2003 | Liu      | San jose |    200 | 1002 |
| 2008 | Cisneros | San Jose |    300 | 1007 |
+------+----------+----------+--------+------+
2 rows in set (0.00 sec)

mysql>
```

11. Match salespeople to customers according to what city they lived in.

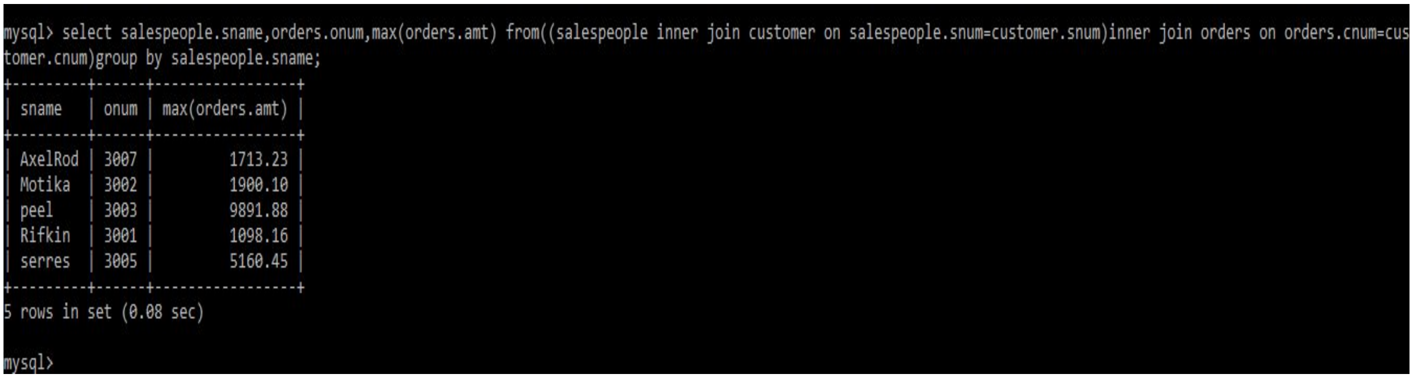
Ans:



```
mysql> select salespeople.sname,salespeople.city,customer.cname,customer.city from salespeople inner join customer on salespeople.snum=customer.snum;
+---------+-----------+----------+----------+
| sname   | city      | cname    | city     |
+---------+-----------+----------+----------+
| peel    | london    | Hoffman  | London   |
| peel    | london    | Clemens  | London   |
| serres  | san jose  | Liu      | San jose |
| serres  | san jose  | Grass    | Berlin   |
| AxelRod | New York  | Giovanni | Rome     |
| Motika  | London    | Pereira  | Rome     |
| Rifkin  | Barcelona | Cisneros | San Jose |
+---------+-----------+----------+----------+
7 rows in set (0.00 sec)

mysql>
```

12. Find the largest order taken by each salesperson.

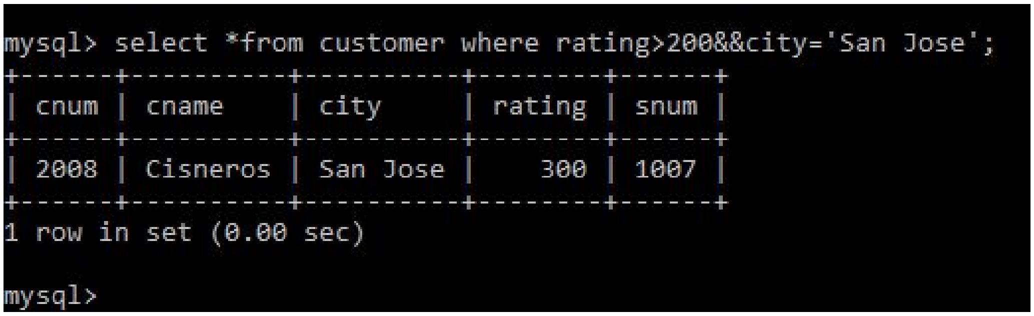
Ans:



```
mysql> select salespeople.sname,orders.onum,max(orders.amt) from((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on orders.cnum=cus
tomer.cnum)group by salespeople.sname;
+---------+------+----------------+
| sname   | onum | max(orders.amt) |
+---------+------+----------------+
| AxelRod | 3007 |        1713.23 |
| Motika  | 3002 |        1900.10 |
| peel    | 3003 |        9891.88 |
| Rifkin  | 3001 |        1098.16 |
| serres  | 3005 |        5160.45 |
+---------+------+----------------+
5 rows in set (0.08 sec)

mysql>
```

13. Find customers in San Jose who have a rating above 200.

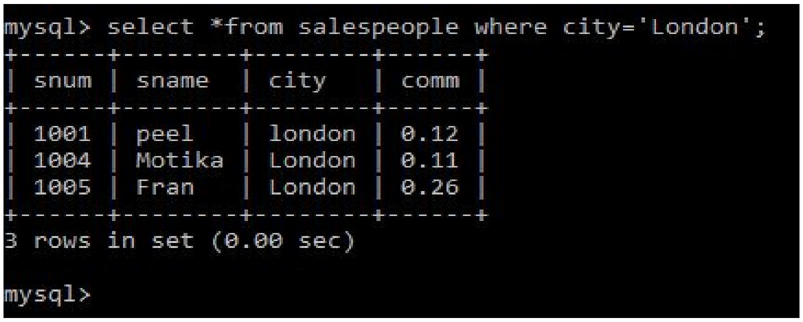
Ans:



```
mysql> select *from customer where rating>200&&city='San Jose';
+------+----------+----------+--------+------+
| cnum | cname    | city     | rating | snum |
+------+----------+----------+--------+------+
| 2008 | Cisneros | San Jose |    300 | 1007 |
+------+----------+----------+--------+------+
1 row in set (0.00 sec)

mysql>
```

14. List the names and commissions of all salespeople in London.

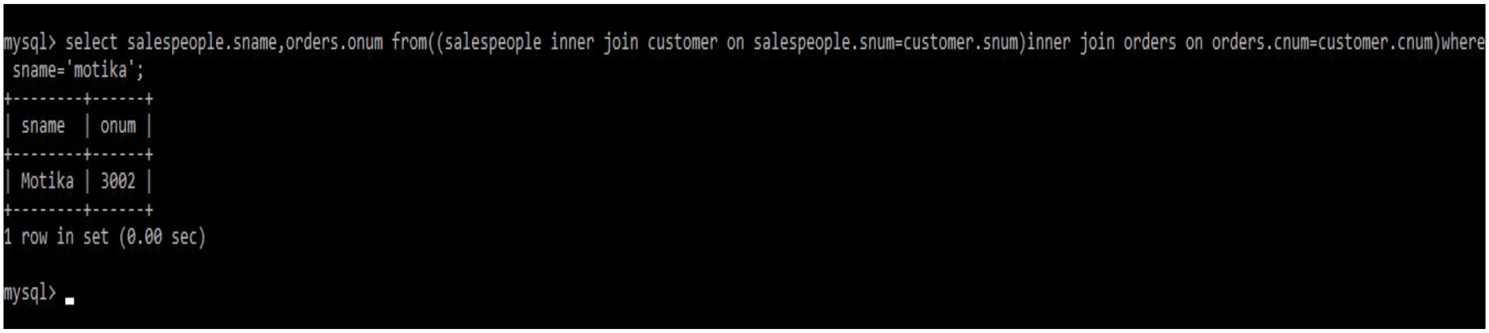
Ans:



```
mysql> select *from salespeople where city='London';
+------+--------+--------+------+
| snum | sname  | city   | comm |
+------+--------+--------+------+
| 1001 | peel   | london | 0.12 |
| 1004 | Motika | London | 0.11 |
| 1005 | Fran   | London | 0.26 |
+------+--------+--------+------+
3 rows in set (0.00 sec)

mysql>
```

15. List all the orders of salesperson Motika from the Orders table.

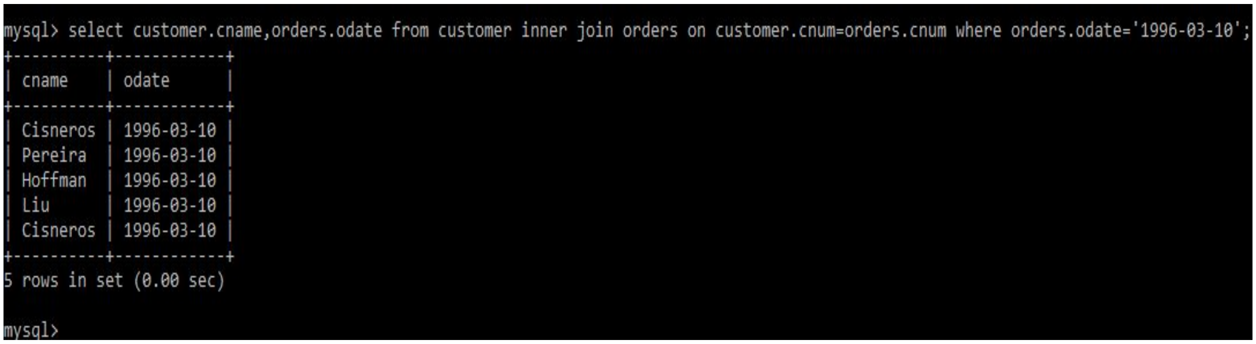
Ans:



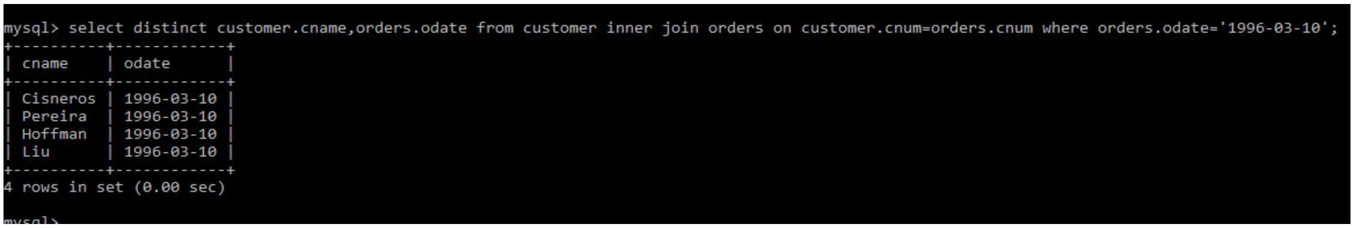
```
mysql> select salespeople.sname,orders.onum from((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on orders.cnum=customer.cnum)where
 sname='motika';
+--------+------+
| sname  | onum |
+--------+------+
| Motika | 3002 |
+--------+------+
1 row in set (0.00 sec)

mysql> 
```

16. Find all customers with orders on October 3.

Ans:



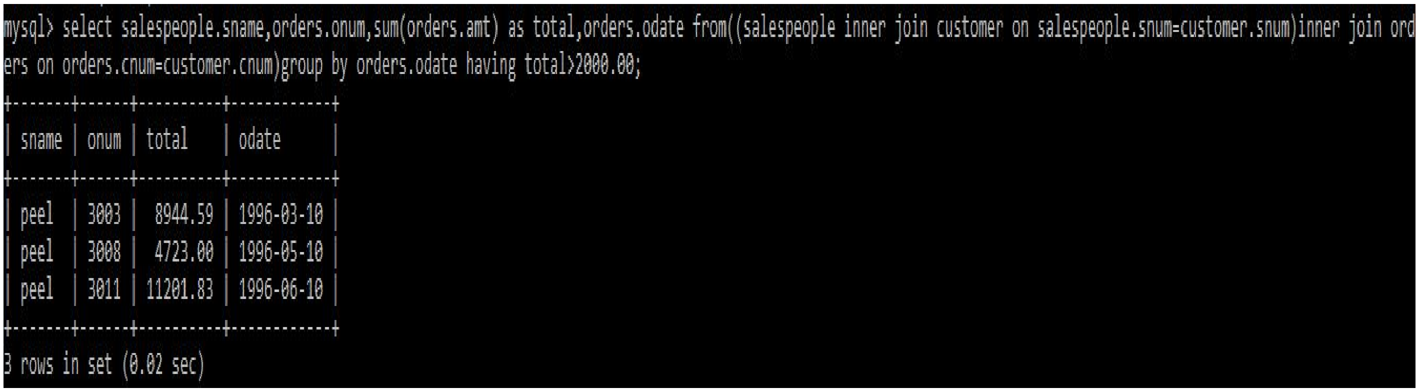
```
mysql> select customer.cname,orders.odate from customer inner join orders on customer.cnum=orders.cnum where orders.odate='1996-03-10';
+----------+------------+
| cname    | odate      |
+----------+------------+
| Cisneros | 1996-03-10 |
| Pereira  | 1996-03-10 |
| Hoffman  | 1996-03-10 |
| Liu      | 1996-03-10 |
| Cisneros | 1996-03-10 |
+----------+------------+
5 rows in set (0.00 sec)

mysql>
```



```
mysql> select distinct customer.cname,orders.odate from customer inner join orders on customer.cnum=orders.cnum where orders.odate='1996-03-10';
+----------+------------+
| cname    | odate      |
+----------+------------+
| Cisneros | 1996-03-10 |
| Pereira  | 1996-03-10 |
| Hoffman  | 1996-03-10 |
| Liu      | 1996-03-10 |
+----------+------------+
4 rows in set (0.00 sec)

mysql>
```

17. Give the sums of the amounts from the Orders table, grouped by date, eliminating all those dates where the SUM was not at least 2000.00 above the MAX amount.

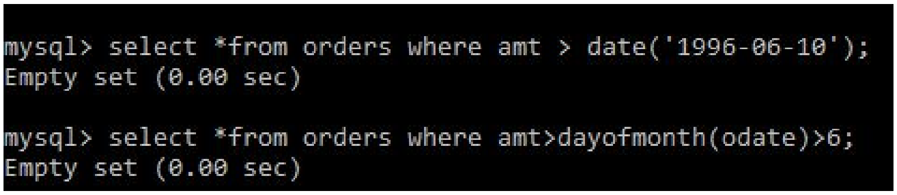
Ans:



```
mysql> select salespeople.sname,orders.onum,sum(orders.amt) as total,orders.odate from((salespeople inner join customer on salespeople.snum=customer.snum)inner join ord
ers on orders.cnum=customer.cnum)group by orders.odate having total>2000.00;
+-------+------+----------+------------+
| sname | onum | total    | odate      |
+-------+------+----------+------------+
| peel  | 3003 |  8944.59 | 1996-03-10 |
| peel  | 3008 |  4723.00 | 1996-05-10 |
| peel  | 3011 | 11201.83 | 1996-06-10 |
+-------+------+----------+------------+
3 rows in set (0.02 sec)
```

18. Select all orders that had amounts that were greater than at least one of the orders from October 6.

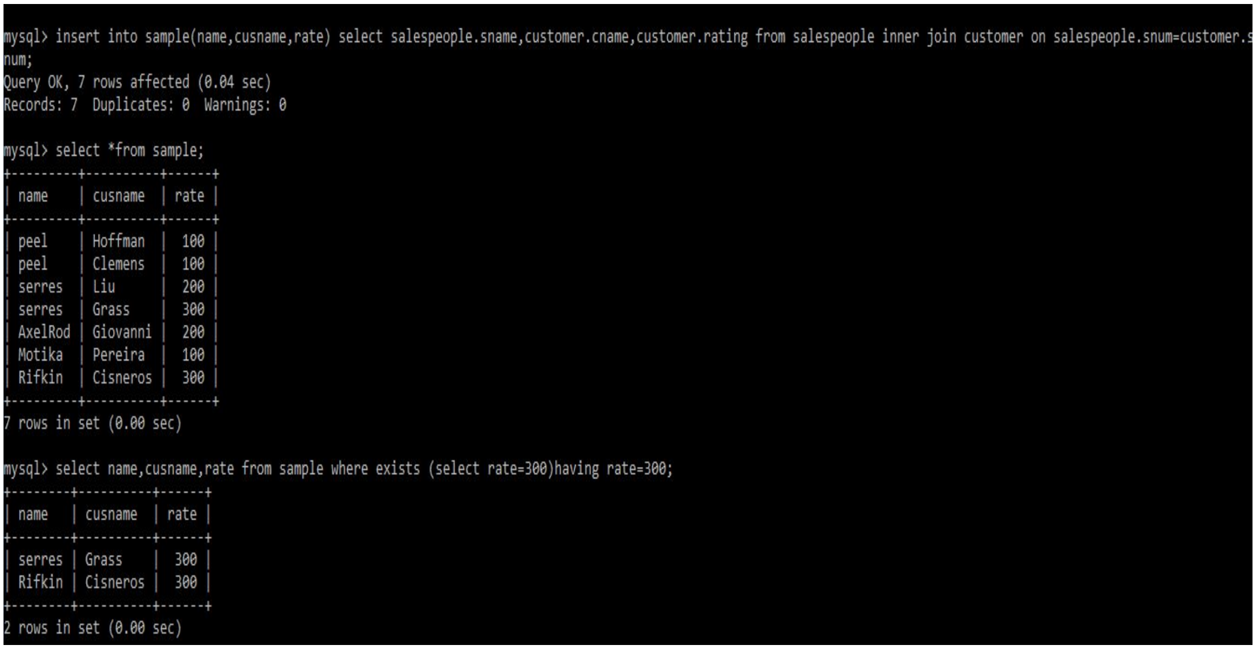
Ans:



```
mysql> select *from orders where amt > date('1996-06-10');
Empty set (0.00 sec)

mysql> select *from orders where amt>dayofmonth(odate)>6;
Empty set (0.00 sec)
```

19. Write a query that uses the EXISTS operator to extract all salespeople who have customers with a rating of 300.

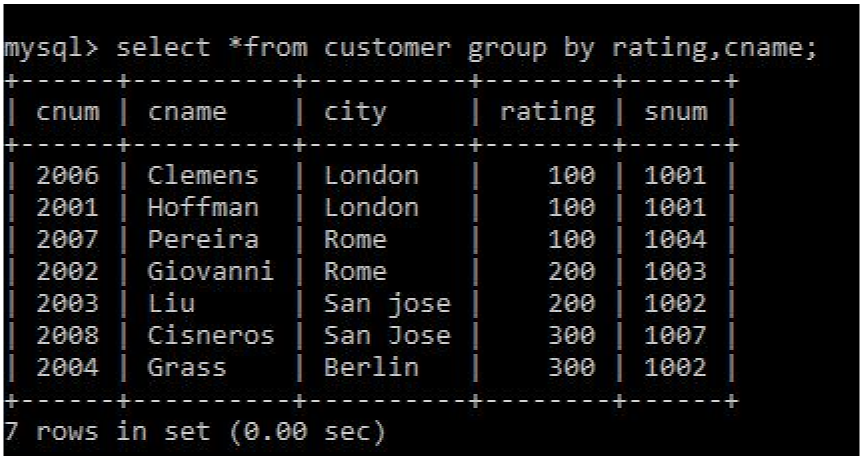
Ans:



```
mysql> insert into sample(name,cusname,rate) select salespeople.sname,customer.cname,customer.rating from salespeople inner join customer on salespeople.snum=customer.s
num;
Query OK, 7 rows affected (0.04 sec)
Records: 7  Duplicates: 0  Warnings: 0

mysql> select *from sample;
+---------+----------+------+
| name    | cusname  | rate |
+---------+----------+------+
| peel    | Hoffman  |  100 |
| peel    | Clemens  |  100 |
| serres  | Liu      |  200 |
| serres  | Grass    |  300 |
| AxelRod | Giovanni |  200 |
| Motika  | Pereira  |  100 |
| Rifkin  | Cisneros |  300 |
+---------+----------+------+
7 rows in set (0.00 sec)

mysql> select name,cusname,rate from sample where exists (select rate=300)having rate=300;
+--------+----------+------+
| name   | cusname  | rate |
+--------+----------+------+
| serres | Grass    |  300 |
| Rifkin | Cisneros |  300 |
+--------+----------+------+
2 rows in set (0.00 sec)
```

20. Find all pairs of customers having the same rating.

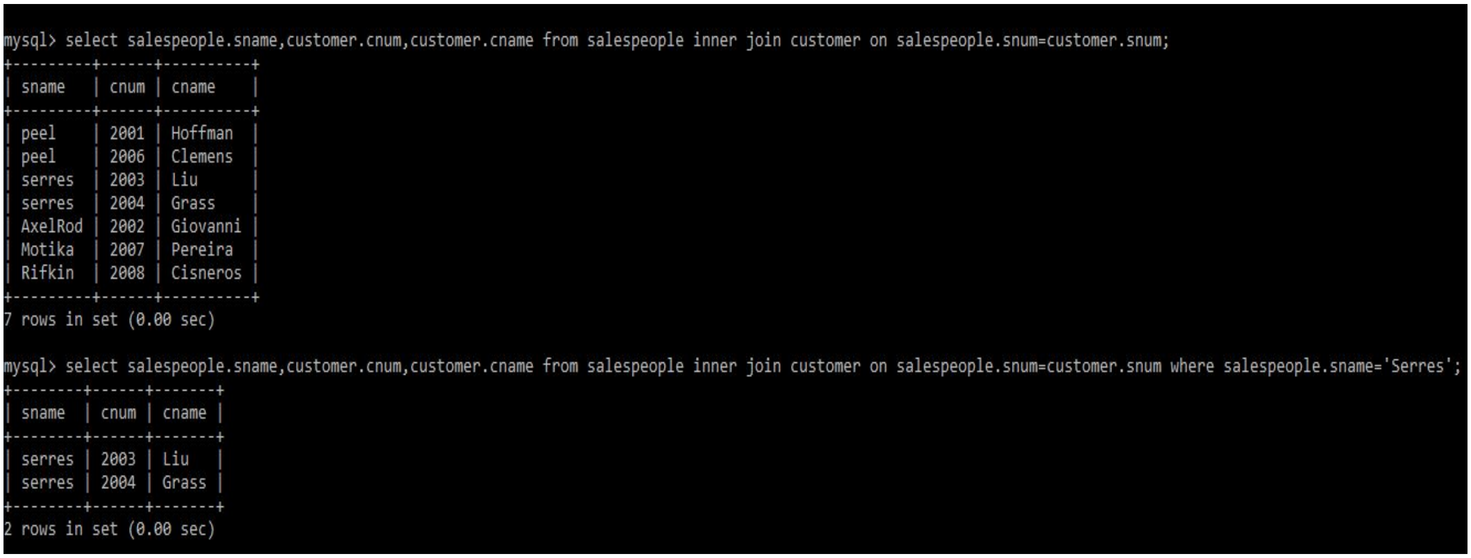
Ans:



```
mysql> select *from customer group by rating,cname;
+------+----------+----------+--------+------+
| cnum | cname    | city     | rating | snum |
+------+----------+----------+--------+------+
| 2006 | Clemens  | London   |    100 | 1001 |
| 2001 | Hoffman  | London   |    100 | 1001 |
| 2007 | Pereira  | Rome     |    100 | 1004 |
| 2002 | Giovanni | Rome     |    200 | 1003 |
| 2003 | Liu      | San jose |    200 | 1002 |
| 2008 | Cisneros | San Jose |    300 | 1007 |
| 2004 | Grass    | Berlin   |    300 | 1002 |
+------+----------+----------+--------+------+
7 rows in set (0.00 sec)
```

21. Find all customers whose CNUM is 1000 above the SNUM of Serres.

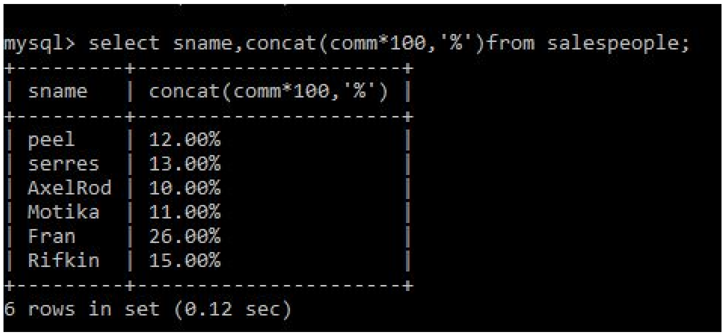
Ans:



```
mysql> select salespeople.sname,customer.cnum,customer.cname from salespeople inner join customer on salespeople.snum=customer.snum;
+---------+------+----------+
| sname   | cnum | cname    |
+---------+------+----------+
| peel    | 2001 | Hoffman  |
| peel    | 2006 | Clemens  |
| serres  | 2003 | Liu      |
| serres  | 2004 | Grass    |
| AxelRod | 2002 | Giovanni |
| Motika  | 2007 | Pereira  |
| Rifkin  | 2008 | Cisneros |
+---------+------+----------+
7 rows in set (0.00 sec)

mysql> select salespeople.sname,customer.cnum,customer.cname from salespeople inner join customer on salespeople.snum=customer.snum where salespeople.sname='Serres';
+--------+------+-------+
| sname  | cnum | cname |
+--------+------+-------+
| serres | 2003 | Liu   |
| serres | 2004 | Grass |
+--------+------+-------+
2 rows in set (0.00 sec)
```

22. Give the salespeople’s commissions as percentages instead of decimal numbers.

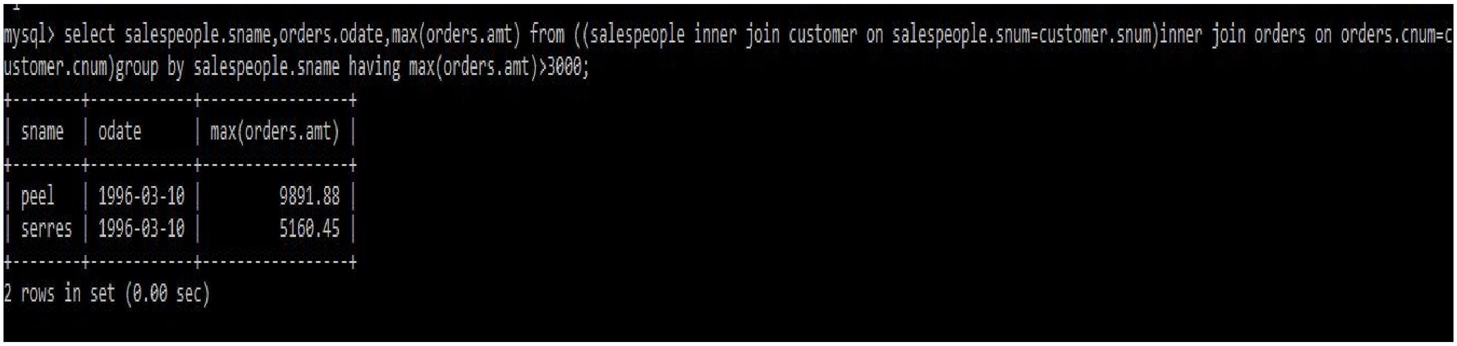
Ans:



```
mysql> select sname,concat(comm*100,'%')from salespeople;
+---------+----------------------+
| sname   | concat(comm*100,'%') |
+---------+----------------------+
| peel    | 12.00%               |
| serres  | 13.00%               |
| AxelRod | 10.00%               |
| Motika  | 11.00%               |
| Fran    | 26.00%               |
| Rifkin  | 15.00%               |
+---------+----------------------+
6 rows in set (0.12 sec)
```

23. Find the largest order taken by each salesperson on each date, eliminating those MAX orders which are less than $3000.00 in value.

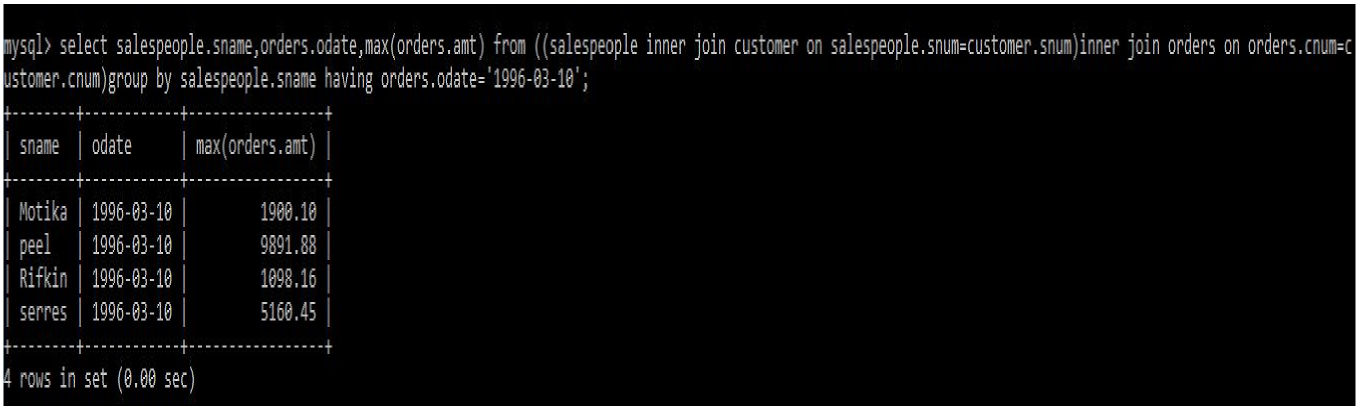
Ans:



```
mysql> select salespeople.sname,orders.odate,max(orders.amt) from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on orders.cnum=c
ustomer.cnum)group by salespeople.sname having max(orders.amt)>3000;
+--------+------------+-----------------+
| sname  | odate      | max(orders.amt) |
+--------+------------+-----------------+
| peel   | 1996-03-10 |         9891.88 |
| serres | 1996-03-10 |         5160.45 |
+--------+------------+-----------------+
2 rows in set (0.00 sec)
```

24. List the largest orders for October 3, for each salesperson.

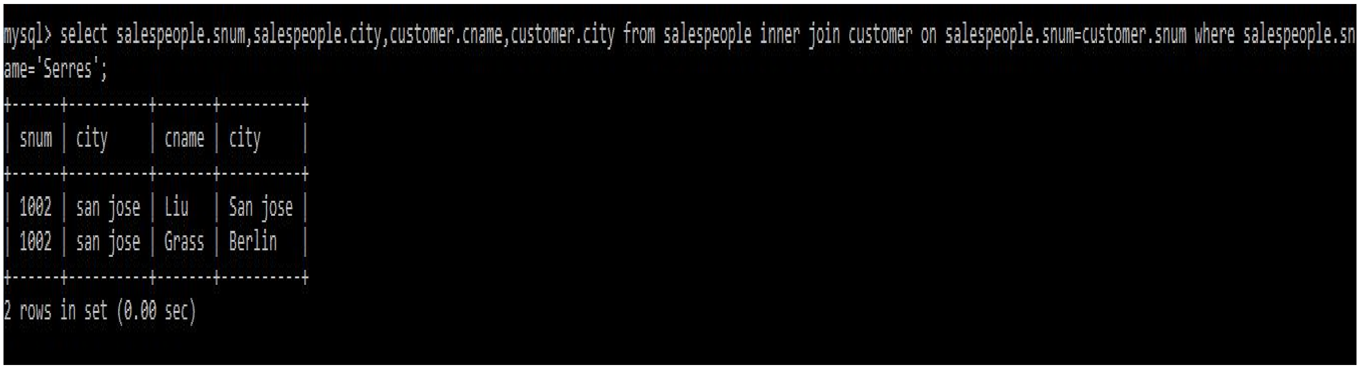
Ans:



```
mysql> select salespeople.sname,orders.odate,max(orders.amt) from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on orders.cnum=c
ustomer.cnum)group by salespeople.sname having orders.odate='1996-03-10';
+--------+-----------+----------------+
| sname  | odate     | max(orders.amt) |
+--------+-----------+----------------+
| Motika | 1996-03-10 |        1900.10 |
| peel   | 1996-03-10 |        9891.88 |
| Rifkin | 1996-03-10 |        1098.16 |
| serres | 1996-03-10 |        5160.45 |
+--------+-----------+----------------+
4 rows in set (0.00 sec)
```

25. Find all customers located in cities where Serres (SNUM 1002) has customers.

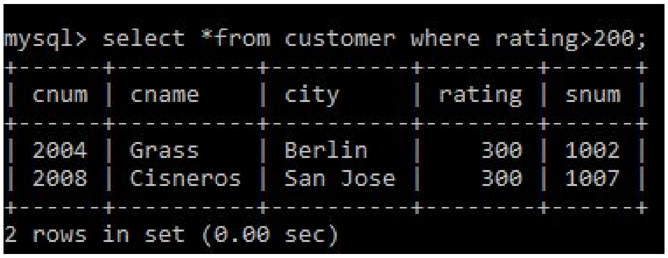
Ans:



```
mysql> select salespeople.snum,salespeople.city,customer.cname,customer.city from salespeople inner join customer on salespeople.snum=customer.snum where salespeople.sn
ame='Serres';
+------+---------+-------+----------+
| snum | city     | cname | city     |
+------+---------+-------+----------+
| 1002 | san jose | Liu   | San jose |
| 1002 | san jose | Grass | Berlin   |
+------+---------+-------+----------+
2 rows in set (0.00 sec)
```

26. Select all customers with a rating above 200.00.

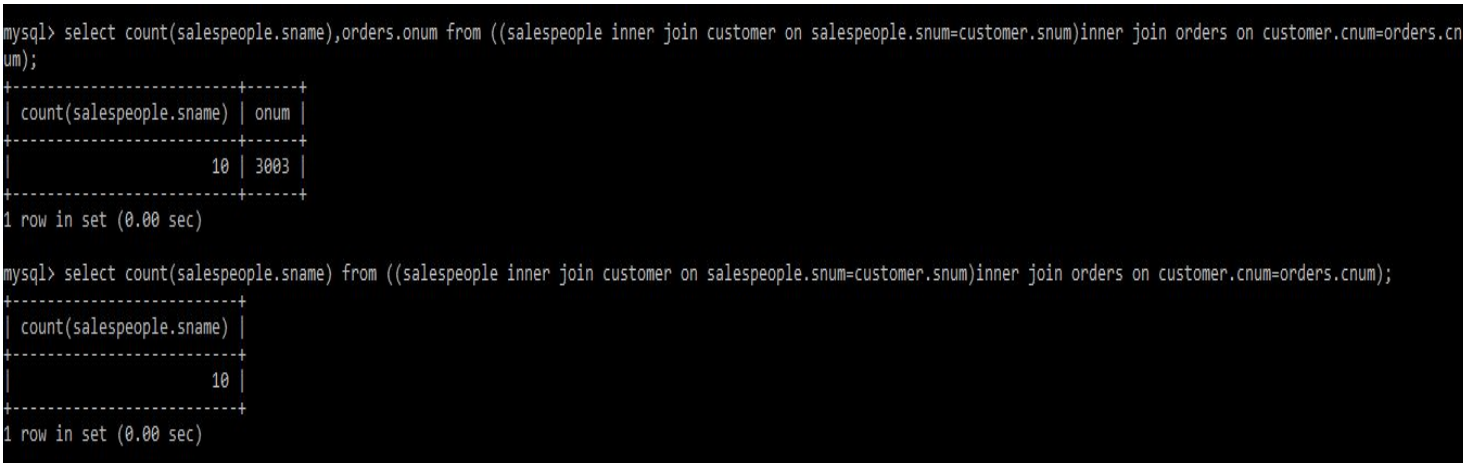
Ans:



```
mysql> select *from customer where rating>200;
+------+----------+----------+--------+------+
| cnum | cname    | city     | rating | snum |
+------+----------+----------+--------+------+
| 2004 | Grass    | Berlin   |    300 | 1002 |
| 2008 | Cisneros | San Jose |    300 | 1007 |
+------+----------+----------+--------+------+
2 rows in set (0.00 sec)
```

27. Count the number of salespeople currently listing orders in the Orders table.

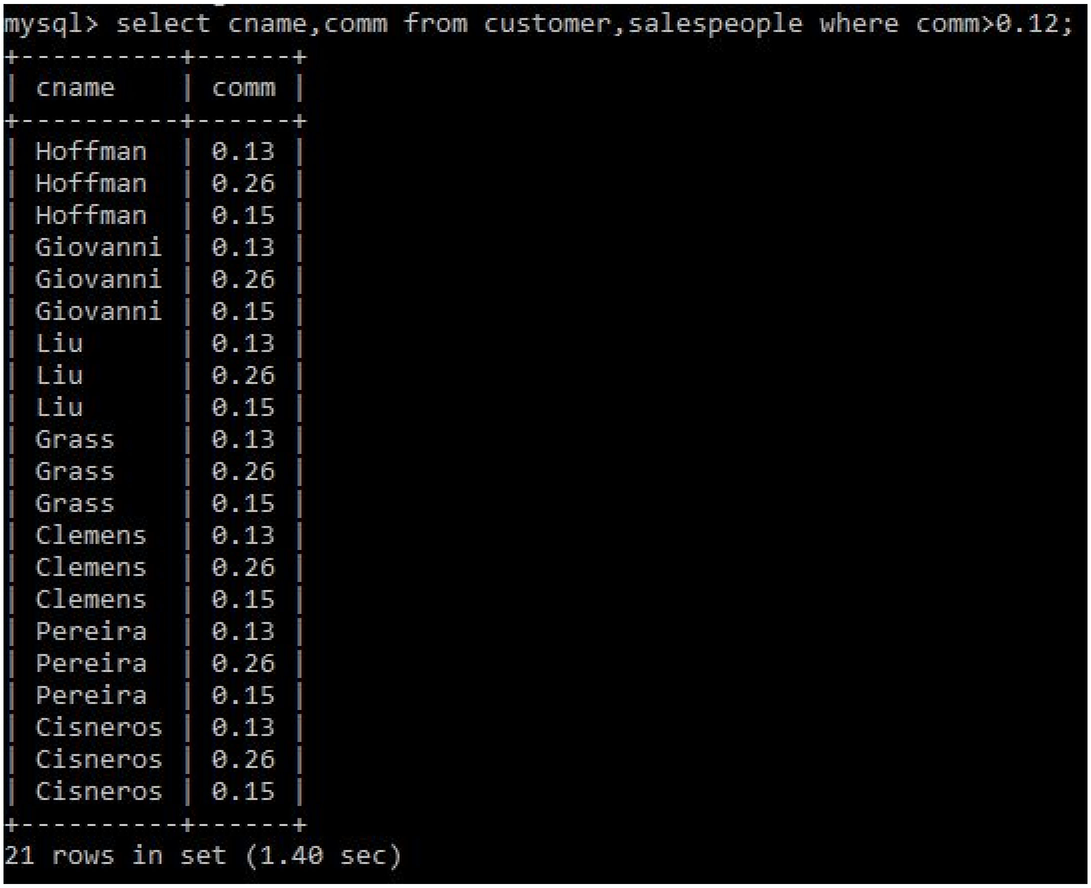
Ans:



```
mysql> select count(salespeople.sname),orders.onum from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cn
um);
+-------------------------+------+
| count(salespeople.sname) | onum |
+-------------------------+------+
|                      10 | 3003 |
+-------------------------+------+
1 row in set (0.00 sec)

mysql> select count(salespeople.sname) from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum);
+-------------------------+
| count(salespeople.sname) |
+-------------------------+
|                      10 |
+-------------------------+
1 row in set (0.00 sec)
```

28. Write a query that produces all customers serviced by salespeople with a commission above 12%. Output the customer's name and the salesperson's rate of commission.

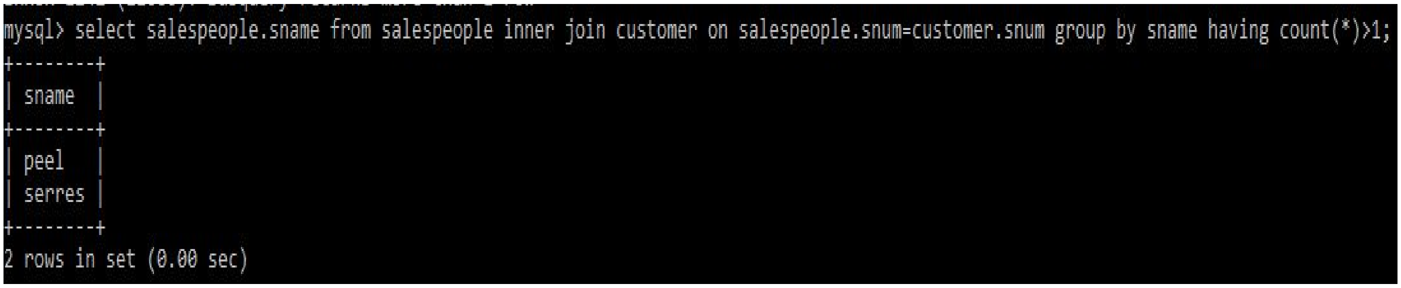
Ans;



```
mysql> select cname,comm from customer,salespeople where comm>0.12;
+----------+------+
| cname    | comm |
+----------+------+
| Hoffman  | 0.13 |
| Hoffman  | 0.26 |
| Hoffman  | 0.15 |
| Giovanni | 0.13 |
| Giovanni | 0.26 |
| Giovanni | 0.15 |
| Liu      | 0.13 |
| Liu      | 0.26 |
| Liu      | 0.15 |
| Grass    | 0.13 |
| Grass    | 0.26 |
| Grass    | 0.15 |
| Clemens  | 0.13 |
| Clemens  | 0.26 |
| Clemens  | 0.15 |
| Pereira  | 0.13 |
| Pereira  | 0.26 |
| Pereira  | 0.15 |
| Cisneros | 0.13 |
| Cisneros | 0.26 |
| Cisneros | 0.15 |
+----------+------+
21 rows in set (1.40 sec)
```

29. Find salespeople who have multiple customers.

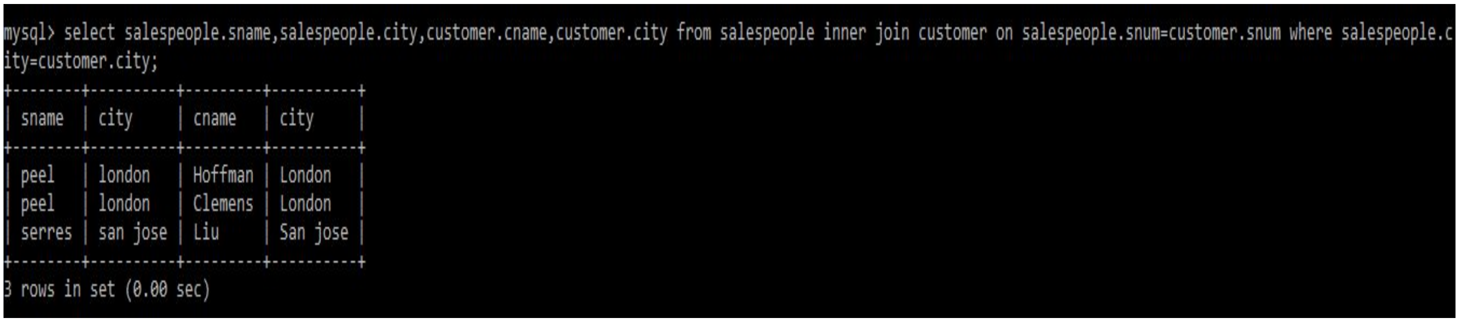
Ans:



```
mysql> select salespeople.sname from salespeople inner join customer on salespeople.snum=customer.snum group by sname having count(*)>1;
+--------+
| sname  |
+--------+
| peel   |
| serres |
+--------+
2 rows in set (0.00 sec)
```

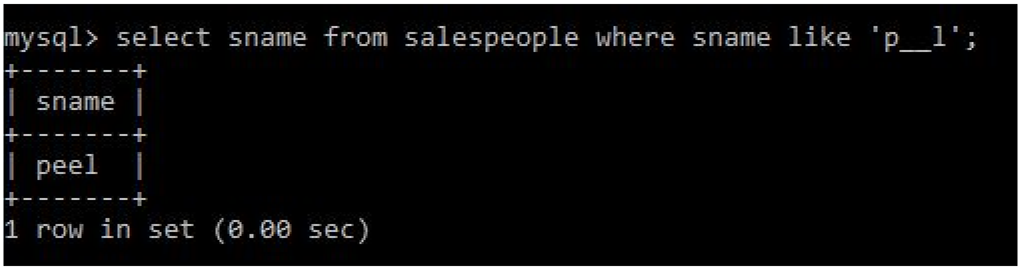
30. Find salespeople with customers located in their city.

Ans:



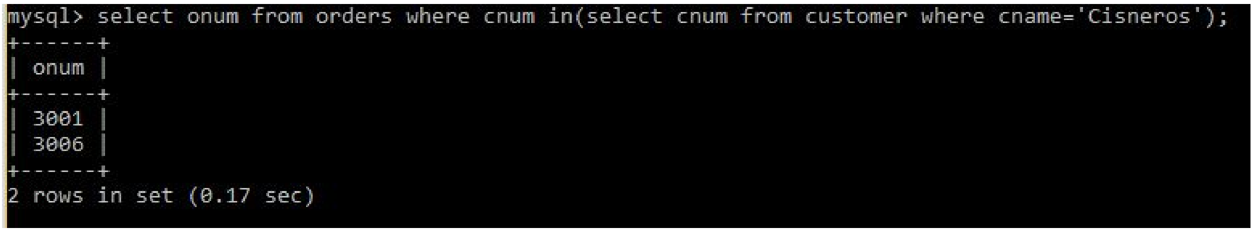
```
mysql> select salespeople.sname,salespeople.city,customer.cname,customer.city from salespeople inner join customer on salespeople.snum=customer.snum where salespeople.c
ity=customer.city;
+-------+---------+--------+---------+
| sname | city    | cname  | city    |
+-------+---------+--------+---------+
| peel  | london  | Hoffman | London  |
| peel  | london  | Clemens | London  |
| serres | san jose | Liu     | San jose |
+-------+---------+--------+---------+
3 rows in set (0.00 sec)
```

31. Find all salespeople whose name starts with ‘P’ and the fourth character is ‘l’.



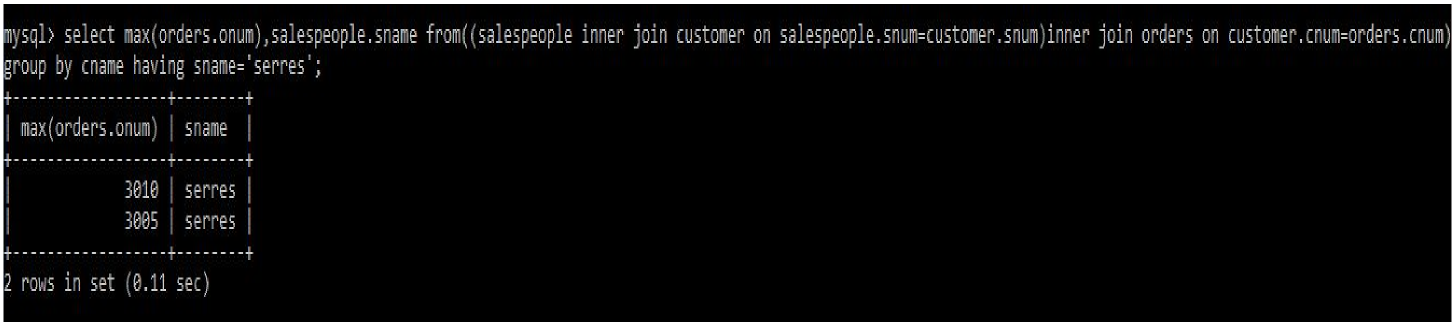
```
mysql> select sname from salespeople where sname like 'p__l';
+-------+
| sname |
+-------+
| peel  |
+-------+
1 row in set (0.00 sec)
```

32. Write a query that uses a subquery to obtain all orders for the customer named Cisneros. Assume you do not know his customer number.



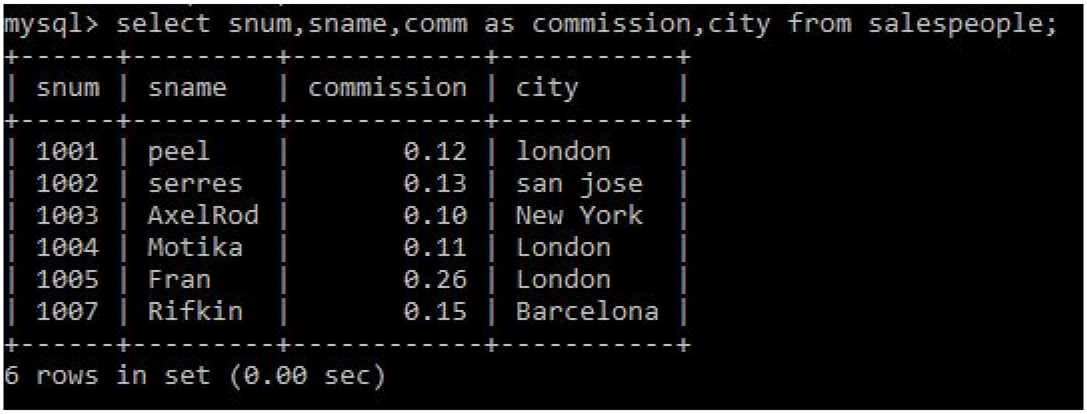
```
mysql> select onum from orders where cnum in(select cnum from customer where cname='Cisneros');
+------+
| onum |
+------+
| 3001 |
| 3006 |
+------+
2 rows in set (0.17 sec)
```

33. Find the largest orders for Serres and Rifkin.



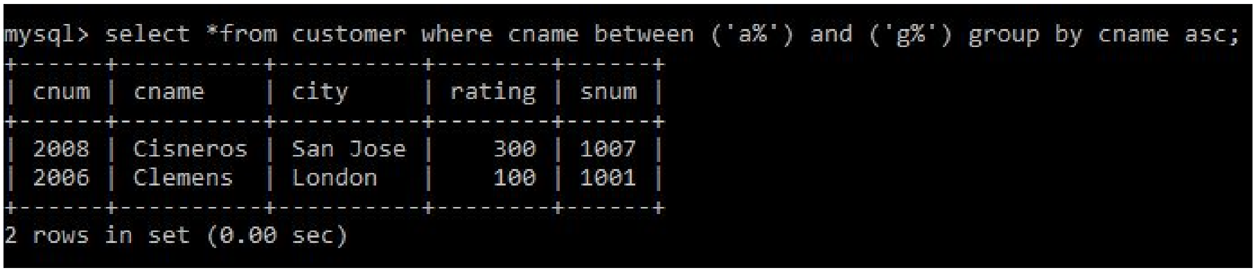
```
mysql> select max(orders.onum),salespeople.sname from((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum)
group by cname having sname='serres';
+-----------------+--------+
| max(orders.onum) | sname  |
+-----------------+--------+
|            3010 | serres |
|            3005 | serres |
+-----------------+--------+
2 rows in set (0.11 sec)
```

34. Extract the Salespeople table in the following order : SNUM, SNAME, COMMISSION, CITY.



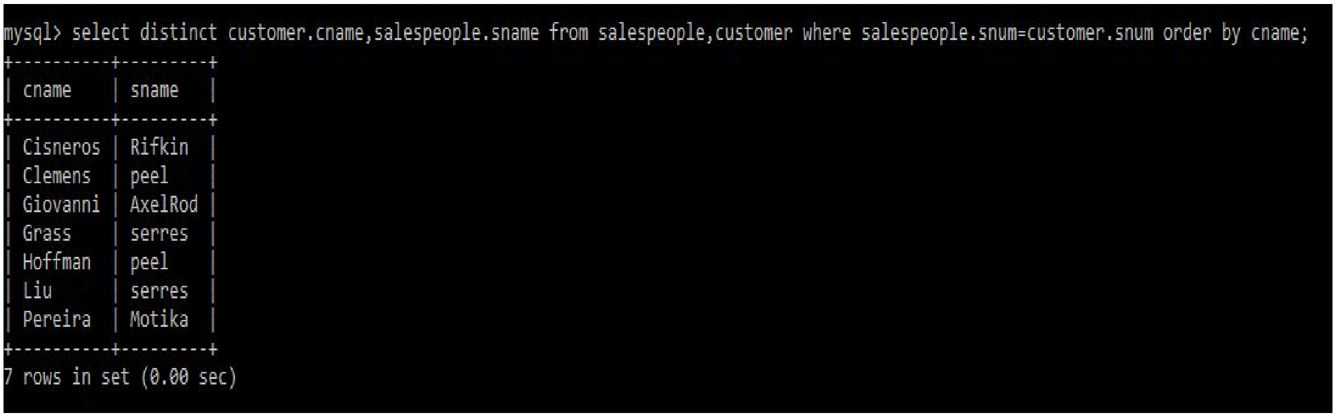
```
mysql> select snum,sname,comm as commission,city from salespeople;
+------+---------+------------+-----------+
| snum | sname   | commission | city      |
+------+---------+------------+-----------+
| 1001 | peel    |       0.12 | london    |
| 1002 | serres  |       0.13 | san jose  |
| 1003 | AxelRod |       0.10 | New York  |
| 1004 | Motika  |       0.11 | London    |
| 1005 | Fran    |       0.26 | London    |
| 1007 | Rifkin  |       0.15 | Barcelona |
+------+---------+------------+-----------+
6 rows in set (0.00 sec)
```

35. Select all customers whose names fall in between ‘A’ and ‘G’ alphabetical range.



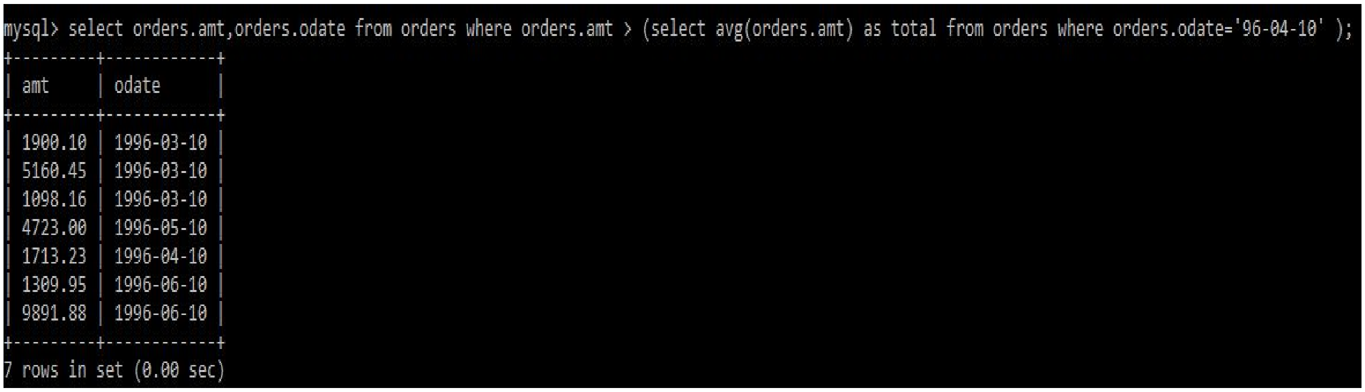
```
mysql> select *from customer where cname between ('a%') and ('g%') group by cname asc;
+------+----------+----------+--------+------+
| cnum | cname    | city     | rating | snum |
+------+----------+----------+--------+------+
| 2008 | Cisneros | San Jose |    300 | 1007 |
| 2006 | Clemens  | London   |    100 | 1001 |
+------+----------+----------+--------+------+
2 rows in set (0.00 sec)
```

36. Select all the possible combinations of customers that you can assign.



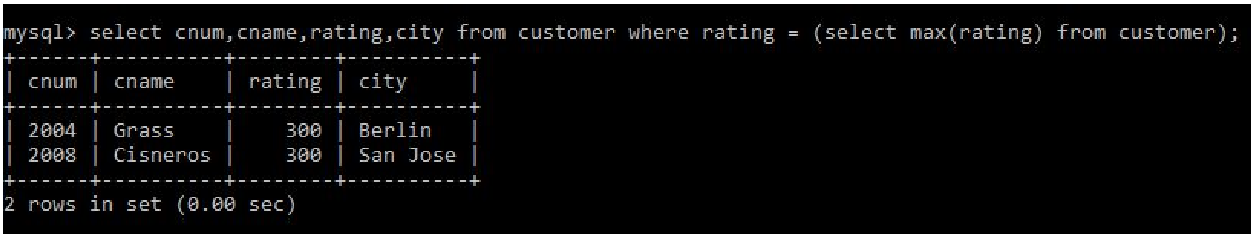
```
mysql> select distinct customer.cname,salespeople.sname from salespeople,customer where salespeople.snum=customer.snum order by cname;
+----------+---------+
| cname    | sname   |
+----------+---------+
| Cisneros | Rifkin  |
| Clemens  | peel    |
| Giovanni | AxelRod |
| Grass    | serres  |
| Hoffman  | peel    |
| Liu      | serres  |
| Pereira  | Motika  |
+----------+---------+
7 rows in set (0.00 sec)
```

37. Select all orders that are greater than the average for October 4.



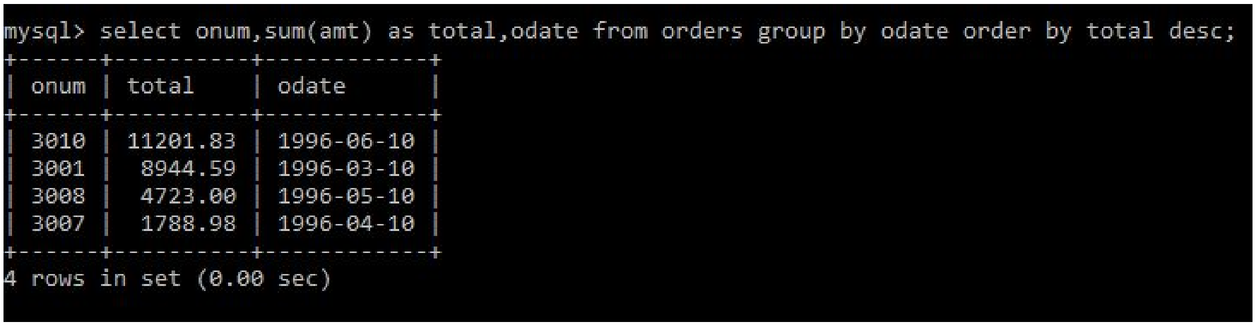
```
mysql> select orders.amt,orders.odate from orders where orders.amt > (select avg(orders.amt) as total from orders where orders.odate='96-04-10' );
+---------+------------+
| amt     | odate      |
+---------+------------+
| 1900.10 | 1996-03-10 |
| 5160.45 | 1996-03-10 |
| 1098.16 | 1996-03-10 |
| 4723.00 | 1996-05-10 |
| 1713.23 | 1996-04-10 |
| 1309.95 | 1996-06-10 |
| 9891.88 | 1996-06-10 |
+---------+------------+
7 rows in set (0.00 sec)
```

38. Write a select command using a corelated subquery that selects the names and numbers of all customers with ratings equal to the maximum for their city.



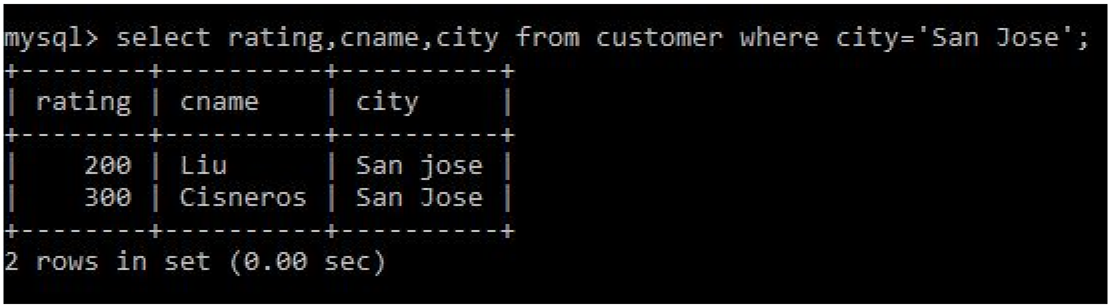
```
mysql> select cnum,cname,rating,city from customer where rating = (select max(rating) from customer);
+------+----------+--------+----------+
| cnum | cname    | rating | city     |
+------+----------+--------+----------+
| 2004 | Grass    |    300 | Berlin   |
| 2008 | Cisneros |    300 | San Jose |
+------+----------+--------+----------+
2 rows in set (0.00 sec)
```

39. Write a query that totals the orders for each day and places the results in descending order.



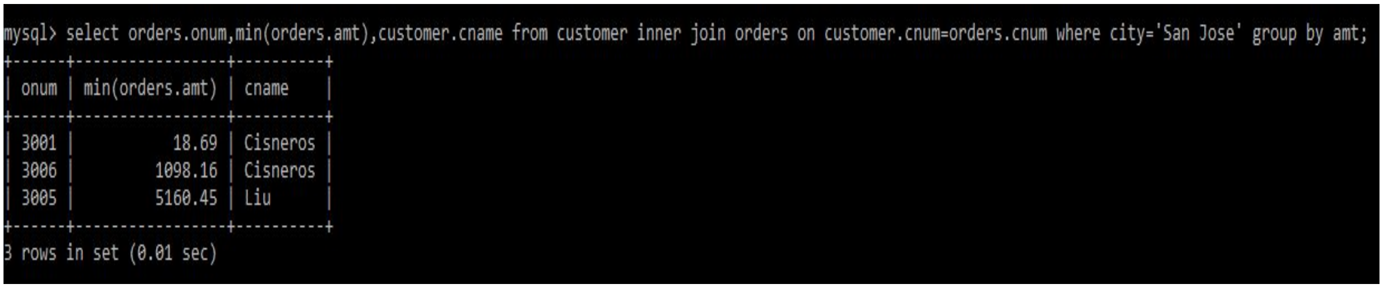
```
mysql> select onum,sum(amt) as total,odate from orders group by odate order by total desc;
+------+----------+------------+
| onum | total    | odate      |
+------+----------+------------+
| 3010 | 11201.83 | 1996-06-10 |
| 3001 |  8944.59 | 1996-03-10 |
| 3008 |  4723.00 | 1996-05-10 |
| 3007 |  1788.98 | 1996-04-10 |
+------+----------+------------+
4 rows in set (0.00 sec)
```

40. Write a select command that produces the rating followed by the name of each customer in San Jose.



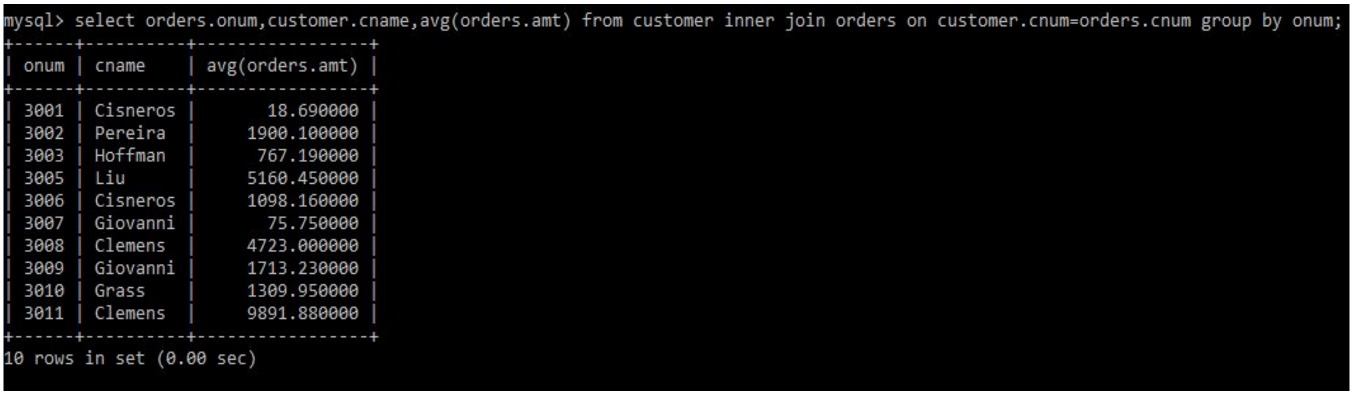
```
mysql> select rating,cname,city from customer where city='San Jose';
+--------+----------+----------+
| rating | cname    | city     |
+--------+----------+----------+
|    200 | Liu      | San jose |
|    300 | Cisneros | San Jose |
+--------+----------+----------+
2 rows in set (0.00 sec)
```

41. Find all orders with amounts smaller than any amount for a customer in San Jose.



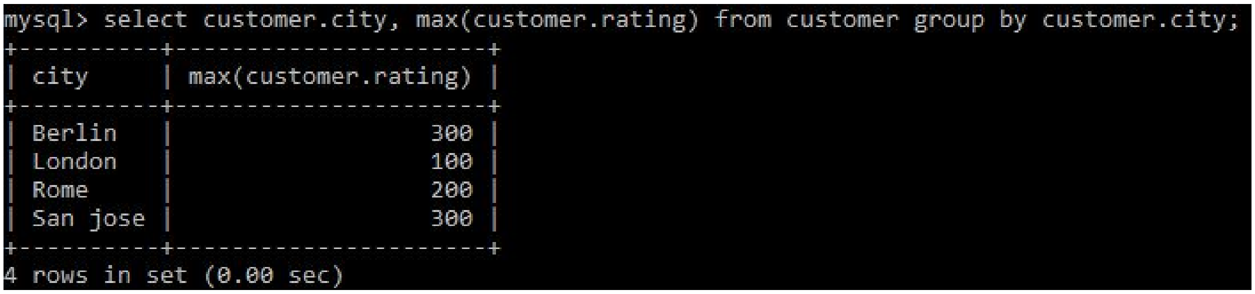
```
mysql> select orders.onum,min(orders.amt),customer.cname from customer inner join orders on customer.cnum=orders.cnum where city='San Jose' group by amt;
+------+----------------+----------+
| onum | min(orders.amt) | cname    |
+------+----------------+----------+
| 3001 |          18.69 | Cisneros |
| 3006 |        1098.16 | Cisneros |
| 3005 |        5160.45 | Liu      |
+------+----------------+----------+
3 rows in set (0.01 sec)
```

42. Find all orders with above average amounts for their customers.



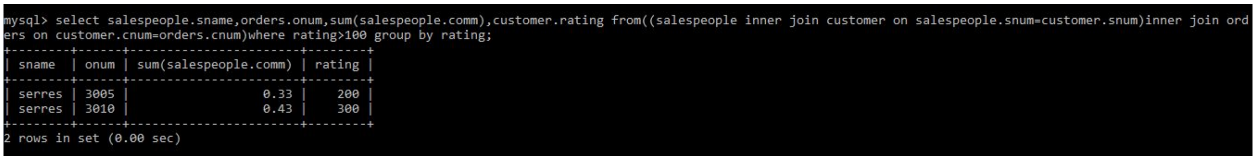
```
mysql> select orders.onum,customer.cname,avg(orders.amt) from customer inner join orders on customer.cnum=orders.cnum group by onum;
+------+----------+-----------------+
| onum | cname    | avg(orders.amt) |
+------+----------+-----------------+
| 3001 | Cisneros |       18.690000 |
| 3002 | Pereira  |     1900.100000 |
| 3003 | Hoffman  |      767.190000 |
| 3005 | Liu      |     5160.450000 |
| 3006 | Cisneros |     1098.160000 |
| 3007 | Giovanni |       75.750000 |
| 3008 | Clemens  |     4723.000000 |
| 3009 | Giovanni |     1713.230000 |
| 3010 | Grass    |     1309.950000 |
| 3011 | Clemens  |     9891.880000 |
+------+----------+-----------------+
10 rows in set (0.00 sec)
```

43. Write a query that selects the highest rating in each city.



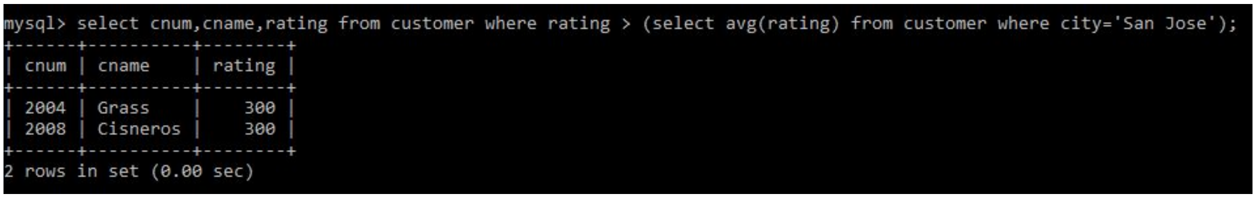
```
mysql> select customer.city, max(customer.rating) from customer group by customer.city;
+----------+----------------------+
| city     | max(customer.rating) |
+----------+----------------------+
| Berlin   |                  300 |
| London   |                  100 |
| Rome     |                  200 |
| San jose |                  300 |
+----------+----------------------+
4 rows in set (0.00 sec)
```

44. Write a query that calculates the amount of the salesperson’s commission on each order by a customer with a rating above 100.00.



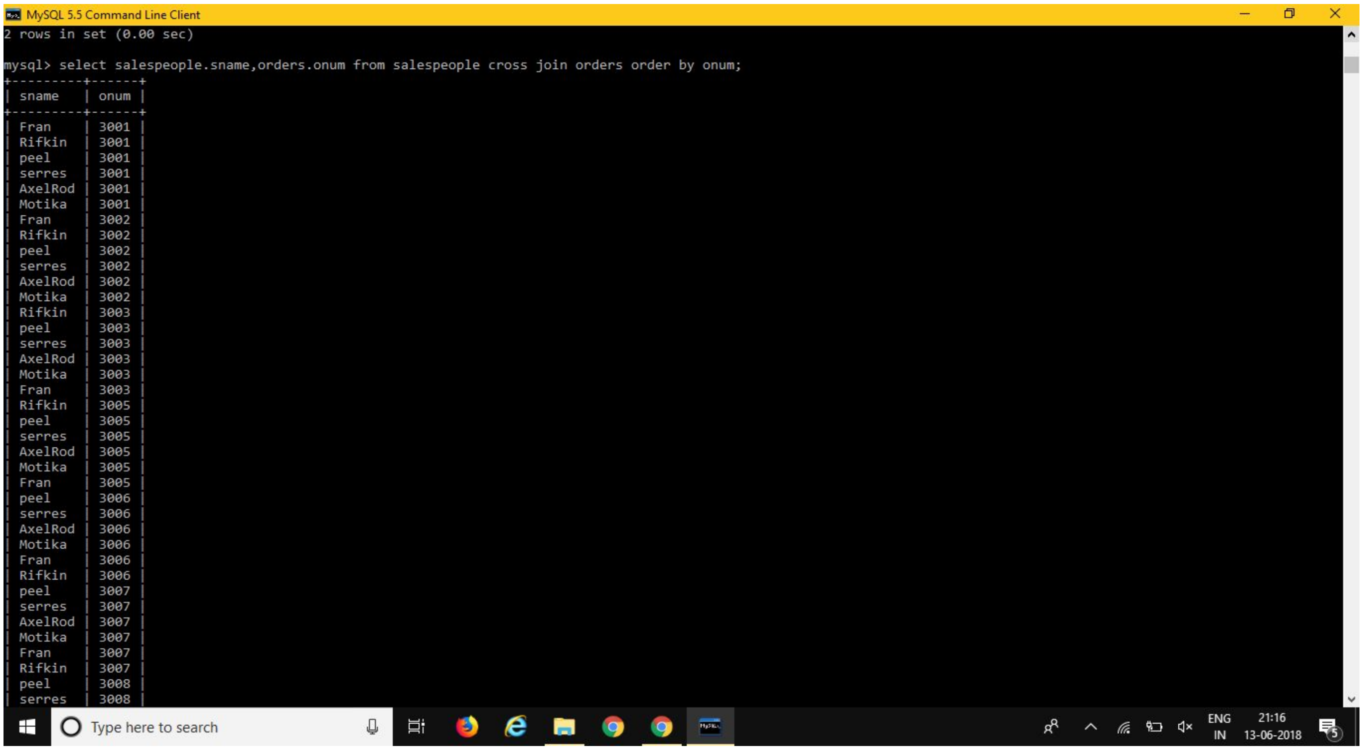
```
mysql> select salespeople.sname,orders.onum,sum(salespeople.comm),customer.rating from((salespeople inner join customer on salespeople.snum=customer.snum)inner join ord
ers on customer.cnum=orders.cnum)where rating>100 group by rating;
+--------+------+-----------------------+--------+
| sname  | onum | sum(salespeople.comm) | rating |
+--------+------+-----------------------+--------+
| serres | 3005 |                  0.33 |    200 |
| serres | 3010 |                  0.43 |    300 |
+--------+------+-----------------------+--------+
2 rows in set (0.00 sec)
```

45. Count the customers with ratings above San Jose’s average.

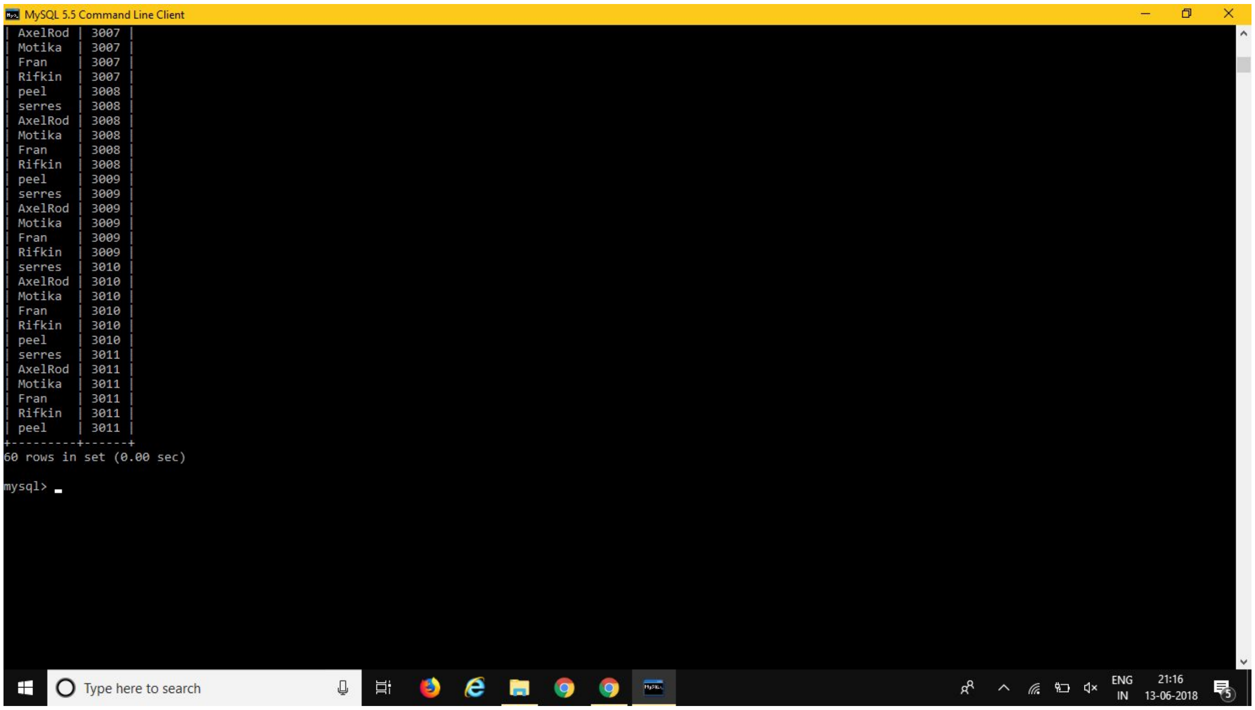


```
mysql> select cnum,cname,rating from customer where rating > (select avg(rating) from customer where city='San Jose');
+------+----------+--------+
| cnum | cname    | rating |
+------+----------+--------+
| 2004 | Grass    |    300 |
| 2008 | Cisneros |    300 |
+------+----------+--------+
2 rows in set (0.00 sec)
```

46. Write a query that produces all pairs of salespeople with themselves as well as duplicate rows with the order reversed.

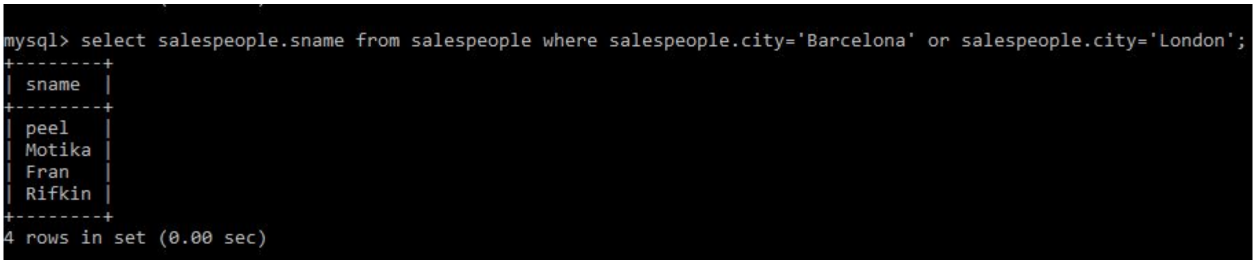


```
2 rows in set (0.00 sec)

mysql> select salespeople.sname,orders.onum from salespeople cross join orders order by onum;
+---------+------+
| sname   | onum |
+---------+------+
| Fran    | 3001 |
| Rifkin  | 3001 |
| peel    | 3001 |
| serres  | 3001 |
| AxelRod | 3001 |
| Motika  | 3001 |
| Fran    | 3002 |
| Rifkin  | 3002 |
| peel    | 3002 |
| serres  | 3002 |
| AxelRod | 3002 |
| Motika  | 3002 |
| Rifkin  | 3003 |
| peel    | 3003 |
| serres  | 3003 |
| AxelRod | 3003 |
| Motika  | 3003 |
| Fran    | 3003 |
| Rifkin  | 3005 |
| peel    | 3005 |
| serres  | 3005 |
| AxelRod | 3005 |
| Motika  | 3005 |
| Fran    | 3005 |
| peel    | 3006 |
| serres  | 3006 |
| AxelRod | 3006 |
| Motika  | 3006 |
| Fran    | 3006 |
| Rifkin  | 3006 |
| peel    | 3007 |
| serres  | 3007 |
| AxelRod | 3007 |
| Motika  | 3007 |
| Fran    | 3007 |
| Rifkin  | 3007 |
| peel    | 3008 |
| serres  | 3008 |
```



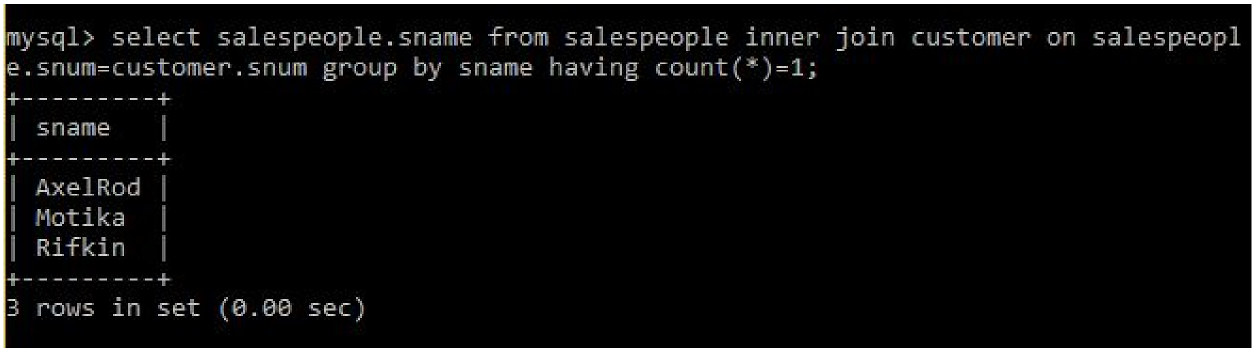
```
| AxelRod | 3007 |
| Motika  | 3007 |
| Fran    | 3007 |
| Rifkin  | 3007 |
| peel    | 3008 |
| serres  | 3008 |
| AxelRod | 3008 |
| Motika  | 3008 |
| Fran    | 3008 |
| Rifkin  | 3008 |
| peel    | 3009 |
| serres  | 3009 |
| AxelRod | 3009 |
| Motika  | 3009 |
| Fran    | 3009 |
| Rifkin  | 3009 |
| serres  | 3010 |
| AxelRod | 3010 |
| Motika  | 3010 |
| Fran    | 3010 |
| Rifkin  | 3010 |
| peel    | 3010 |
| serres  | 3011 |
| AxelRod | 3011 |
| Motika  | 3011 |
| Fran    | 3011 |
| Rifkin  | 3011 |
| peel    | 3011 |
+---------+------+
60 rows in set (0.00 sec)

mysql> _
```

47. Find all salespeople that are located in either Barcelona or London.



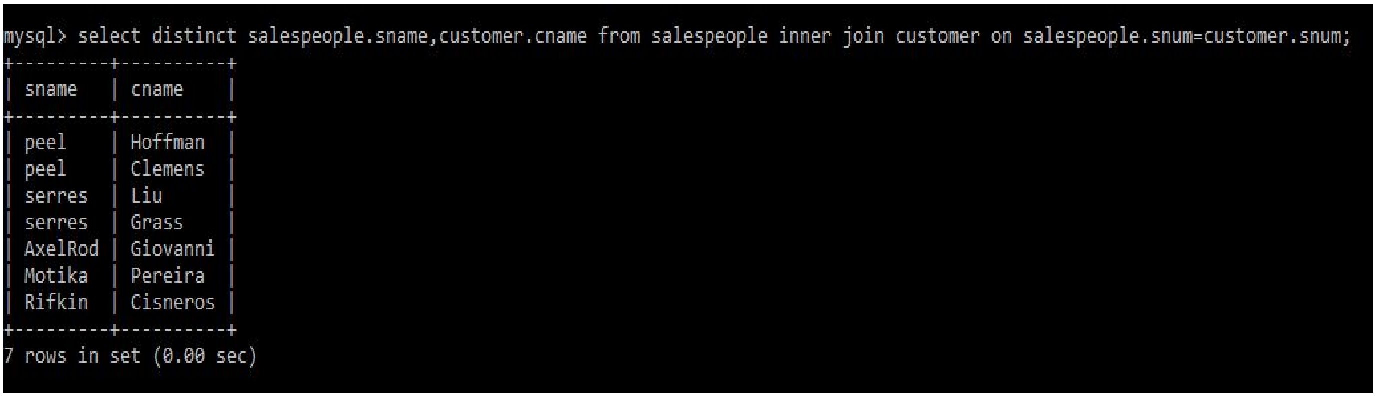
```
mysql> select salespeople.sname from salespeople where salespeople.city='Barcelona' or salespeople.city='London';
+--------+
| sname  |
+--------+
| peel   |
| Motika |
| Fran   |
| Rifkin |
+--------+
4 rows in set (0.00 sec)
```

48. Find all salespeople with only one customer.



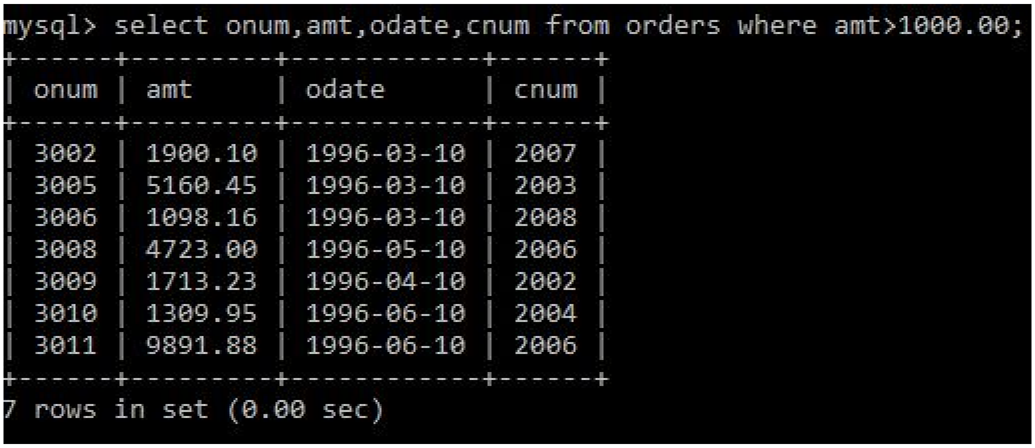
```
mysql> select salespeople.sname from salespeople inner join customer on salespeopl
e.snum=customer.snum group by sname having count(*)=1;
+---------+
| sname   |
+---------+
| AxelRod |
| Motika  |
| Rifkin  |
+---------+
3 rows in set (0.00 sec)
```

49. Write a query that joins the Customer table to itself to find all pairs of customers served by a single salesperson.



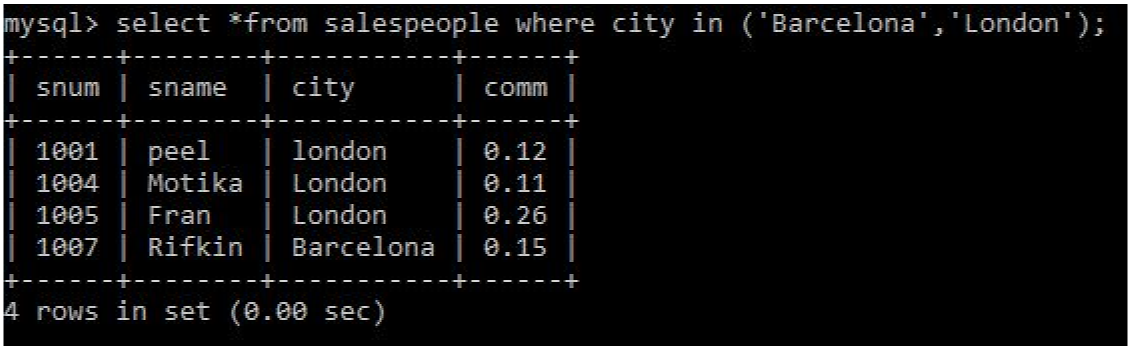
```
mysql> select distinct salespeople.sname,customer.cname from salespeople inner join customer on salespeople.snum=customer.snum;
+---------+----------+
| sname   | cname    |
+---------+----------+
| peel    | Hoffman  |
| peel    | Clemens  |
| serres  | Liu      |
| serres  | Grass    |
| AxelRod | Giovanni |
| Motika  | Pereira  |
| Rifkin  | Cisneros |
+---------+----------+
7 rows in set (0.00 sec)
```

50. Write a query that will give you all orders for more than $1000.00



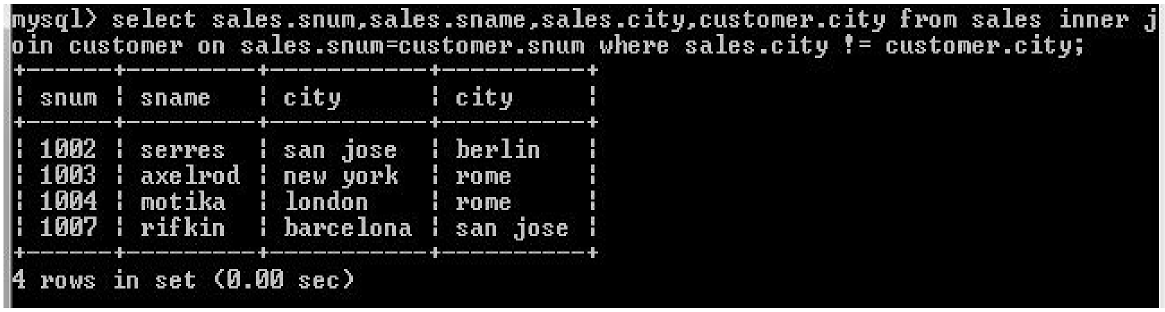
```
mysql> select onum,amt,odate,cnum from orders where amt>1000.00;
+------+---------+------------+------+
| onum | amt     | odate      | cnum |
+------+---------+------------+------+
| 3002 | 1900.10 | 1996-03-10 | 2007 |
| 3005 | 5160.45 | 1996-03-10 | 2003 |
| 3006 | 1098.16 | 1996-03-10 | 2008 |
| 3008 | 4723.00 | 1996-05-10 | 2006 |
| 3009 | 1713.23 | 1996-04-10 | 2002 |
| 3010 | 1309.95 | 1996-06-10 | 2004 |
| 3011 | 9891.88 | 1996-06-10 | 2006 |
+------+---------+------------+------+
7 rows in set (0.00 sec)
```

51. Write a query that lists each order number followed by the name of the customer who made that order.

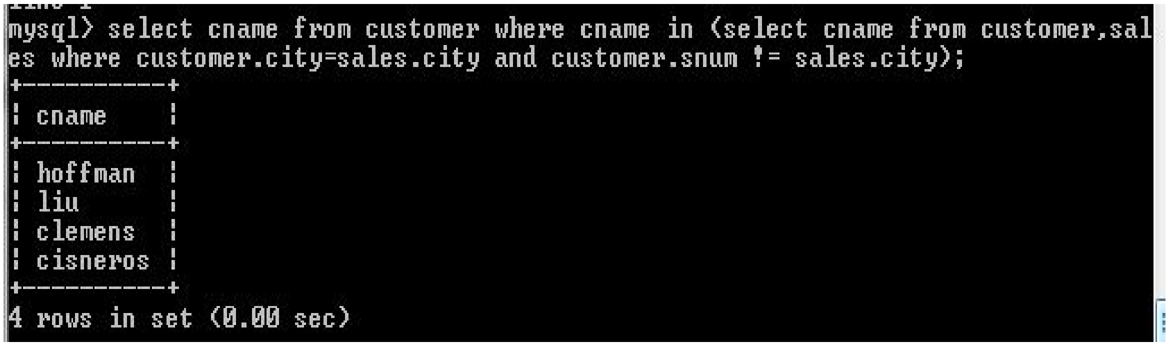


```
mysql> select *from salespeople where city in ('Barcelona','London');
+------+--------+-----------+------+
| snum | sname  | city      | comm |
+------+--------+-----------+------+
| 1001 | peel   | london    | 0.12 |
| 1004 | Motika | London    | 0.11 |
| 1005 | Fran   | London    | 0.26 |
| 1007 | Rifkin | Barcelona | 0.15 |
+------+--------+-----------+------+
4 rows in set (0.00 sec)
```

52. Write 2 queries that select all salespeople (by name and number) who have customers in their cities who they do not service, one using a join and one a corelated subquery. Which solution is more elegant?



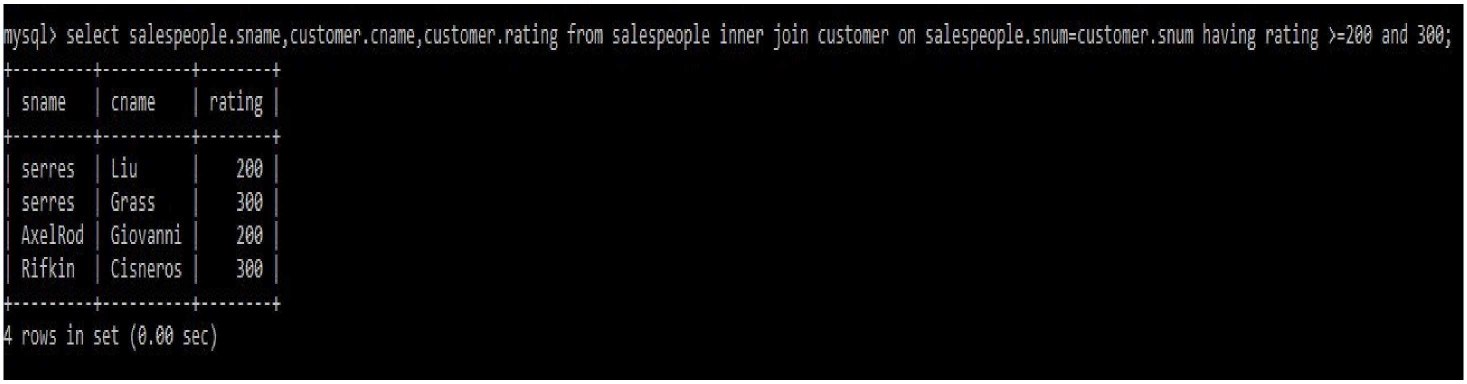
```
mysql> select sales.snum,sales.sname,sales.city,customer.city from sales inner j
oin customer on sales.snum=customer.snum where sales.city != customer.city;
+------+---------+-----------+----------+
| snum | sname   | city      | city     |
+------+---------+-----------+----------+
| 1002 | serres  | san jose  | berlin   |
| 1003 | axelrod | new york  | rome     |
| 1004 | motika  | london    | rome     |
| 1007 | rifkin  | barcelona | san jose |
+------+---------+-----------+----------+
4 rows in set (0.00 sec)
```



```
mysql> select cname from customer where cname in (select cname from customer,sal
es where customer.city=sales.city and customer.snum != sales.city);
+----------+
| cname    |
+----------+
| hoffman  |
| liu      |
| clemens  |
| cisneros |
+----------+
4 rows in set (0.00 sec)
```

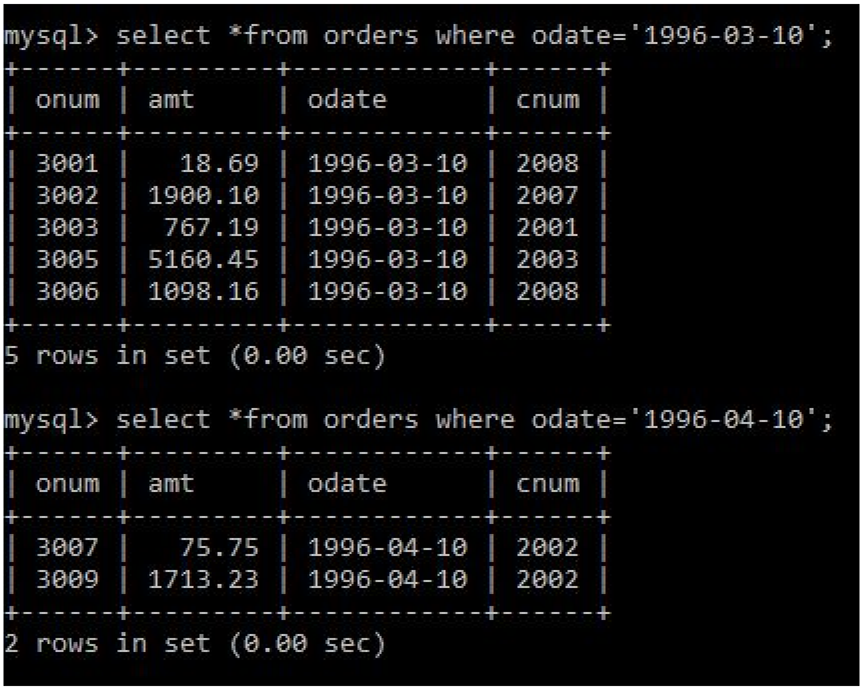
Ans:: Join query is more elegant than other queries.

53. Write a query that selects all customers whose ratings are equal to or greater than ANY (in the SQL sense) of Serres'?



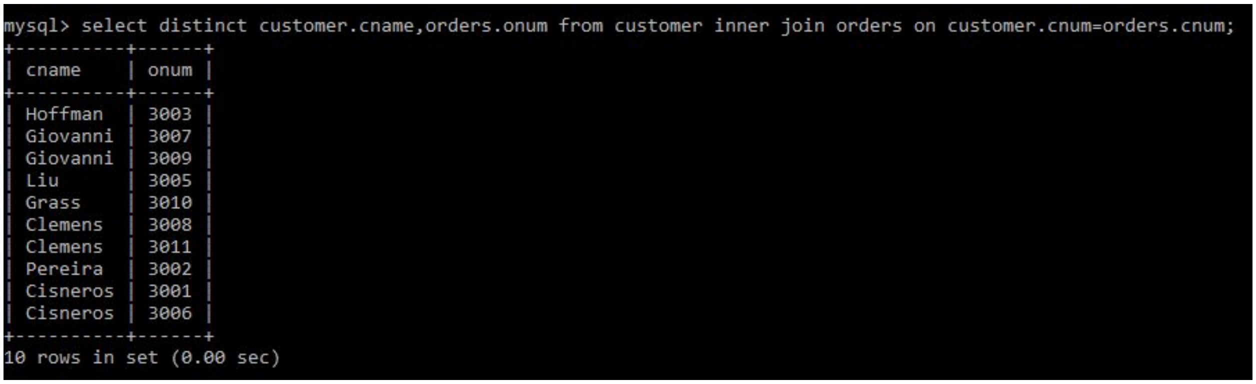
```
mysql> select salespeople.sname,customer.cname,customer.rating from salespeople inner join customer on salespeople.snum=customer.snum having rating >=200 and 300;
+---------+---------+--------+
| sname   | cname   | rating |
+---------+---------+--------+
| serres  | Liu     |    200 |
| serres  | Grass   |    300 |
| AxelRod | Giovanni|    200 |
| Rifkin  | Cisneros|    300 |
+---------+---------+--------+
4 rows in set (0.00 sec)
```

54. Write 2 queries that will produce all orders taken on October 3 or October 4.



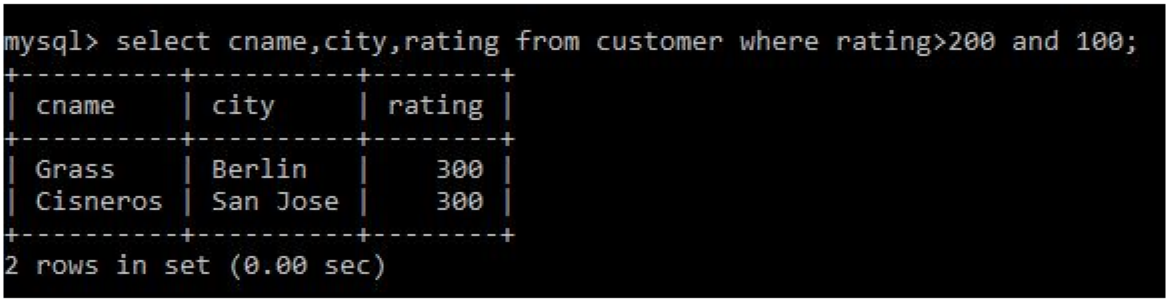
```
mysql> select *from orders where odate='1996-03-10';
+------+---------+------------+------+
| onum | amt     | odate      | cnum |
+------+---------+------------+------+
| 3001 |   18.69 | 1996-03-10 | 2008 |
| 3002 | 1900.10 | 1996-03-10 | 2007 |
| 3003 |  767.19 | 1996-03-10 | 2001 |
| 3005 | 5160.45 | 1996-03-10 | 2003 |
| 3006 | 1098.16 | 1996-03-10 | 2008 |
+------+---------+------------+------+
5 rows in set (0.00 sec)

mysql> select *from orders where odate='1996-04-10';
+------+---------+------------+------+
| onum | amt     | odate      | cnum |
+------+---------+------------+------+
| 3007 |   75.75 | 1996-04-10 | 2002 |
| 3009 | 1713.23 | 1996-04-10 | 2002 |
+------+---------+------------+------+
2 rows in set (0.00 sec)
```

55. Write a query that produces all pairs of orders by a given customer. Name that customer and eliminate duplicates.



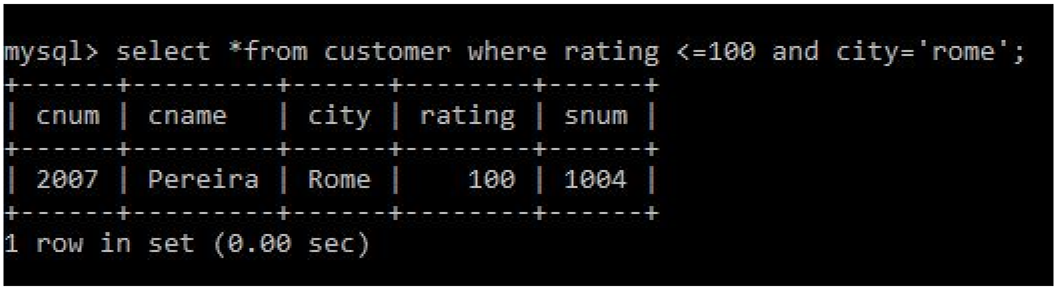
```
mysql> select distinct customer.cname,orders.onum from customer inner join orders on customer.cnum=orders.cnum;
+----------+------+
| cname    | onum |
+----------+------+
| Hoffman  | 3003 |
| Giovanni | 3007 |
| Giovanni | 3009 |
| Liu      | 3005 |
| Grass    | 3010 |
| Clemens  | 3008 |
| Clemens  | 3011 |
| Pereira  | 3002 |
| Cisneros | 3001 |
| Cisneros | 3006 |
+----------+------+
10 rows in set (0.00 sec)
```

56. Find only those customers whose ratings are higher than every customer in Rome.



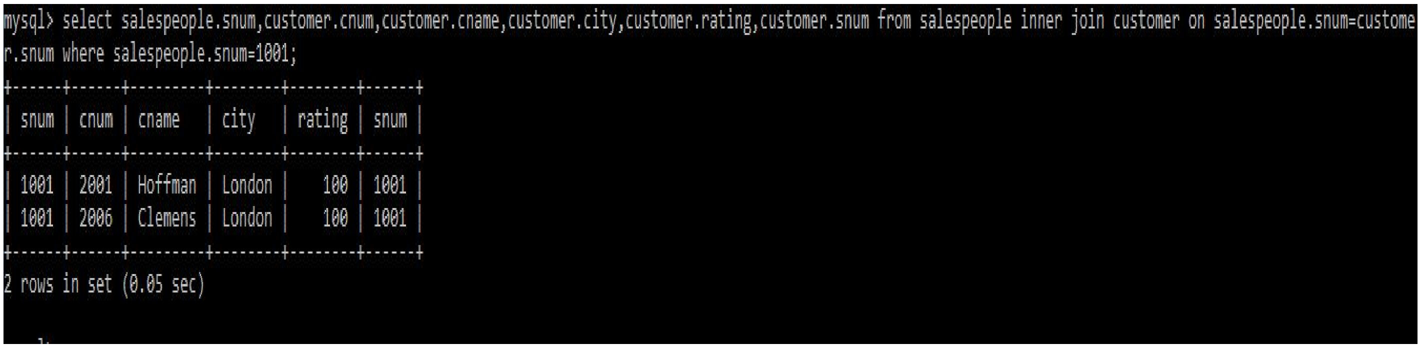
```
mysql> select cname,city,rating from customer where rating>200 and 100;
+----------+----------+--------+
| cname    | city     | rating |
+----------+----------+--------+
| Grass    | Berlin   |    300 |
| Cisneros | San Jose |    300 |
+----------+----------+--------+
2 rows in set (0.00 sec)
```

57. Write a query on the Customers table whose output will exclude all customers with a rating <= 100.00, unless they are located in Rome.



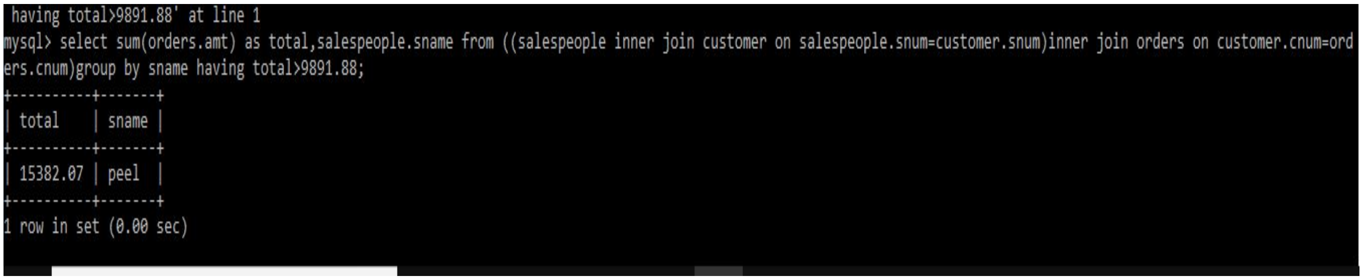
```
mysql> select *from customer where rating <=100 and city='rome';
+------+---------+------+--------+------+
| cnum | cname   | city | rating | snum |
+------+---------+------+--------+------+
| 2007 | Pereira | Rome |    100 | 1004 |
+------+---------+------+--------+------+
1 row in set (0.00 sec)
```

58. Find all rows from the Customers table for which the salesperson number is 1001.



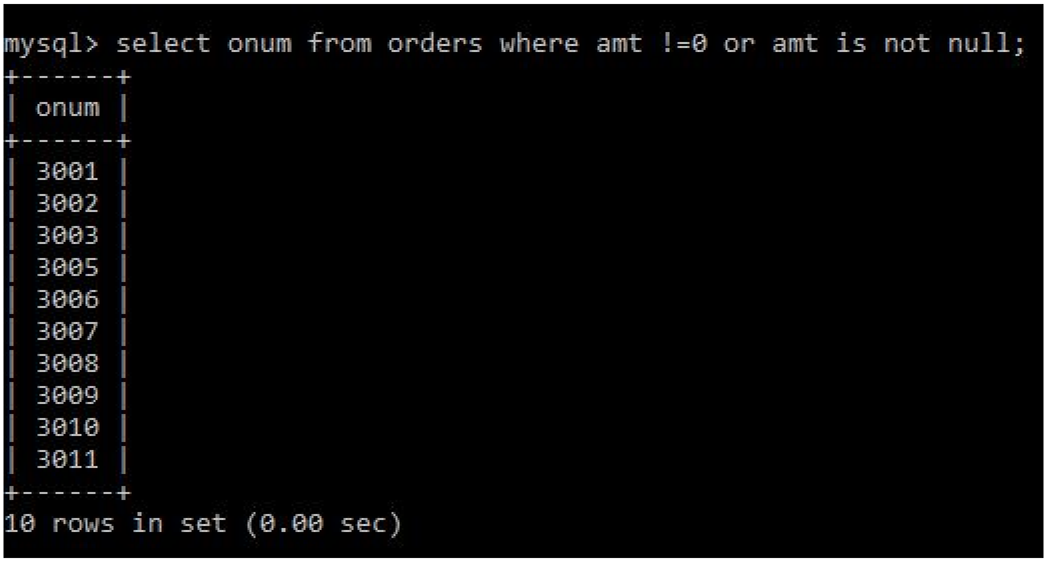
```
mysql> select salespeople.snum,customer.cnum,customer.cname,customer.city,customer.rating,customer.snum from salespeople inner join customer on salespeople.snum=custome
r.snum where salespeople.snum=1001;
+------+------+---------+--------+--------+------+
| snum | cnum | cname   | city   | rating | snum |
+------+------+---------+--------+--------+------+
| 1001 | 2001 | Hoffman | London |    100 | 1001 |
| 1001 | 2006 | Clemens | London |    100 | 1001 |
+------+------+---------+--------+--------+------+
2 rows in set (0.05 sec)
```

59. Find the total amount in Orders for each salesperson for whom this total is greater than the amount of the largest order in the table.



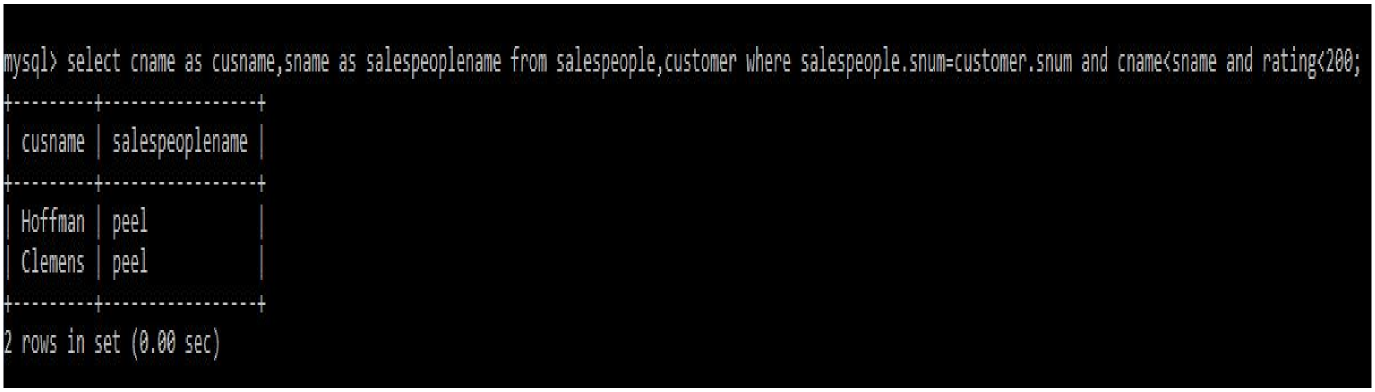
```
 having total>9891.88' at line 1
mysql> select sum(orders.amt) as total,salespeople.sname from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=ord
ers.cnum)group by sname having total>9891.88;
+----------+-------+
| total    | sname |
+----------+-------+
| 15382.07 | peel  |
+----------+-------+
1 row in set (0.00 sec)
```

60. Write a query that selects all orders save those with zeroes or NULLs in the amount field.



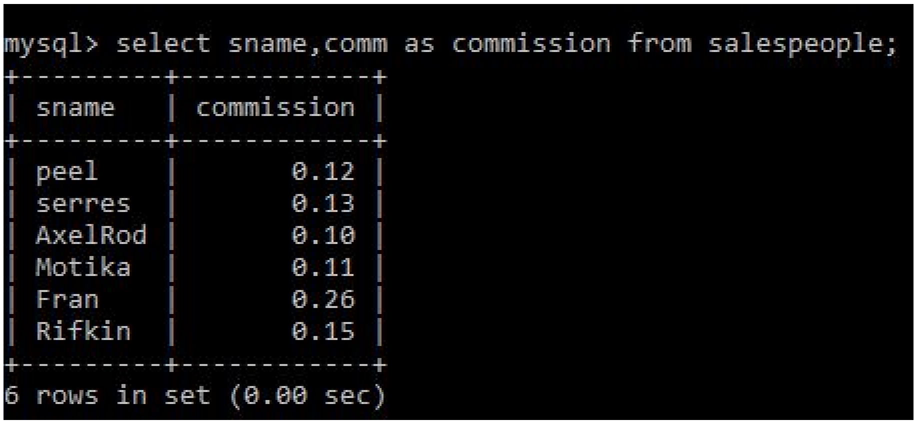
```
mysql> select onum from orders where amt !=0 or amt is not null;
+------+
| onum |
+------+
| 3001 |
| 3002 |
| 3003 |
| 3005 |
| 3006 |
| 3007 |
| 3008 |
| 3009 |
| 3010 |
| 3011 |
+------+
10 rows in set (0.00 sec)
```

61. Produce all combinations of salespeople and customer names such that the former precedes the latter alphabetically, and the latter has a rating of less than 200.



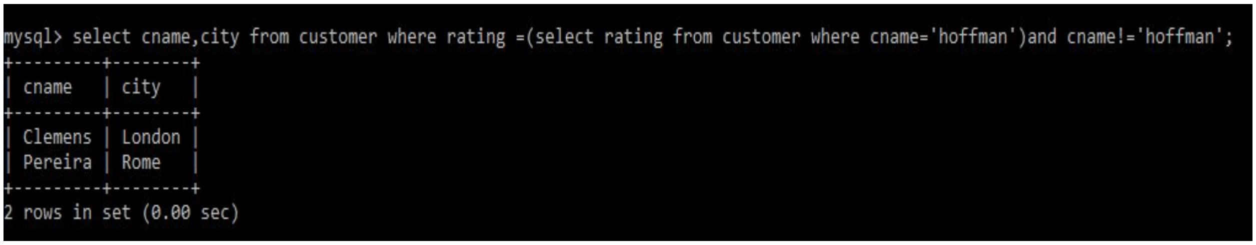
```
mysql> select cname as cusname,sname as salespeoplename from salespeople,customer where salespeople.snum=customer.snum and cname<sname and rating<200;
+---------+-----------------+
| cusname | salespeoplename |
+---------+-----------------+
| Hoffman | peel            |
| Clemens | peel            |
+---------+-----------------+
2 rows in set (0.00 sec)
```

62. List all Salespeople’s names and the Commission they have earned.



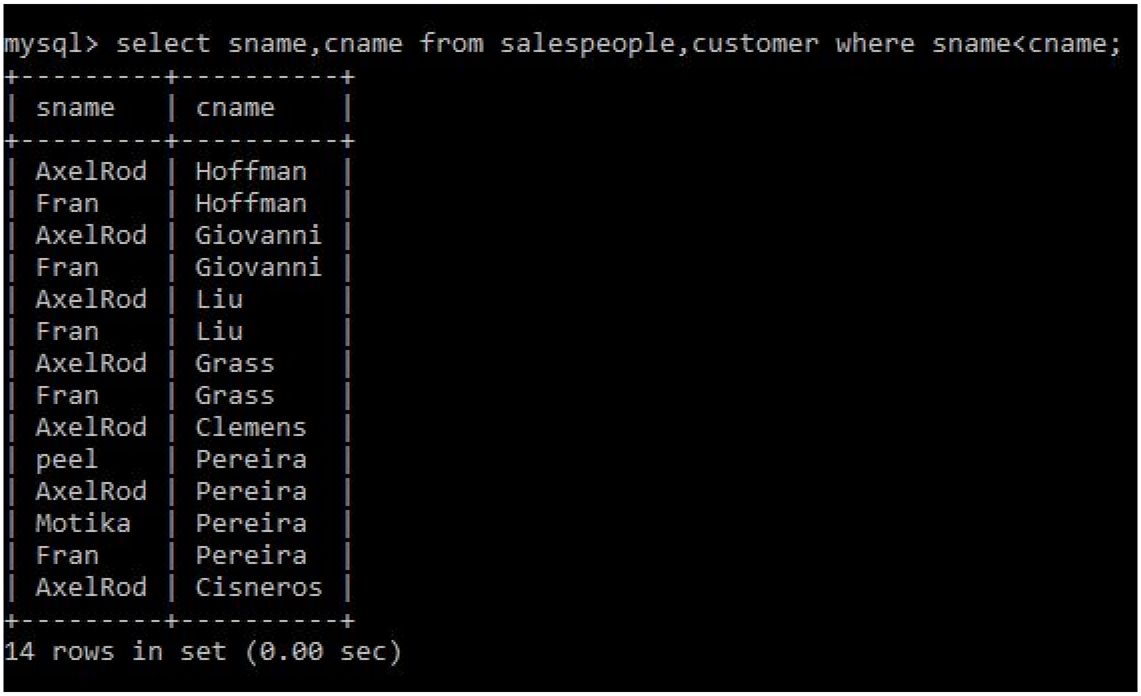
```
mysql> select sname,comm as commission from salespeople;
+---------+------------+
| sname   | commission |
+---------+------------+
| peel    |       0.12 |
| serres  |       0.13 |
| AxelRod |       0.10 |
| Motika  |       0.11 |
| Fran    |       0.26 |
| Rifkin  |       0.15 |
+---------+------------+
6 rows in set (0.00 sec)
```

63. Write a query that produces the names and cities of all customers with the same rating as Hoffman. Write the query using Hoffman's CNUM rather than his rating, so that it would still be usable if his rating changed.



```
mysql> select cname,city from customer where rating =(select rating from customer where cname='hoffman')and cname!='hoffman';
+---------+--------+
| cname   | city   |
+---------+--------+
| Clemens | London |
| Pereira | Rome   |
+---------+--------+
2 rows in set (0.00 sec)
```

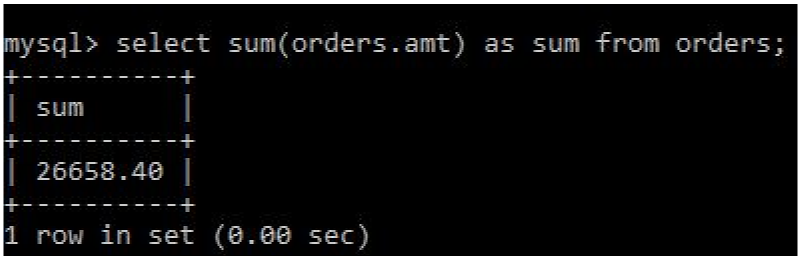
64. Find all salespeople for whom there are customers that follow them in alphabetical order.



```
mysql> select sname,cname from salespeople,customer where sname<cname;
+---------+----------+
| sname   | cname    |
+---------+----------+
| AxelRod | Hoffman  |
| Fran    | Hoffman  |
| AxelRod | Giovanni |
| Fran    | Giovanni |
| AxelRod | Liu      |
| Fran    | Liu      |
| AxelRod | Grass    |
| Fran    | Grass    |
| AxelRod | Clemens  |
| peel    | Pereira  |
| AxelRod | Pereira  |
| Motika  | Pereira  |
| Fran    | Pereira  |
| AxelRod | Cisneros |
+---------+----------+
14 rows in set (0.00 sec)
```

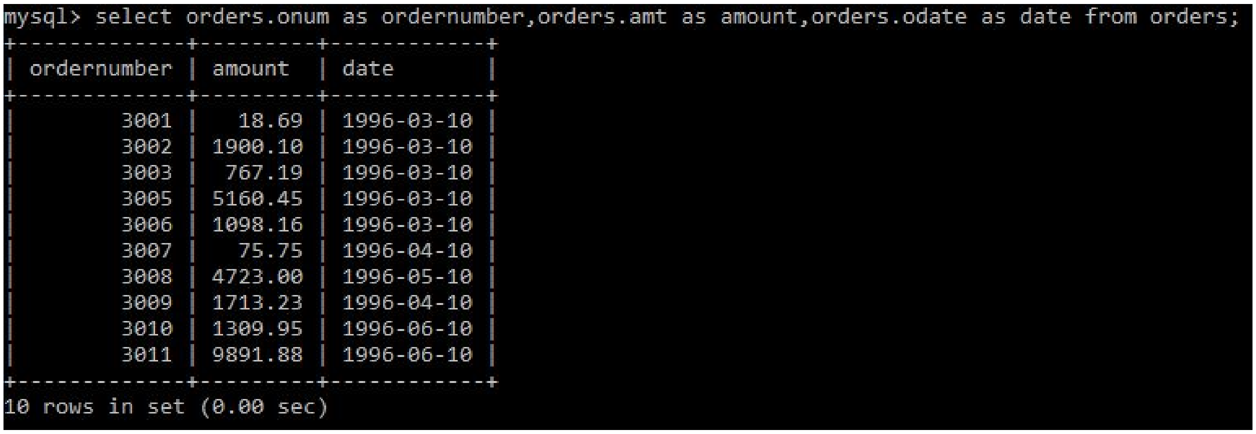
65. Write a query that produces the names and ratings of all customers of all who have above average orders.

66. Find the SUM of all purchases from the Orders table.



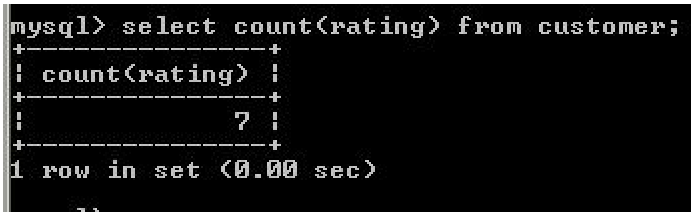
```
mysql> select sum(orders.amt) as sum from orders;
+----------+
| sum      |
+----------+
| 26658.40 |
+----------+
1 row in set (0.00 sec)
```

67. Write a SELECT command that produces the order number, amount and date for all rows in the order table.



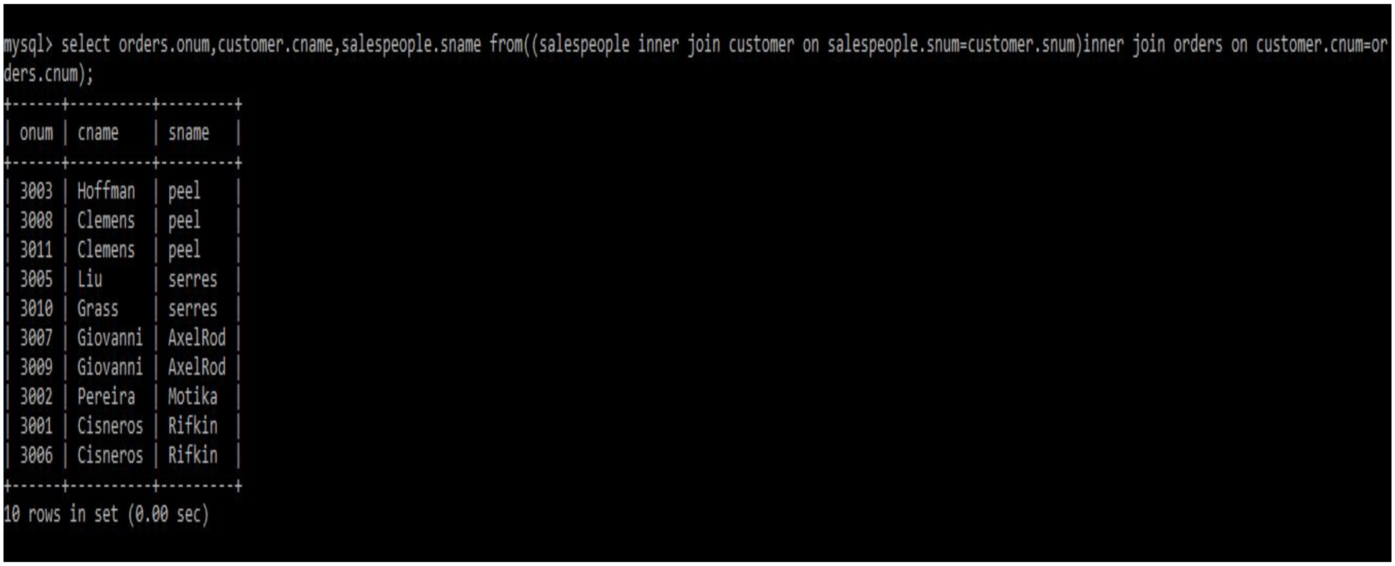
```
mysql> select orders.onum as ordernumber,orders.amt as amount,orders.odate as date from orders;
+-------------+---------+------------+
| ordernumber | amount  | date       |
+-------------+---------+------------+
|        3001 |   18.69 | 1996-03-10 |
|        3002 | 1900.10 | 1996-03-10 |
|        3003 |  767.19 | 1996-03-10 |
|        3005 | 5160.45 | 1996-03-10 |
|        3006 | 1098.16 | 1996-03-10 |
|        3007 |   75.75 | 1996-04-10 |
|        3008 | 4723.00 | 1996-05-10 |
|        3009 | 1713.23 | 1996-04-10 |
|        3010 | 1309.95 | 1996-06-10 |
|        3011 | 9891.88 | 1996-06-10 |
+-------------+---------+------------+
10 rows in set (0.00 sec)
```

68. Count the number of nonNULL rating fields in the Customers table (including repeats).



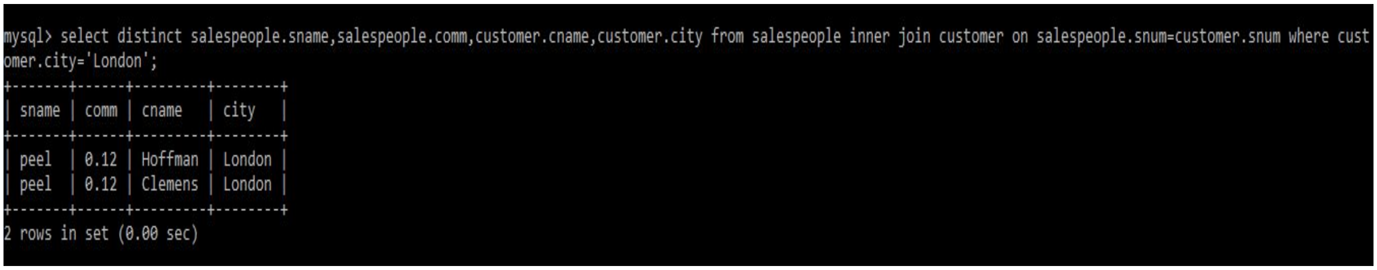
```
mysql> select count(rating) from customer;
+---------------+
| count(rating) |
+---------------+
|             7 |
+---------------+
1 row in set (0.00 sec)
```

69. Write a query that gives the names of both the salesperson and the customer for each order after the order number.



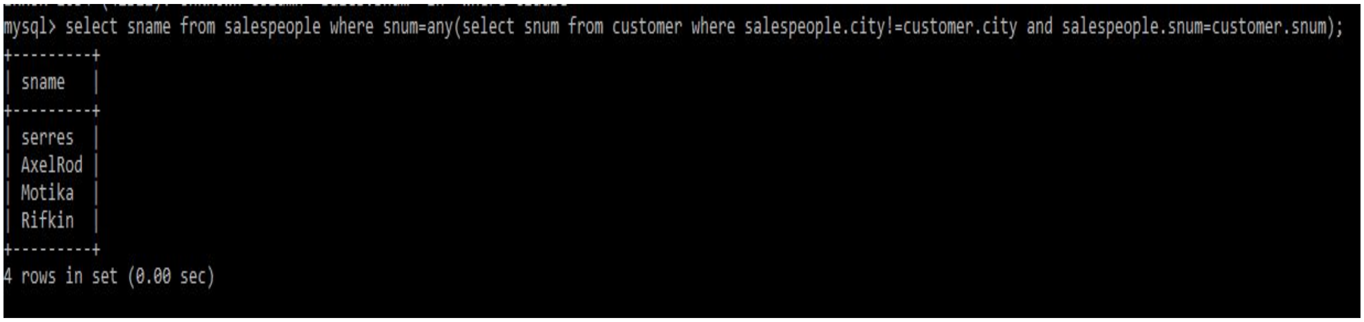
```
mysql> select orders.onum,customer.cname,salespeople.sname from((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=or
ders.cnum);
+------+----------+---------+
| onum | cname    | sname   |
+------+----------+---------+
| 3003 | Hoffman  | peel    |
| 3008 | Clemens  | peel    |
| 3011 | Clemens  | peel    |
| 3005 | Liu      | serres  |
| 3010 | Grass    | serres  |
| 3007 | Giovanni | AxelRod |
| 3009 | Giovanni | AxelRod |
| 3002 | Pereira  | Motika  |
| 3001 | Cisneros | Rifkin  |
| 3006 | Cisneros | Rifkin  |
+------+----------+---------+
10 rows in set (0.00 sec)
```

70. List the commissions of all salespeople servicing customers in London.



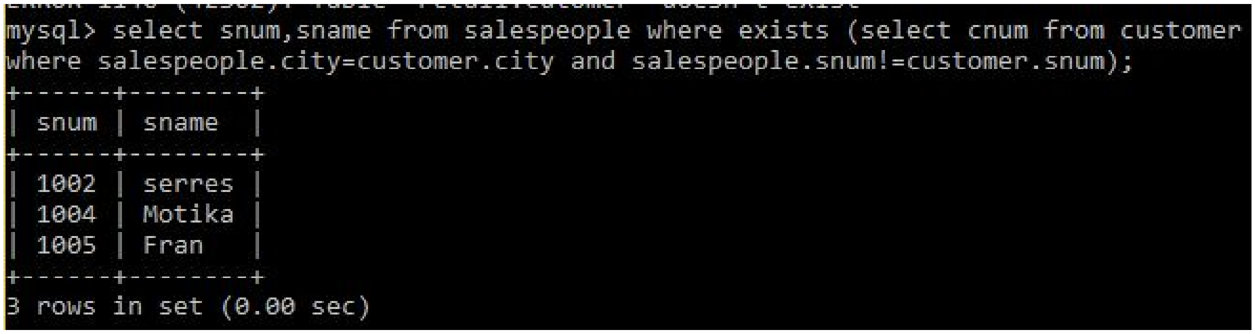
```
mysql> select distinct salespeople.sname,salespeople.comm,customer.cname,customer.city from salespeople inner join customer on salespeople.snum=customer.snum where cust
omer.city='London';
+-------+------+---------+--------+
| sname | comm | cname   | city   |
+-------+------+---------+--------+
| peel  | 0.12 | Hoffman | London |
| peel  | 0.12 | Clemens | London |
+-------+------+---------+--------+
2 rows in set (0.00 sec)
```

71. Write a query using ANY or ALL that will find all salespeople who have no customers
located in their city.



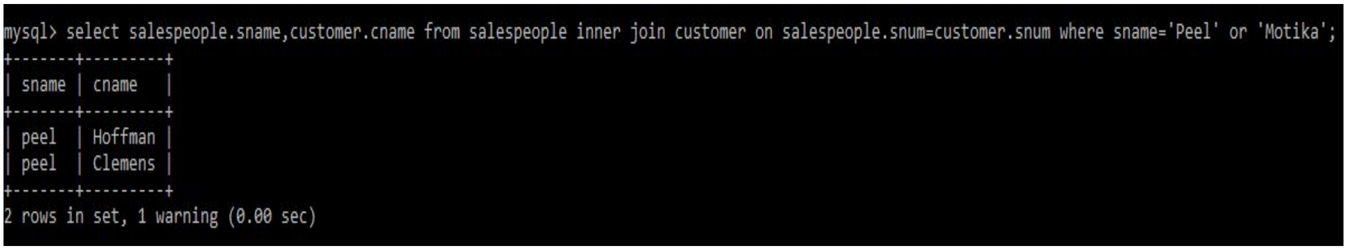
```
mysql> select sname from salespeople where snum=any(select snum from customer where salespeople.city!=customer.city and salespeople.snum=customer.snum);
+---------+
| sname   |
+---------+
| serres  |
| AxelRod |
| Motika  |
| Rifkin  |
+---------+
4 rows in set (0.00 sec)
```

72. Write a query using the EXISTS operator that selects all salespeople with customers located
in their cities who are not assigned to them.



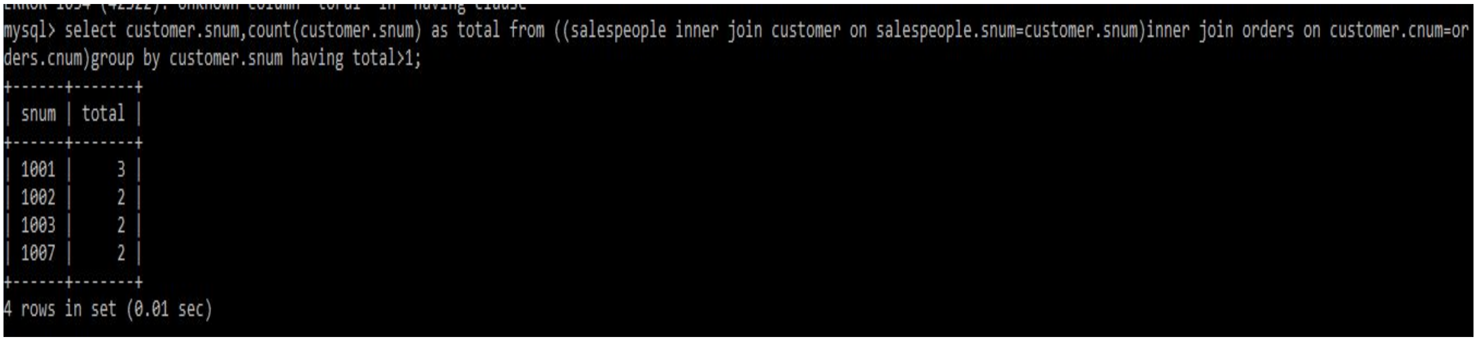
```
mysql> select snum,sname from salespeople where exists (select cnum from customer
where salespeople.city=customer.city and salespeople.snum!=customer.snum);
+------+--------+
| snum | sname  |
+------+--------+
| 1002 | serres |
| 1004 | Motika |
| 1005 | Fran   |
+------+--------+
3 rows in set (0.00 sec)
```

73. Write a query that selects all customers serviced by Peel or Motika. (Hint : The SNUM field
relates the two tables to one another.)



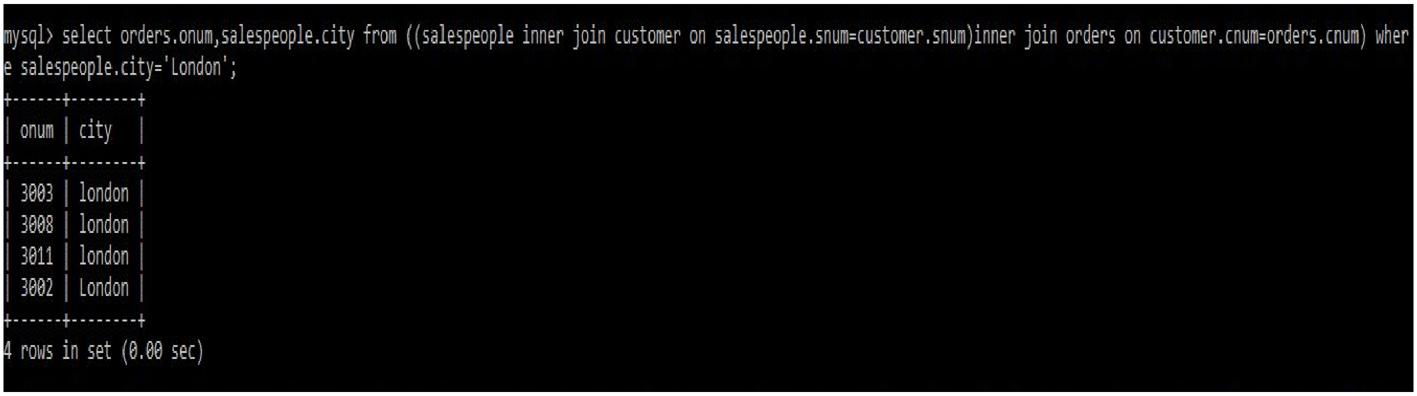
```
mysql> select salespeople.sname,customer.cname from salespeople inner join customer on salespeople.snum=customer.snum where sname='Peel' or 'Motika';
+-------+---------+
| sname | cname   |
+-------+---------+
| peel  | Hoffman |
| peel  | Clemens |
+-------+---------+
2 rows in set, 1 warning (0.00 sec)
```

74. Count the number of salespeople registering orders for each day. (If a salesperson has more than one order on a given day, he or she should be counted only once.)



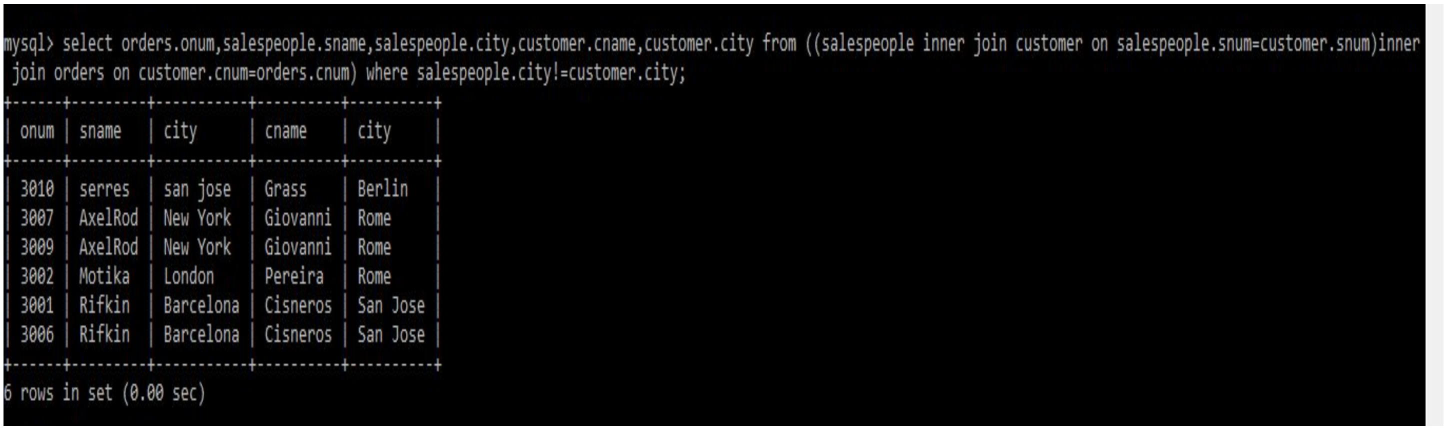
```
ERROR 1054 (42S22): Unknown column 'total' in 'having clause'
mysql> select customer.snum,count(customer.snum) as total from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=or
ders.cnum)group by customer.snum having total>1;
+------+-------+
| snum | total |
+------+-------+
| 1001 |     3 |
| 1002 |     2 |
| 1003 |     2 |
| 1007 |     2 |
+------+-------+
4 rows in set (0.01 sec)
```

75. Find all orders attributed to salespeople in London.



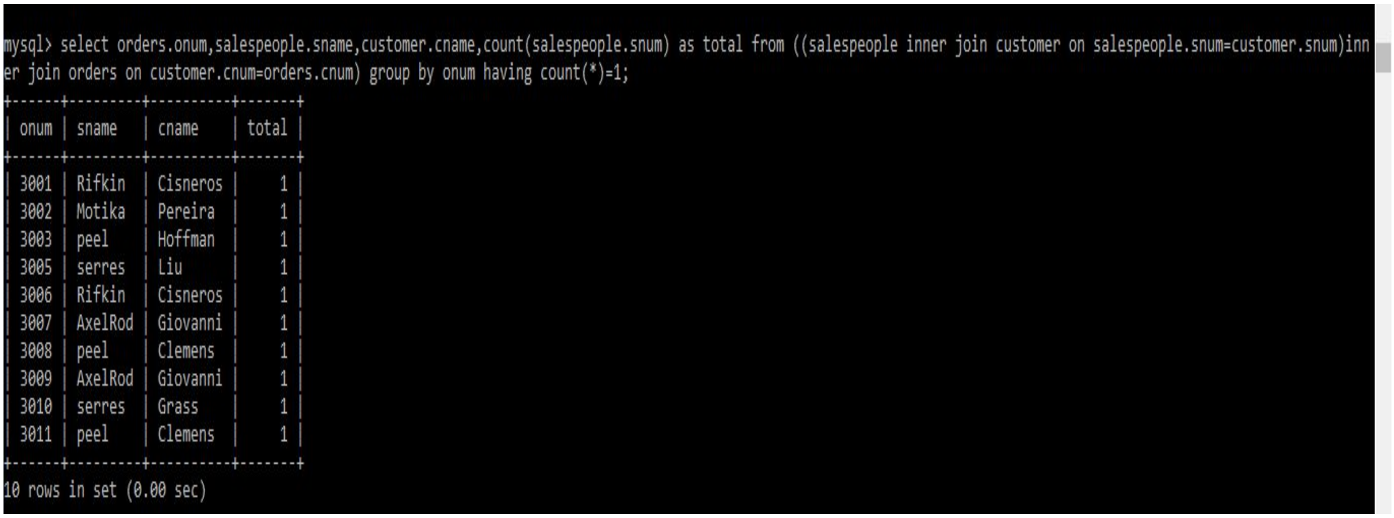
```
mysql> select orders.onum,salespeople.city from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum) wher
e salespeople.city='London';
+------+--------+
| onum | city   |
+------+--------+
| 3003 | london |
| 3008 | london |
| 3011 | london |
| 3002 | London |
+------+--------+
4 rows in set (0.00 sec)
```

76. Find all orders by customers not located in the same cities as their salespeople.



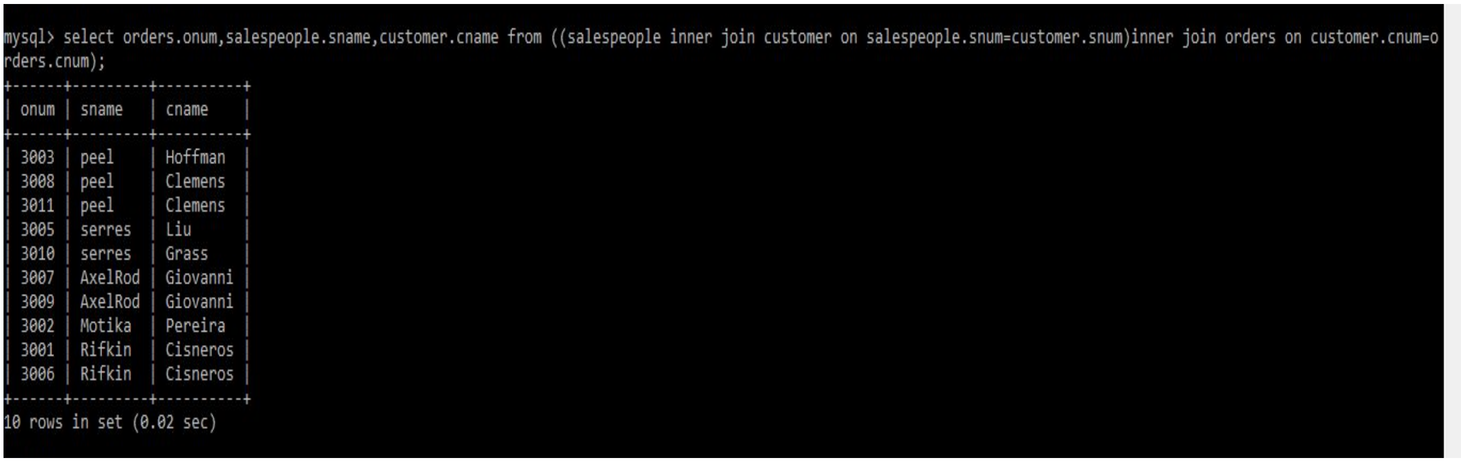
```
mysql> select orders.onum,salespeople.sname,salespeople.city,customer.cname,customer.city from ((salespeople inner join customer on salespeople.snum=customer.snum)inner
 join orders on customer.cnum=orders.cnum) where salespeople.city!=customer.city;
+------+---------+-----------+----------+----------+
| onum | sname   | city      | cname    | city     |
+------+---------+-----------+----------+----------+
| 3010 | serres  | san jose  | Grass    | Berlin   |
| 3007 | AxelRod | New York  | Giovanni | Rome     |
| 3009 | AxelRod | New York  | Giovanni | Rome     |
| 3002 | Motika  | London    | Pereira  | Rome     |
| 3001 | Rifkin  | Barcelona | Cisneros | San Jose |
| 3006 | Rifkin  | Barcelona | Cisneros | San Jose |
+------+---------+-----------+----------+----------+
6 rows in set (0.00 sec)
```

77. Find all salespeople who have customers with more than one current order.



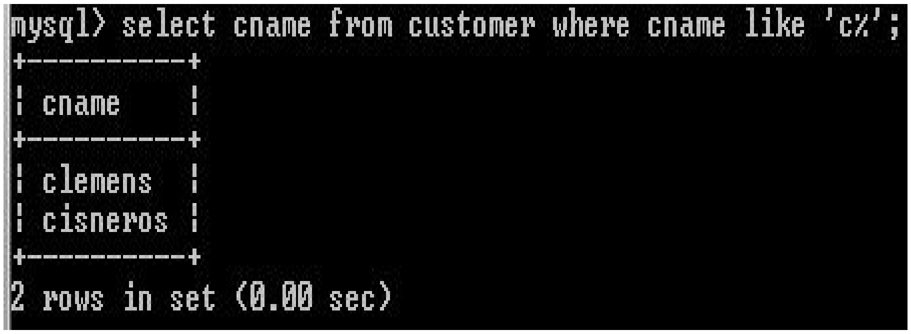
```
mysql> select orders.onum,salespeople.sname,customer.cname,count(salespeople.snum) as total from ((salespeople inner join customer on salespeople.snum=customer.snum)inn
er join orders on customer.cnum=orders.cnum) group by onum having count(*)=1;
+------+---------+----------+-------+
| onum | sname   | cname    | total |
+------+---------+----------+-------+
| 3001 | Rifkin  | Cisneros |     1 |
| 3002 | Motika  | Pereira  |     1 |
| 3003 | peel    | Hoffman  |     1 |
| 3005 | serres  | Liu      |     1 |
| 3006 | Rifkin  | Cisneros |     1 |
| 3007 | AxelRod | Giovanni |     1 |
| 3008 | peel    | Clemens  |     1 |
| 3009 | AxelRod | Giovanni |     1 |
| 3010 | serres  | Grass    |     1 |
| 3011 | peel    | Clemens  |     1 |
+------+---------+----------+-------+
10 rows in set (0.00 sec)
```

78. Write a query that extracts from the Customers table every customer assigned to a salesperson who currently has at least one other customer (besides the customer being selected) with orders in the Orders table.



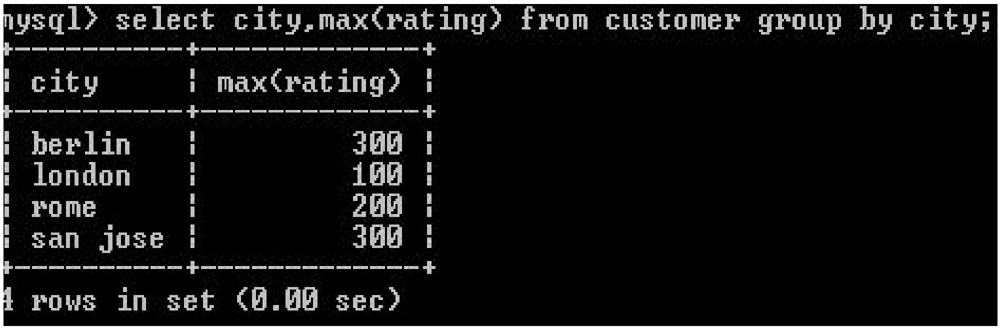
```
mysql> select orders.onum,salespeople.sname,customer.cname from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=o
rders.cnum);
+------+---------+----------+
| onum | sname   | cname    |
+------+---------+----------+
| 3003 | peel    | Hoffman  |
| 3008 | peel    | Clemens  |
| 3011 | peel    | Clemens  |
| 3005 | serres  | Liu      |
| 3010 | serres  | Grass    |
| 3007 | AxelRod | Giovanni |
| 3009 | AxelRod | Giovanni |
| 3002 | Motika  | Pereira  |
| 3001 | Rifkin  | Cisneros |
| 3006 | Rifkin  | Cisneros |
+------+---------+----------+
10 rows in set (0.02 sec)
```

79. Write a query that selects all customers whose names begin with ‘C’.



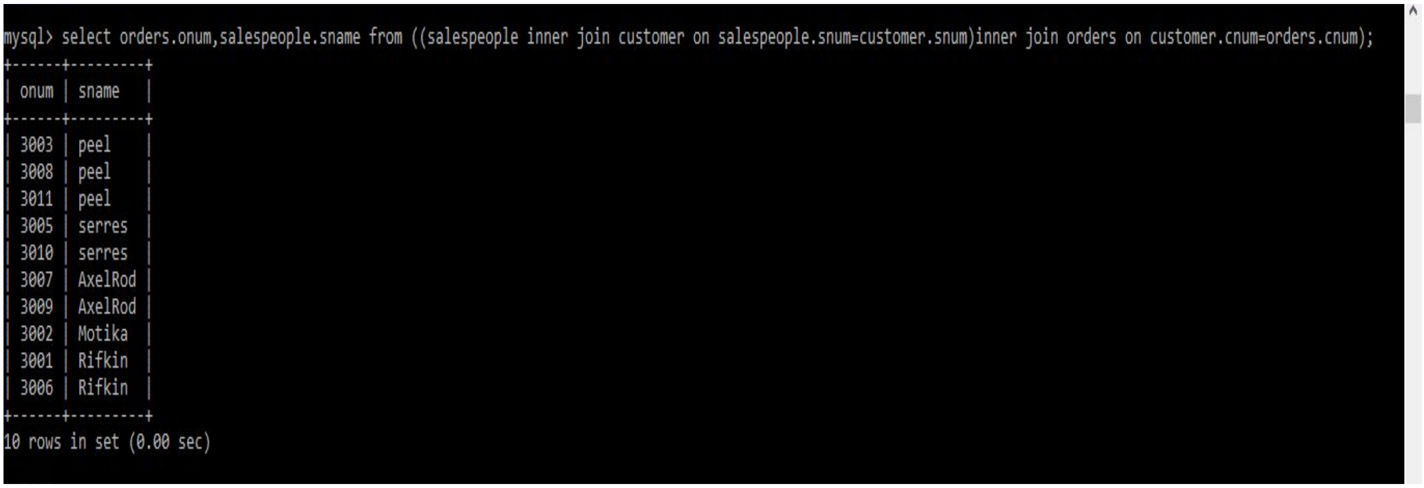
```
mysql> select cname from customer where cname like 'c%';
+----------+
| cname    |
+----------+
| clemens  |
| cisneros |
+----------+
2 rows in set (0.00 sec)
```

80. Write a query on the Customers table that will find the highest rating in each city. Put the output in this form : for the city (city) the highest rating is : (rating).



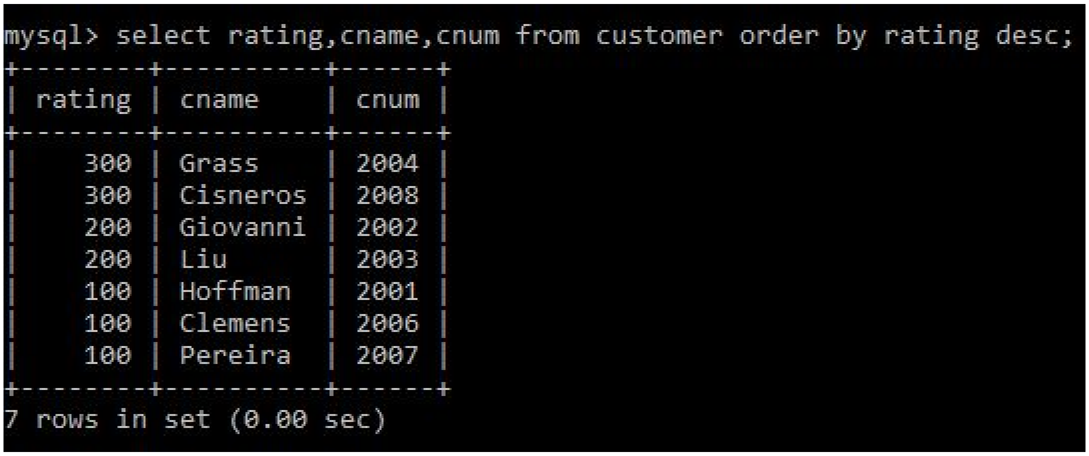
```
mysql> select city,max(rating) from customer group by city;
+----------+--------------+
| city     | max(rating)  |
+----------+--------------+
| berlin   |          300 |
| london   |          100 |
| rome     |          200 |
| san jose |          300 |
+----------+--------------+
4 rows in set (0.00 sec)
```

81. Write a query that will produce the SNUM values of all salespeople with orders currently in the Orders table (without any repeats).



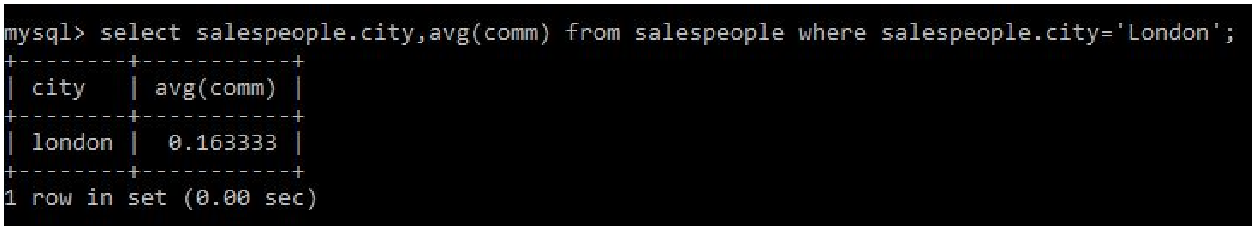
```
mysql> select orders.onum,salespeople.sname from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum);
+------+---------+
| onum | sname   |
+------+---------+
| 3003 | peel    |
| 3008 | peel    |
| 3011 | peel    |
| 3005 | serres  |
| 3010 | serres  |
| 3007 | AxelRod |
| 3009 | AxelRod |
| 3002 | Motika  |
| 3001 | Rifkin  |
| 3006 | Rifkin  |
+------+---------+
10 rows in set (0.00 sec)
```

82. Write a query that lists customers in descending order of rating. Output the rating field first, followed by the customer’s names and numbers.



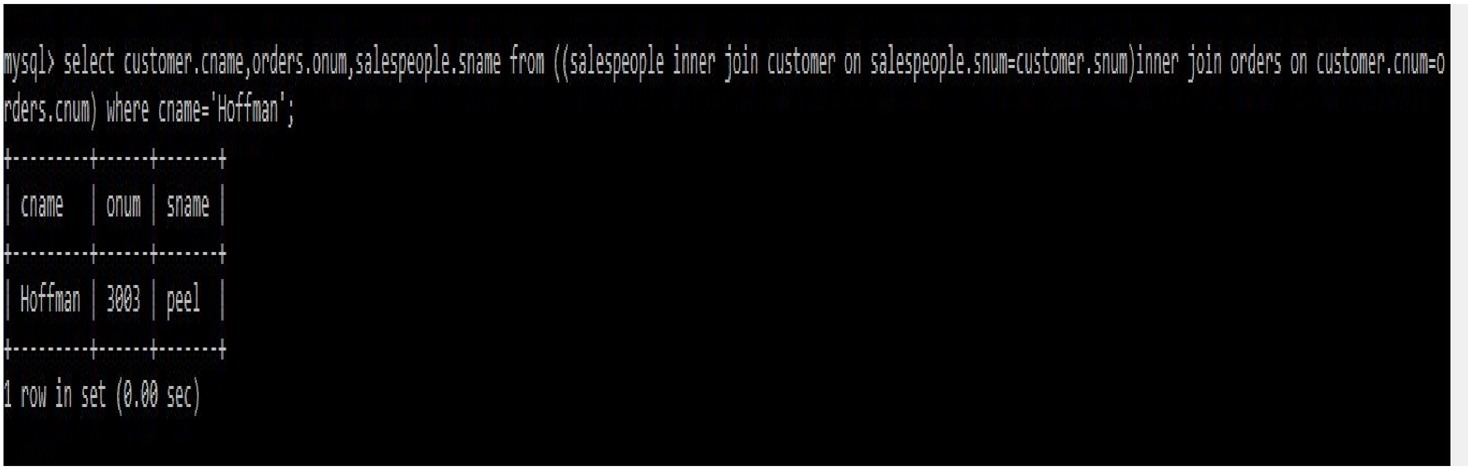
```
mysql> select rating,cname,cnum from customer order by rating desc;
+--------+----------+------+
| rating | cname    | cnum |
+--------+----------+------+
|    300 | Grass    | 2004 |
|    300 | Cisneros | 2008 |
|    200 | Giovanni | 2002 |
|    200 | Liu      | 2003 |
|    100 | Hoffman  | 2001 |
|    100 | Clemens  | 2006 |
|    100 | Pereira  | 2007 |
+--------+----------+------+
7 rows in set (0.00 sec)
```

83. Find the average commission for salespeople in London.



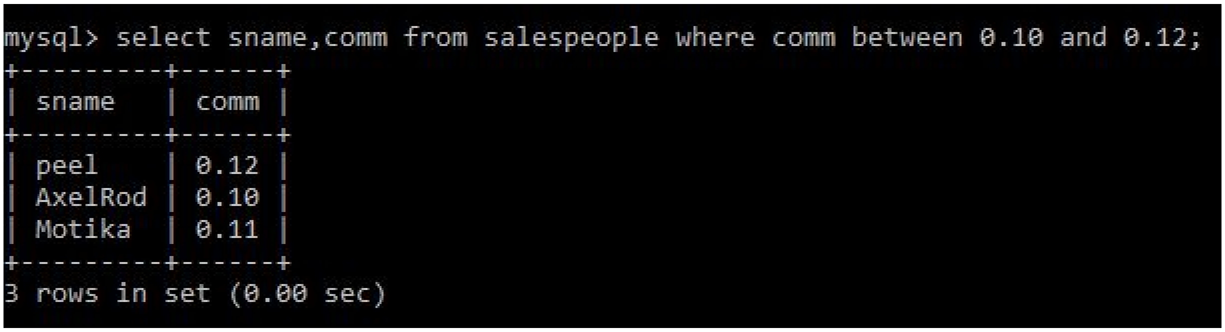
```
mysql> select salespeople.city,avg(comm) from salespeople where salespeople.city='London';
+--------+----------+
| city   | avg(comm) |
+--------+----------+
| london |  0.163333 |
+--------+----------+
1 row in set (0.00 sec)
```

84. Find all orders credited to the same salesperson who services Hoffman (CNUM 2001).



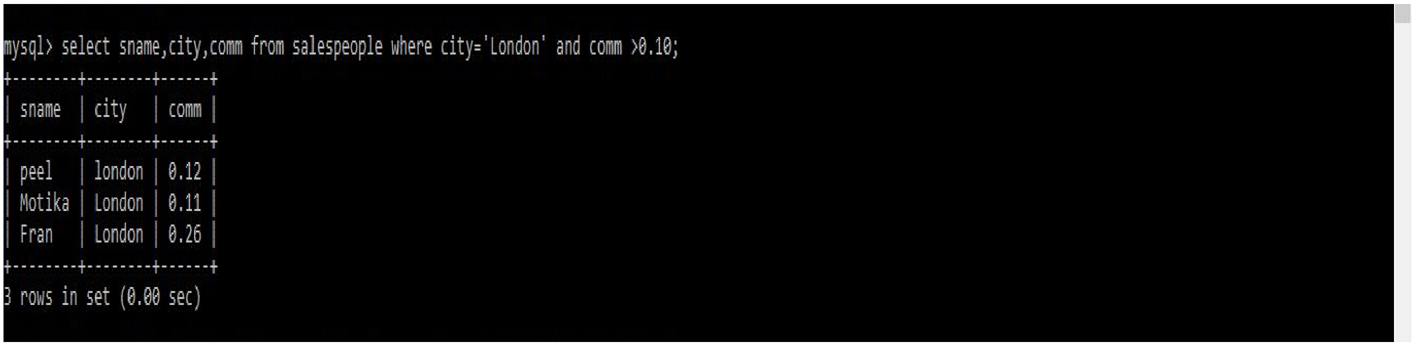
```
mysql> select customer.cname,orders.onum,salespeople.sname from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=o
rders.cnum) where cname='Hoffman';
+---------+------+-------+
| cname   | onum | sname |
+---------+------+-------+
| Hoffman | 3003 | peel  |
+---------+------+-------+
1 row in set (0.00 sec)
```

85. Find all salespeople whose commission is in between 0.10 and 0.12 (both inclusive).



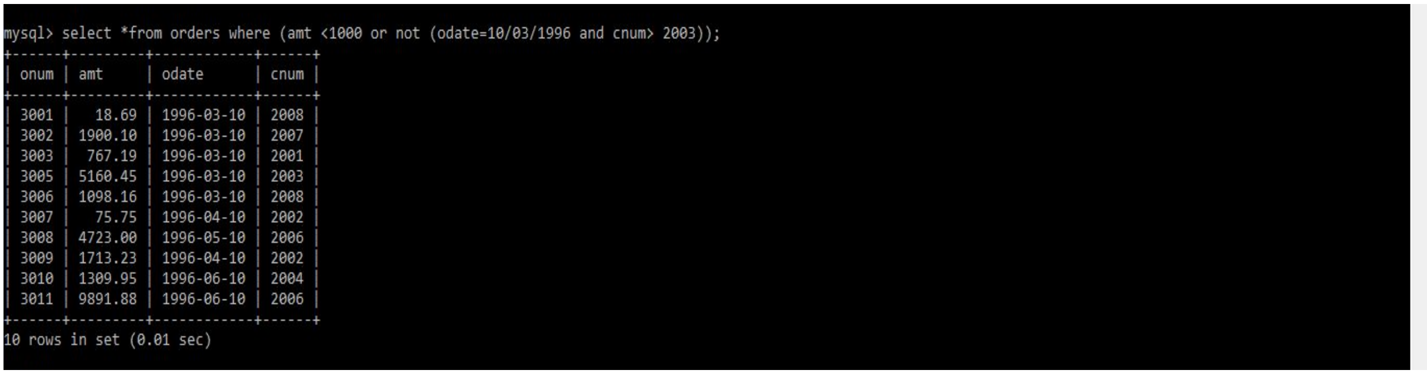
```
mysql> select sname,comm from salespeople where comm between 0.10 and 0.12;
+---------+------+
| sname   | comm |
+---------+------+
| peel    | 0.12 |
| AxelRod | 0.10 |
| Motika  | 0.11 |
+---------+------+
3 rows in set (0.00 sec)
```

86. Write a query that will give you the names and cities of all salespeople in London with a commission above 0.10.



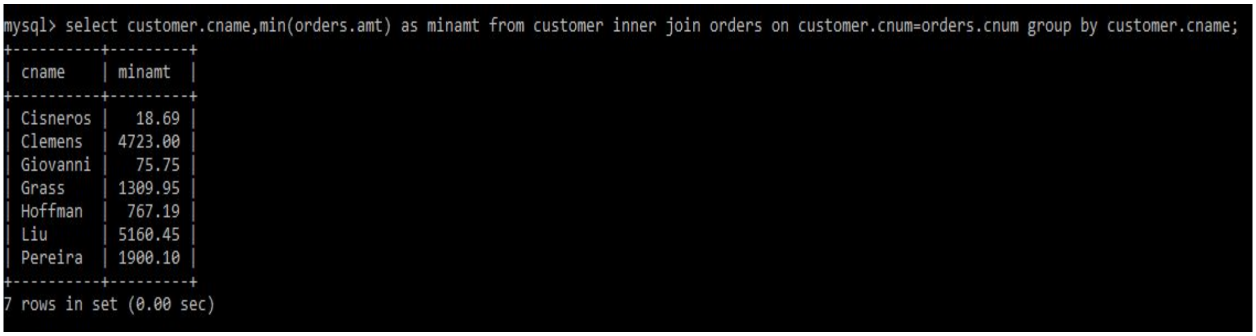
```
mysql> select sname,city,comm from salespeople where city='London' and comm >0.10;
+--------+--------+------+
| sname  | city   | comm |
+--------+--------+------+
| peel   | london | 0.12 |
| Motika | London | 0.11 |
| Fran   | London | 0.26 |
+--------+--------+------+
3 rows in set (0.00 sec)
```

87. What will be the output from the following query? SELECT * FROM ORDERS where (amt < 1000 OR NOT (odate = 10/03/1996 AND cnum > 2003));



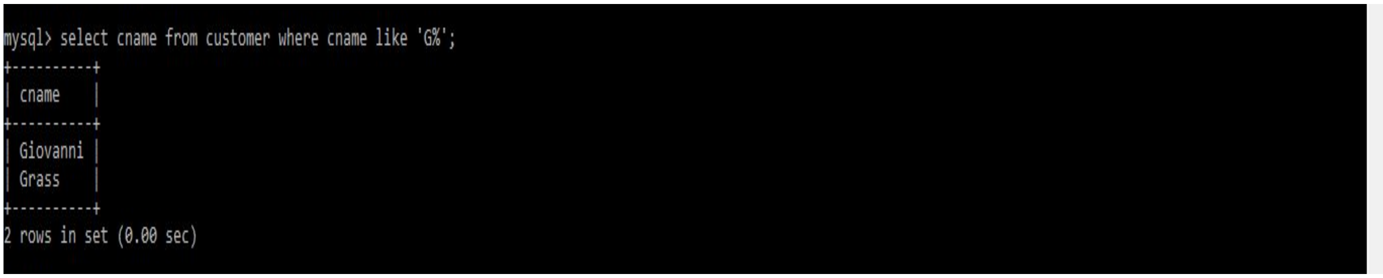
```
mysql> select *from orders where (amt <1000 or not (odate=10/03/1996 and cnum> 2003));
+------+---------+------------+------+
| onum | amt     | odate      | cnum |
+------+---------+------------+------+
| 3001 |   18.69 | 1996-03-10 | 2008 |
| 3002 | 1900.10 | 1996-03-10 | 2007 |
| 3003 |  767.19 | 1996-03-10 | 2001 |
| 3005 | 5160.45 | 1996-03-10 | 2003 |
| 3006 | 1098.16 | 1996-03-10 | 2008 |
| 3007 |   75.75 | 1996-04-10 | 2002 |
| 3008 | 4723.00 | 1996-05-10 | 2006 |
| 3009 | 1713.23 | 1996-04-10 | 2002 |
| 3010 | 1309.95 | 1996-06-10 | 2004 |
| 3011 | 9891.88 | 1996-06-10 | 2006 |
+------+---------+------------+------+
10 rows in set (0.01 sec)
```

88. Write a query that selects each customer’s smallest order.



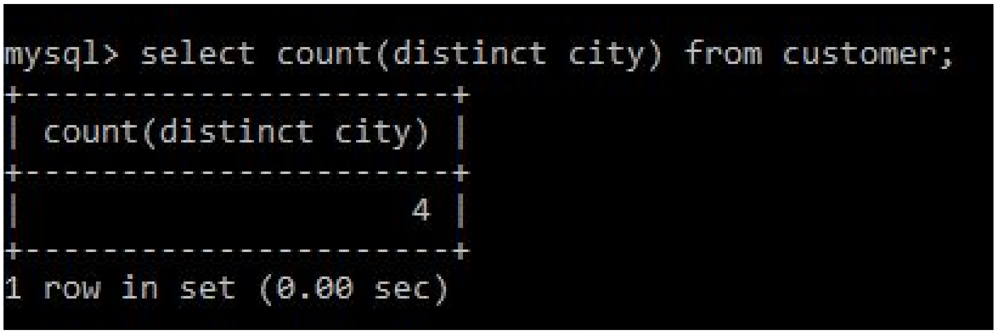
```
mysql> select customer.cname,min(orders.amt) as minamt from customer inner join orders on customer.cnum=orders.cnum group by customer.cname;
+----------+---------+
| cname    | minamt  |
+----------+---------+
| Cisneros |   18.69 |
| Clemens  | 4723.00 |
| Giovanni |   75.75 |
| Grass    | 1309.95 |
| Hoffman  |  767.19 |
| Liu      | 5160.45 |
| Pereira  | 1900.10 |
+----------+---------+
7 rows in set (0.00 sec)
```

89. Write a query that selects the first customer in alphabetical order whose name begins with G.



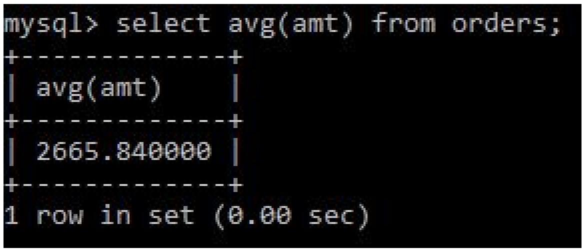
```
mysql> select cname from customer where cname like 'G%';
+----------+
| cname    |
+----------+
| Giovanni |
| Grass    |
+----------+
2 rows in set (0.00 sec)
```

90. Write a query that counts the number of different nonNULL city values in the Customers table.



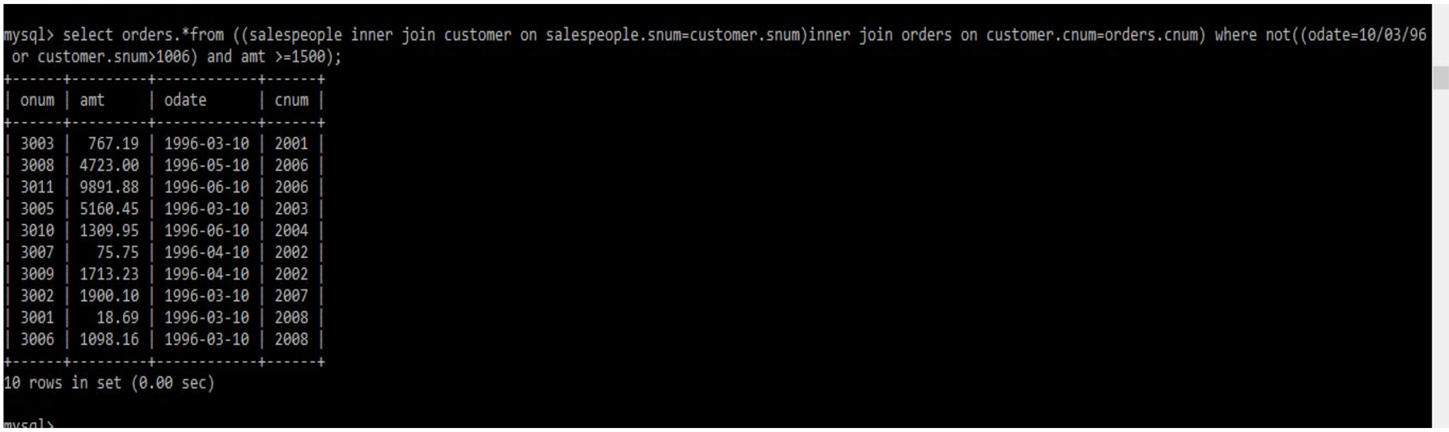
```
mysql> select count(distinct city) from customer;
+----------------------+
| count(distinct city) |
+----------------------+
|                    4 |
+----------------------+
1 row in set (0.00 sec)
```

91. Find the average amount from the Orders table.



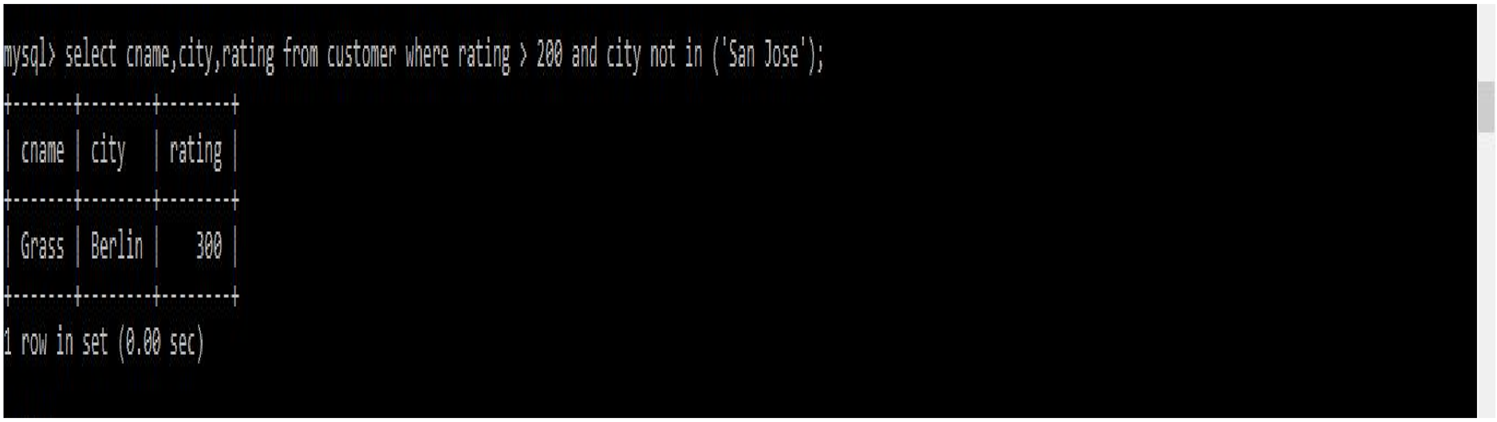
```
mysql> select avg(amt) from orders;
+-------------+
| avg(amt)    |
+-------------+
| 2665.840000 |
+-------------+
1 row in set (0.00 sec)
```

92. What would be the output from the following query? SELECT * FROM ORDERS WHERE NOT (odate = 10/03/96 OR snum > 1006) AND amt >= 1500);



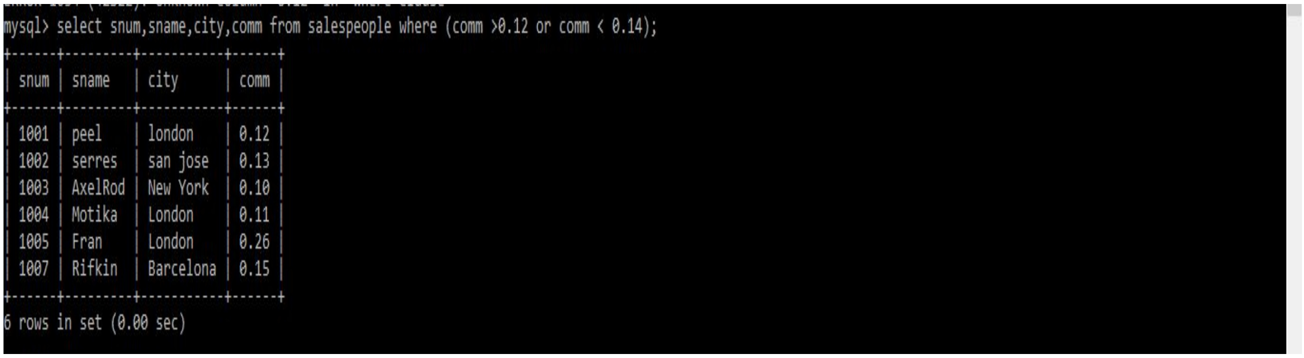
```
mysql> select orders.*from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum) where not((odate=10/03/96
 or customer.snum>1006) and amt >=1500);
+------+---------+------------+------+
| onum | amt     | odate      | cnum |
+------+---------+------------+------+
| 3003 |  767.19 | 1996-03-10 | 2001 |
| 3008 | 4723.00 | 1996-05-10 | 2006 |
| 3011 | 9891.88 | 1996-06-10 | 2006 |
| 3005 | 5160.45 | 1996-03-10 | 2003 |
| 3010 | 1309.95 | 1996-06-10 | 2004 |
| 3007 |   75.75 | 1996-04-10 | 2002 |
| 3009 | 1713.23 | 1996-04-10 | 2002 |
| 3002 | 1900.10 | 1996-03-10 | 2007 |
| 3001 |   18.69 | 1996-03-10 | 2008 |
| 3006 | 1098.16 | 1996-03-10 | 2008 |
+------+---------+------------+------+
10 rows in set (0.00 sec)

mysql>
```

93. Find all customers who are not located in San Jose and whose rating is above 200.



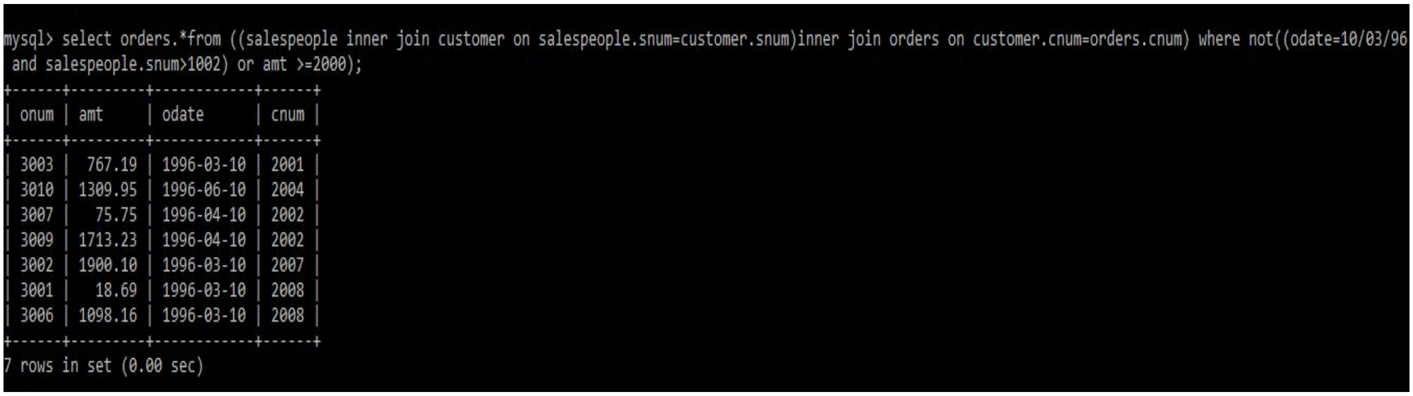
```
mysql> select cname,city,rating from customer where rating > 200 and city not in ('San Jose');
+-------+--------+--------+
| cname | city   | rating |
+-------+--------+--------+
| Grass | Berlin |    300 |
+-------+--------+--------+
1 row in set (0.00 sec)
```

94. Give a simpler way to write this query : SELECT snum, sname city, comm FROM salespeople WHERE (comm > + 0.12 OR comm < 0.14);



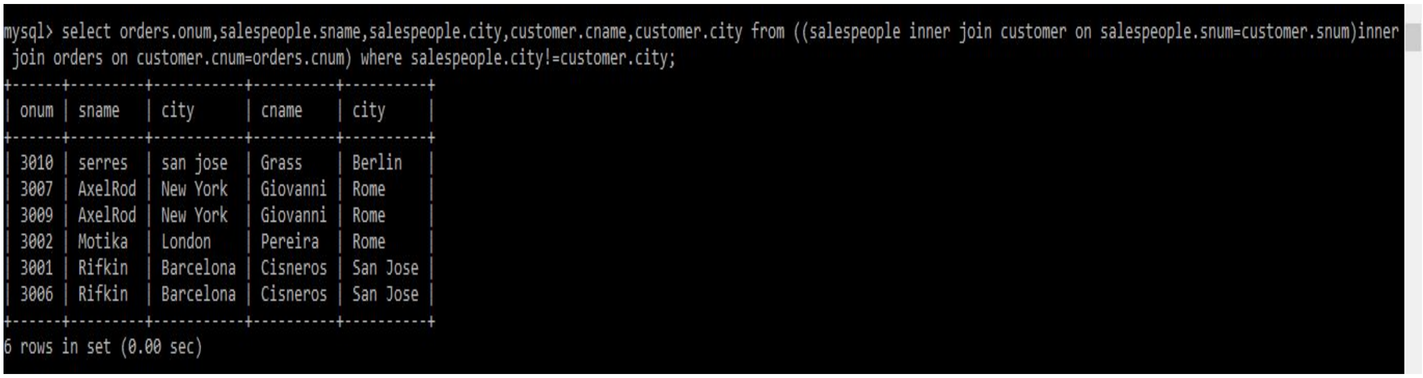
```
mysql> select snum,sname,city,comm from salespeople where (comm >0.12 or comm < 0.14);
+------+---------+-----------+------+
| snum | sname   | city      | comm |
+------+---------+-----------+------+
| 1001 | peel    | london    | 0.12 |
| 1002 | serres  | san jose  | 0.13 |
| 1003 | AxelRod | New York  | 0.10 |
| 1004 | Motika  | London    | 0.11 |
| 1005 | Fran    | London    | 0.26 |
| 1007 | Rifkin  | Barcelona | 0.15 |
+------+---------+-----------+------+
6 rows in set (0.00 sec)
```

95. Evaluate the following query : SELECT * FROM orders WHERE NOT ((odate = 10/03/96 AND snum > 1002) OR amt > 2000.00);



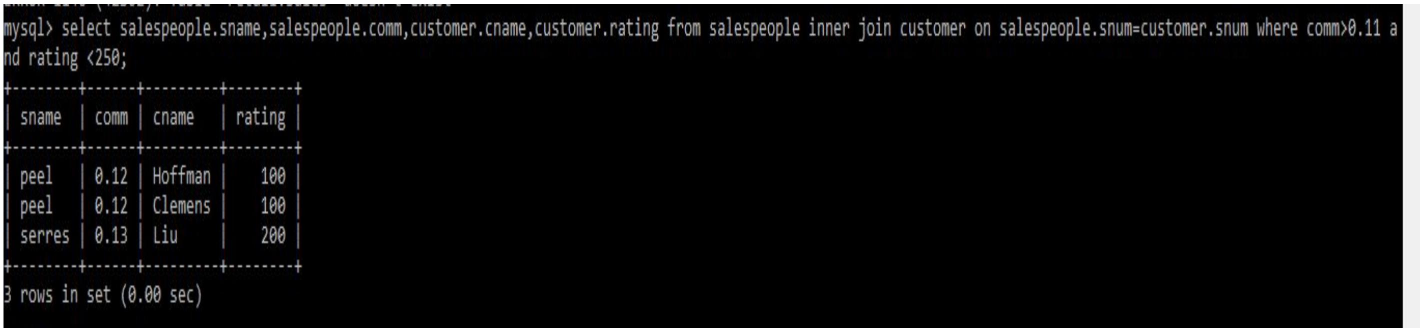
```
mysql> select orders.*from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum) where not((odate=10/03/96
 and salespeople.snum>1002) or amt >=2000);
+------+---------+------------+------+
| onum | amt     | odate      | cnum |
+------+---------+------------+------+
| 3003 |  767.19 | 1996-03-10 | 2001 |
| 3010 | 1309.95 | 1996-06-10 | 2004 |
| 3007 |   75.75 | 1996-04-10 | 2002 |
| 3009 | 1713.23 | 1996-04-10 | 2002 |
| 3002 | 1900.10 | 1996-03-10 | 2007 |
| 3001 |   18.69 | 1996-03-10 | 2008 |
| 3006 | 1098.16 | 1996-03-10 | 2008 |
+------+---------+------------+------+
7 rows in set (0.00 sec)
```

96. Which salespersons attend to customers not in the city they have been assigned to?



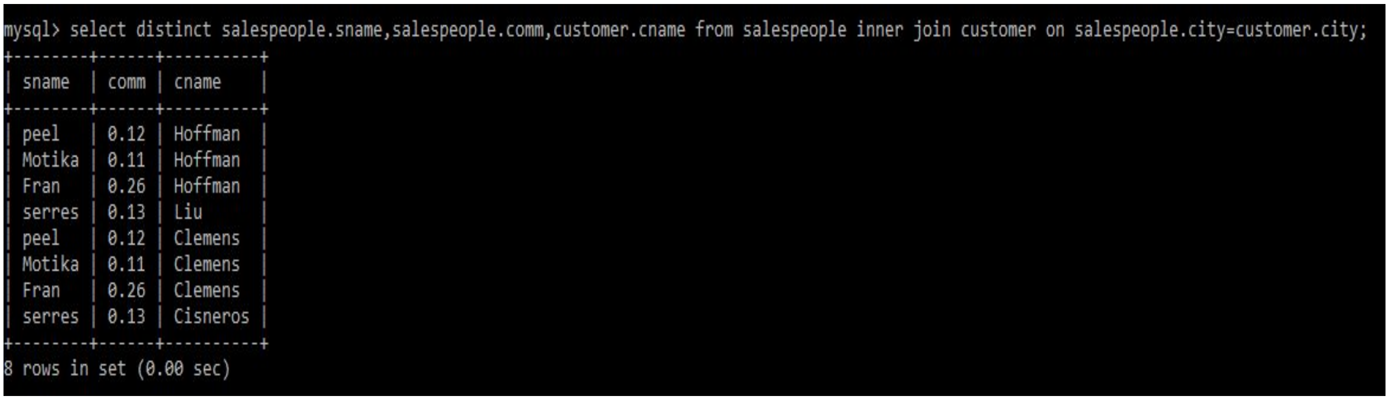
```
mysql> select orders.onum,salespeople.sname,salespeople.city,customer.cname,customer.city from ((salespeople inner join customer on salespeople.snum=customer.snum)inner
 join orders on customer.cnum=orders.cnum) where salespeople.city!=customer.city;
+------+---------+-----------+----------+----------+
| onum | sname   | city      | cname    | city     |
+------+---------+-----------+----------+----------+
| 3010 | serres  | san jose  | Grass    | Berlin   |
| 3007 | AxelRod | New York  | Giovanni | Rome     |
| 3009 | AxelRod | New York  | Giovanni | Rome     |
| 3002 | Motika  | London    | Pereira  | Rome     |
| 3001 | Rifkin  | Barcelona | Cisneros | San Jose |
| 3006 | Rifkin  | Barcelona | Cisneros | San Jose |
+------+---------+-----------+----------+----------+
6 rows in set (0.00 sec)
```

97. Which salespeople get commission greater than 0.11 are serving customers rated less than 250?



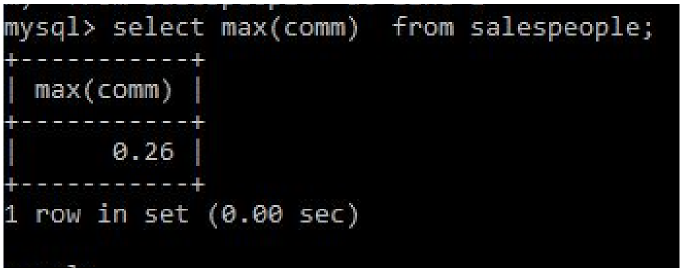
```
mysql> select salespeople.sname,salespeople.comm,customer.cname,customer.rating from salespeople inner join customer on salespeople.snum=customer.snum where comm>0.11 a
nd rating <250;
+--------+------+---------+--------+
| sname  | comm | cname   | rating |
+--------+------+---------+--------+
| peel   | 0.12 | Hoffman |    100 |
| peel   | 0.12 | Clemens |    100 |
| serres | 0.13 | Liu     |    200 |
+--------+------+---------+--------+
3 rows in set (0.00 sec)
```

98. Which salespeople have been assigned to the same city but get different commission percentages?



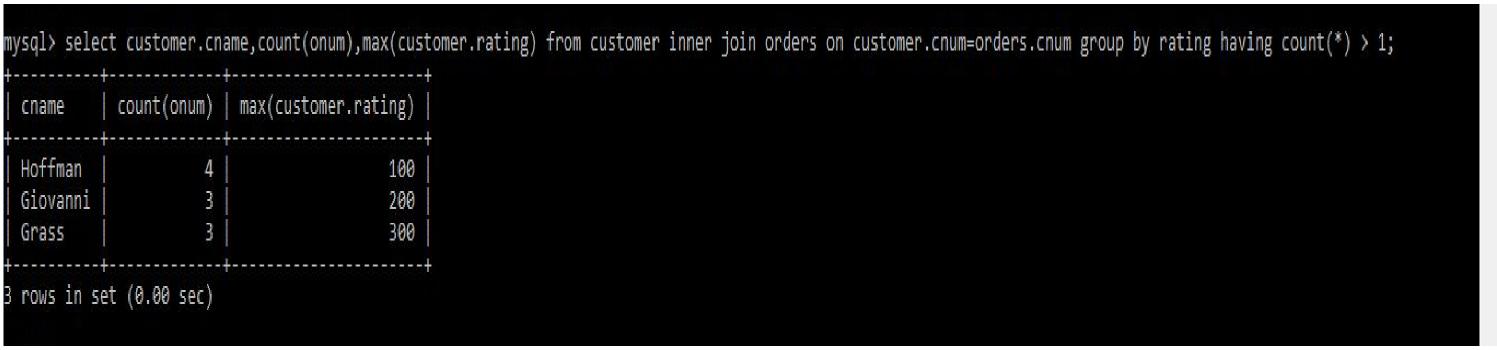
```
mysql> select distinct salespeople.sname,salespeople.comm,customer.cname from salespeople inner join customer on salespeople.city=customer.city;
+--------+------+----------+
| sname  | comm | cname    |
+--------+------+----------+
| peel   | 0.12 | Hoffman  |
| Motika | 0.11 | Hoffman  |
| Fran   | 0.26 | Hoffman  |
| serres | 0.13 | Liu      |
| peel   | 0.12 | Clemens  |
| Motika | 0.11 | Clemens  |
| Fran   | 0.26 | Clemens  |
| serres | 0.13 | Cisneros |
+--------+------+----------+
8 rows in set (0.00 sec)
```

99. Which salesperson has earned the most by way of commission?



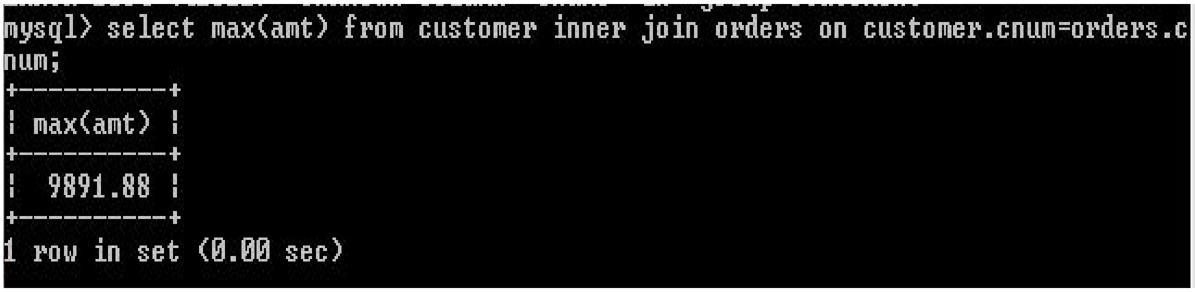
```
mysql> select max(comm)  from salespeople;
+-----------+
| max(comm) |
+-----------+
|      0.26 |
+-----------+
1 row in set (0.00 sec)
```

100.Does the customer who has placed the maximum number of orders have the maximum rating?



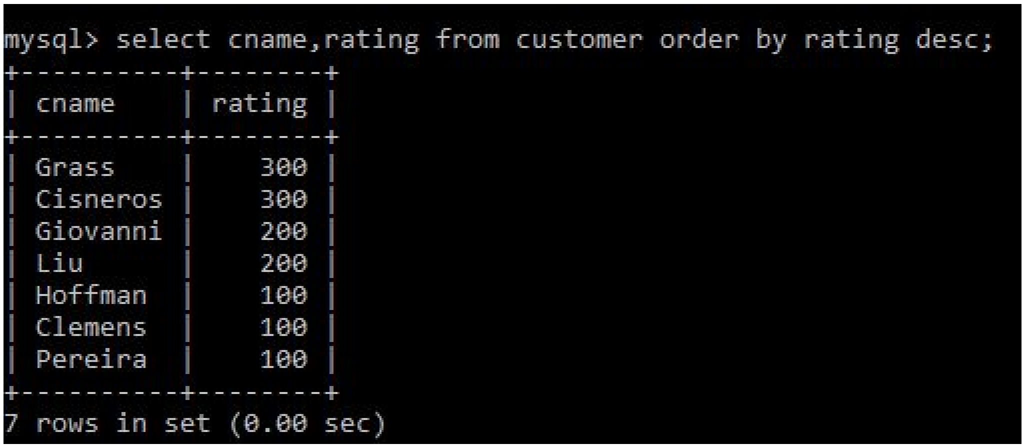
```
mysql> select customer.cname,count(onum),max(customer.rating) from customer inner join orders on customer.cnum=orders.cnum group by rating having count(*) > 1;
+----------+-------------+----------------------+
| cname    | count(onum) | max(customer.rating) |
+----------+-------------+----------------------+
| Hoffman  |           4 |                  100 |
| Giovanni |           3 |                  200 |
| Grass    |           3 |                  300 |
+----------+-------------+----------------------+
3 rows in set (0.00 sec)
```

101.Has the customer who has spent the largest amount of money been given the highest rating?



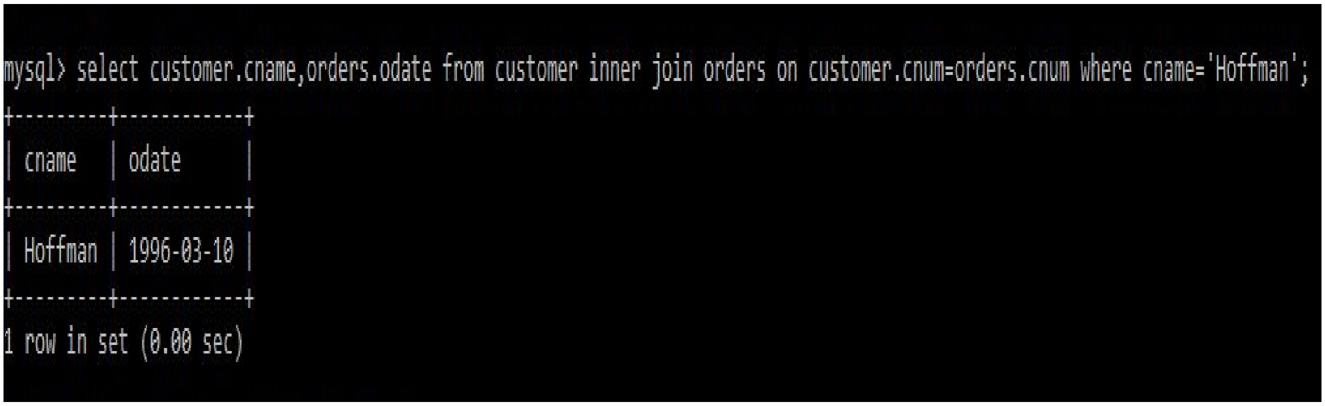
```
mysql> select max(amt) from customer inner join orders on customer.cnum=orders.c
num;
+-----------+
| max(amt) |
+-----------+
|  9891.88 |
+-----------+
1 row in set (0.00 sec)
```

102.List all customers in descending order of customer rating.



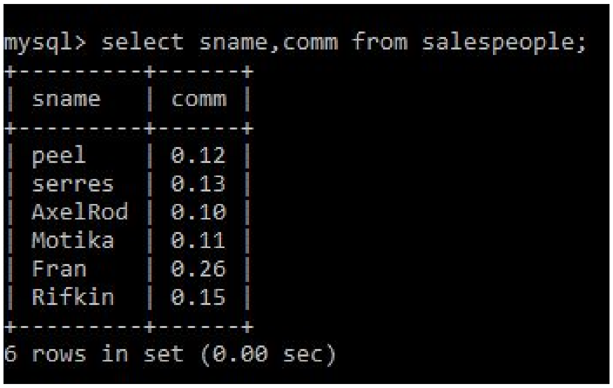
```
mysql> select cname,rating from customer order by rating desc;
+----------+--------+
| cname    | rating |
+----------+--------+
| Grass    |    300 |
| Cisneros |    300 |
| Giovanni |    200 |
| Liu      |    200 |
| Hoffman  |    100 |
| Clemens  |    100 |
| Pereira  |    100 |
+----------+--------+
7 rows in set (0.00 sec)
```

103.On which days has Hoffman placed orders?



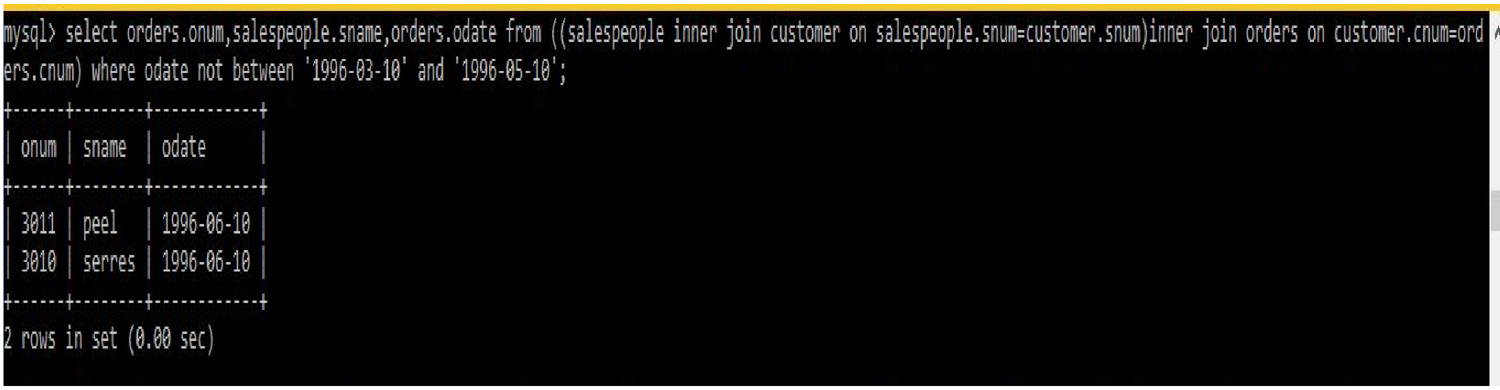
```
mysql> select customer.cname,orders.odate from customer inner join orders on customer.cnum=orders.cnum where cname='Hoffman';
+---------+------------+
| cname   | odate      |
+---------+------------+
| Hoffman | 1996-03-10 |
+---------+------------+
1 row in set (0.00 sec)
```

104.Do all salespeople have different commissions?



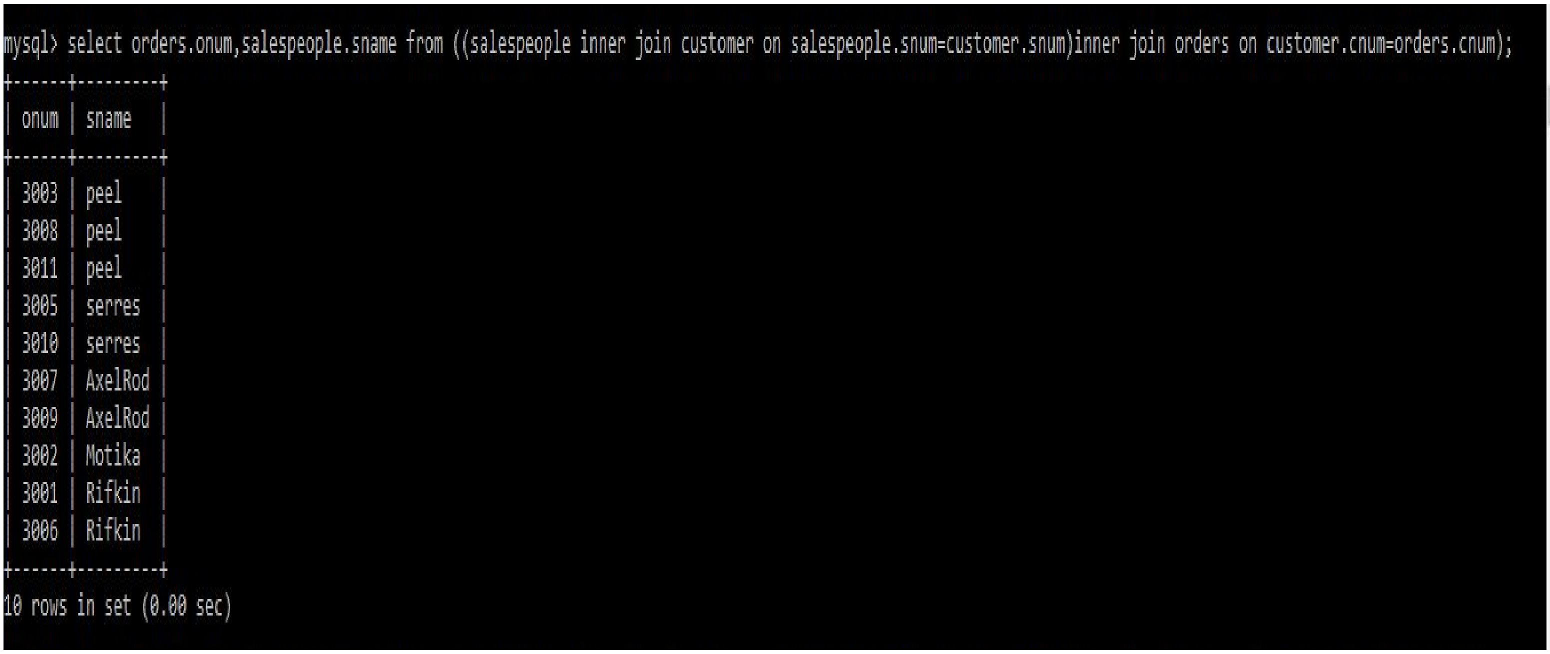
```
mysql> select sname,comm from salespeople;
+---------+------+
| sname   | comm |
+---------+------+
| peel    | 0.12 |
| serres  | 0.13 |
| AxelRod | 0.10 |
| Motika  | 0.11 |
| Fran    | 0.26 |
| Rifkin  | 0.15 |
+---------+------+
6 rows in set (0.00 sec)
```

105.Which salespeople have no orders between 10/03/1996 and 10/05/1996?



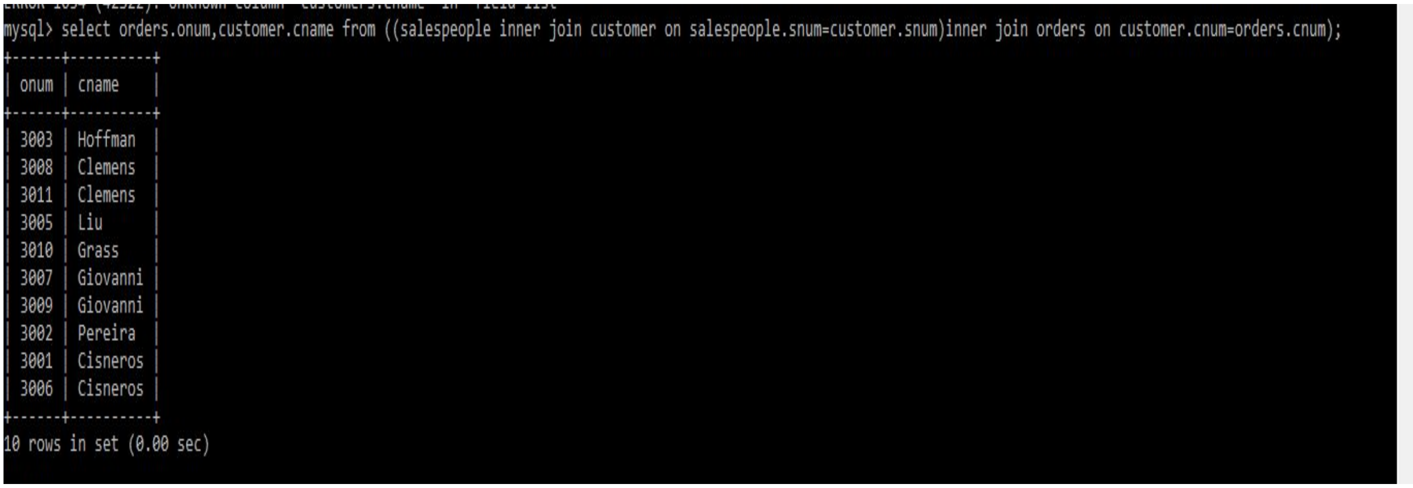
```
mysql> select orders.onum,salespeople.sname,orders.odate from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=ord
ers.cnum) where odate not between '1996-03-10' and '1996-05-10';
+------+--------+------------+
| onum | sname  | odate      |
+------+--------+------------+
| 3011 | peel   | 1996-06-10 |
| 3010 | serres | 1996-06-10 |
+------+--------+------------+
2 rows in set (0.00 sec)
```

106.How many salespersons have succeeded in getting orders?



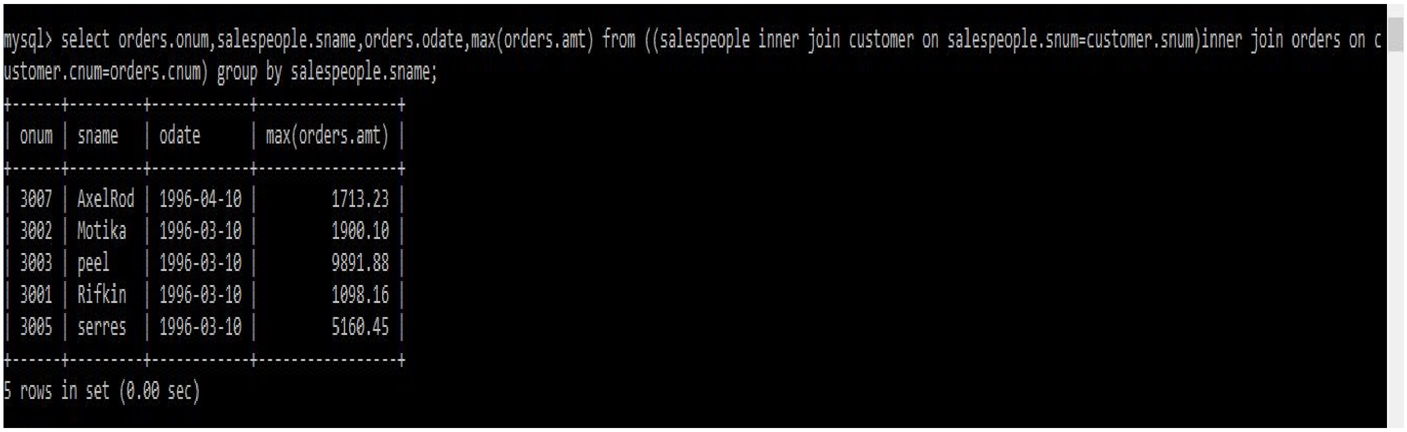
```
mysql> select orders.onum,salespeople.sname from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum);
+------+---------+
| onum | sname   |
+------+---------+
| 3003 | peel    |
| 3008 | peel    |
| 3011 | peel    |
| 3005 | serres  |
| 3010 | serres  |
| 3007 | AxelRod |
| 3009 | AxelRod |
| 3002 | Motika  |
| 3001 | Rifkin  |
| 3006 | Rifkin  |
+------+---------+
10 rows in set (0.00 sec)
```

107.How many customers have placed orders?



```
ERROR 1054 (42S22): Unknown column 'customers.cname' in 'field list'
mysql> select orders.onum,customer.cname from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum);
+------+---------+
| onum | cname   |
+------+---------+
| 3003 | Hoffman |
| 3008 | Clemens |
| 3011 | Clemens |
| 3005 | Liu     |
| 3010 | Grass   |
| 3007 | Giovanni |
| 3009 | Giovanni |
| 3002 | Pereira |
| 3001 | Cisneros |
| 3006 | Cisneros |
+------+---------+
10 rows in set (0.00 sec)
```

108.On which date has each salesperson booked an order of maximum value?



```
mysql> select orders.onum,salespeople.sname,orders.odate,max(orders.amt) from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on c
ustomer.cnum=orders.cnum) group by salespeople.sname;
+------+--------+------------+-----------------+
| onum | sname  | odate      | max(orders.amt) |
+------+--------+------------+-----------------+
| 3007 | AxelRod | 1996-04-10 |        1713.23 |
| 3002 | Motika | 1996-03-10 |         1900.10 |
| 3003 | peel   | 1996-03-10 |         9891.88 |
| 3001 | Rifkin | 1996-03-10 |         1098.16 |
| 3005 | serres | 1996-03-10 |         5160.45 |
+------+--------+------------+-----------------+
5 rows in set (0.00 sec)
```

109.Who is the most successful salesperson?

Ans: Peel

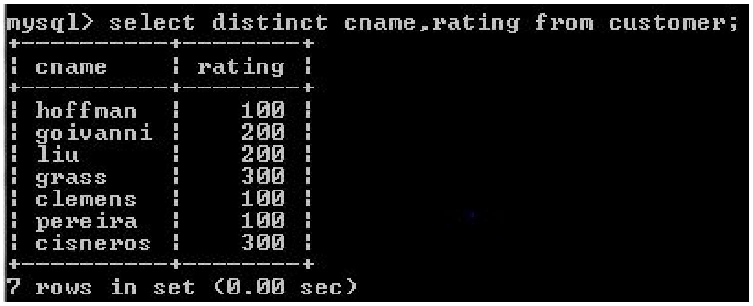
110.Who is the worst customer with respect to the company?

Ans: Hoffman has the lowest value in the amount column.

111.Are all customers not having placed orders greater than 200 totally been serviced by salespersons Peel or Serres?

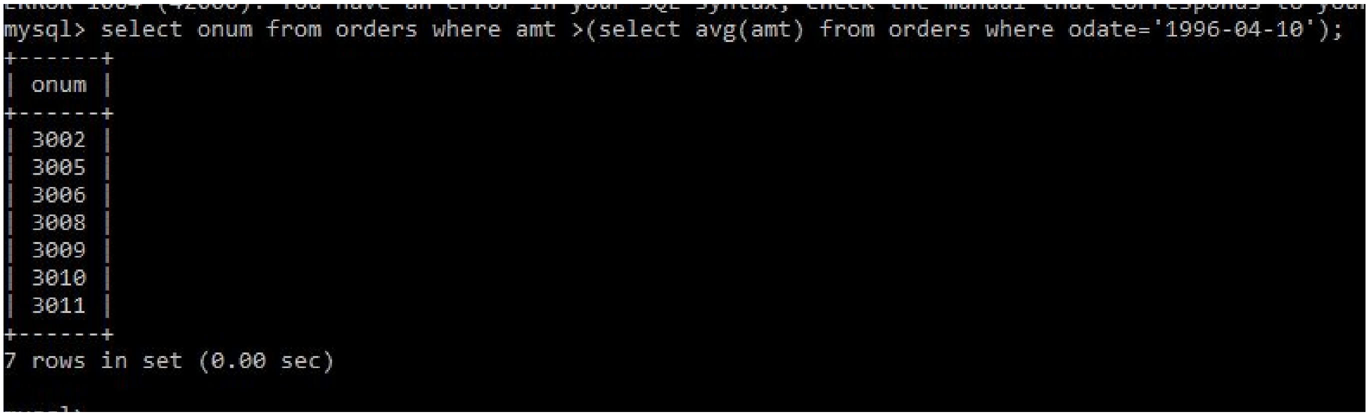
Ans: No,Peel and Serres have served customers who have placed order greater than 200.

112.Which customers have the same rating?



```
mysql> select distinct cname,rating from customer;
+----------+--------+
| cname    | rating |
+----------+--------+
| hoffman  |    100 |
| goivanni |    200 |
| liu      |    200 |
| grass    |    300 |
| clemens  |    100 |
| pereira  |    100 |
| cisneros |    300 |
+----------+--------+
7 rows in set (0.00 sec)
```

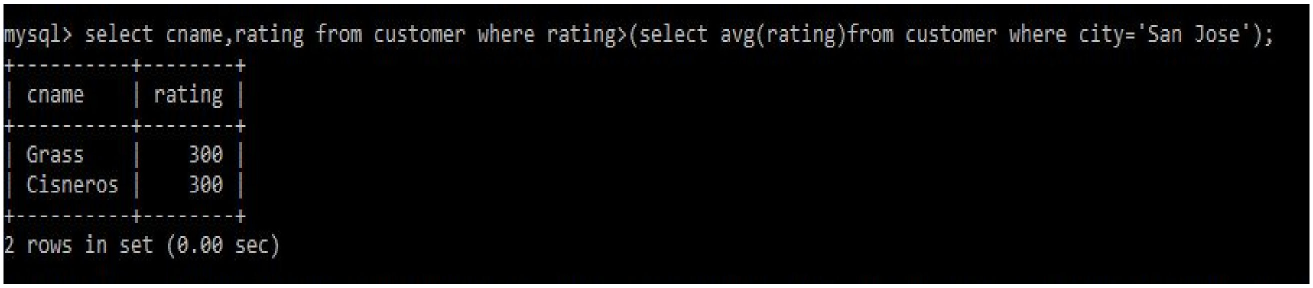
113.Find all orders greater than the average for October 4th.



```
ERROR 1064 (42000): You have an error in your SQL syntax; check the manual that corresponds to your
mysql> select onum from orders where amt >(select avg(amt) from orders where odate='1996-04-10');
+------+
| onum |
+------+
| 3002 |
| 3005 |
| 3006 |
| 3008 |
| 3009 |
| 3010 |
| 3011 |
+------+
7 rows in set (0.00 sec)

mysql>
```

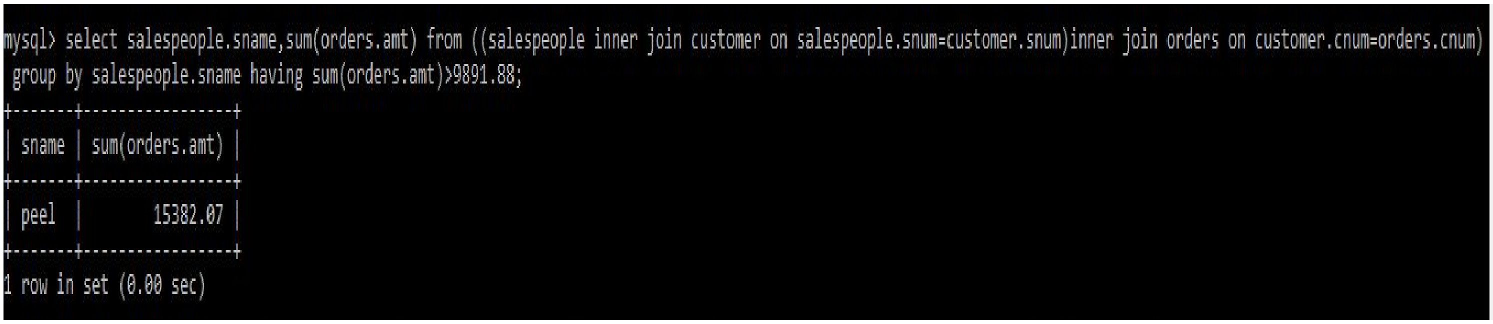
114.Which customers have above average orders?

115.List all customers with ratings above San Jose’s average.



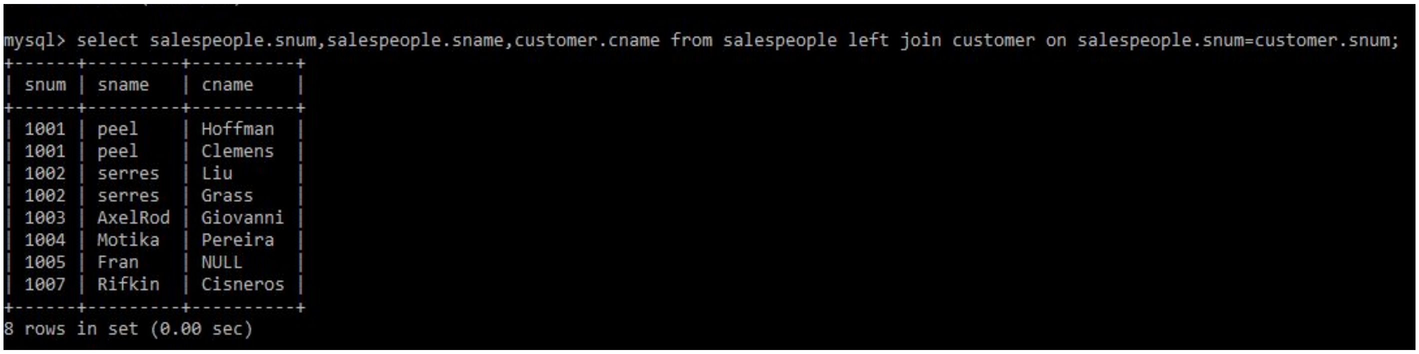
```
mysql> select cname,rating from customer where rating>(select avg(rating)from customer where city='San Jose');
+----------+--------+
| cname    | rating |
+----------+--------+
| Grass    |    300 |
| Cisneros |    300 |
+----------+--------+
2 rows in set (0.00 sec)
```

116.Select the total amount in orders for each salesperson for whom the total is greater than the amount of the largest order in the table.



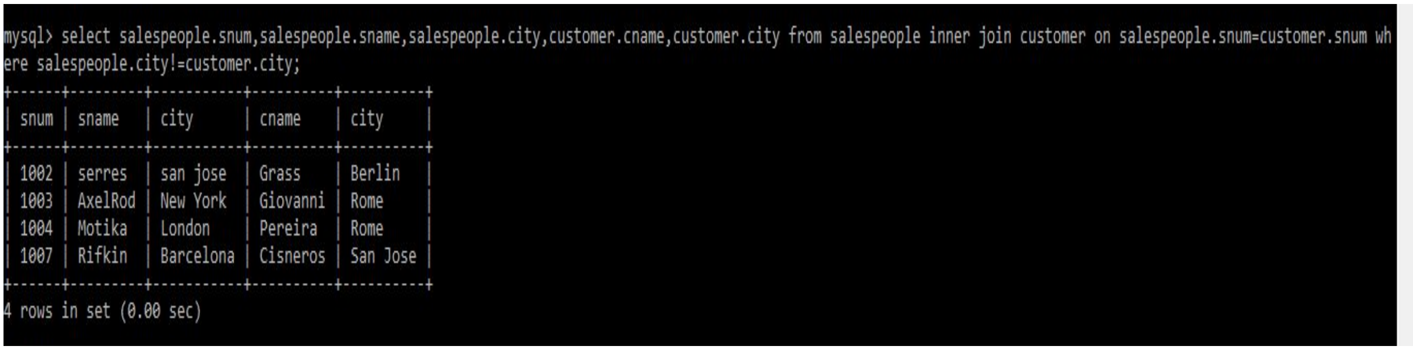
```
mysql> select salespeople.sname,sum(orders.amt) from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on customer.cnum=orders.cnum)
 group by salespeople.sname having sum(orders.amt)>9891.88;
+-------+-----------------+
| sname | sum(orders.amt) |
+-------+-----------------+
| peel  |        15382.07 |
+-------+-----------------+
1 row in set (0.00 sec)
```

117.Give names and numbers of all salespersons who have more than one customer.



```
mysql> select salespeople.snum,salespeople.sname,customer.cname from salespeople left join customer on salespeople.snum=customer.snum;
+------+---------+----------+
| snum | sname   | cname    |
+------+---------+----------+
| 1001 | peel    | Hoffman  |
| 1001 | peel    | Clemens  |
| 1002 | serres  | Liu      |
| 1002 | serres  | Grass    |
| 1003 | AxelRod | Giovanni |
| 1004 | Motika  | Pereira  |
| 1005 | Fran    | NULL     |
| 1007 | Rifkin  | Cisneros |
+------+---------+----------+
8 rows in set (0.00 sec)
```

118.Select all salespersons by name and number who have customers in their city whom they don't service.



```
mysql> select salespeople.snum,salespeople.sname,salespeople.city,customer.cname,customer.city from salespeople inner join customer on salespeople.snum=customer.snum wh
ere salespeople.city!=customer.city;
+------+---------+-----------+----------+----------+
| snum | sname   | city      | cname    | city     |
+------+---------+-----------+----------+----------+
| 1002 | serres  | san jose  | Grass    | Berlin   |
| 1003 | AxelRod | New York  | Giovanni | Rome     |
| 1004 | Motika  | London    | Pereira  | Rome     |
| 1007 | Rifkin  | Barcelona | Cisneros | San Jose |
+------+---------+-----------+----------+----------+
4 rows in set (0.00 sec)
```

119.Which customers' rating should be lowered?

Ans: rating of Grass can be lowered.

120.Is there a case for assigning a salesperson to Berlin?

Ans: No

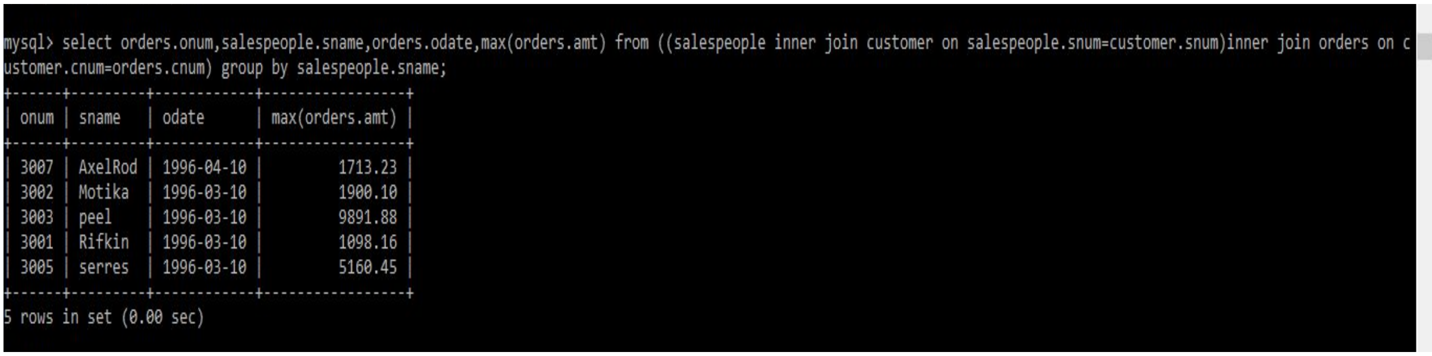
121.Is there any evidence linking the performance of a salesperson to the commission that he or she is being paid?

Ans: No

122.Does the total amount in orders by customer in Rome and London exceed the commission paid to salespersons in London and New York by more than 5 times?

Ans: NO

123.Which is the date, order number, amt and city for each salesperson (by name) for the maximum order he has obtained?

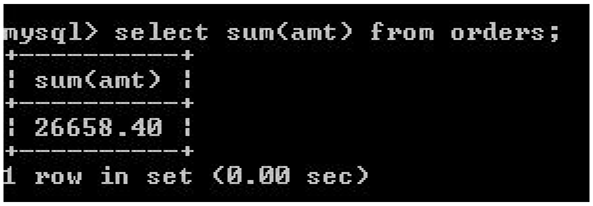


```
mysql> select orders.onum,salespeople.sname,orders.odate,max(orders.amt) from ((salespeople inner join customer on salespeople.snum=customer.snum)inner join orders on c
ustomer.cnum=orders.cnum) group by salespeople.sname;
+------+---------+------------+-----------------+
| onum | sname   | odate      | max(orders.amt) |
+------+---------+------------+-----------------+
| 3007 | AxelRod | 1996-04-10 |         1713.23 |
| 3002 | Motika  | 1996-03-10 |         1900.10 |
| 3003 | peel    | 1996-03-10 |         9891.88 |
| 3001 | Rifkin  | 1996-03-10 |         1098.16 |
| 3005 | serres  | 1996-03-10 |         5160.45 |
+------+---------+------------+-----------------+
5 rows in set (0.00 sec)
```

124.Which salesperson(s) should be fired?

Ans: Fran should be fired from the company.

125.What is the total income for the company?



```
mysql> select sum(amt) from orders;
+-----------+
| sum(amt) |
+-----------+
| 26658.40 |
+-----------+
1 row in set (0.00 sec)
```