Khoi Duong

Prof. VB

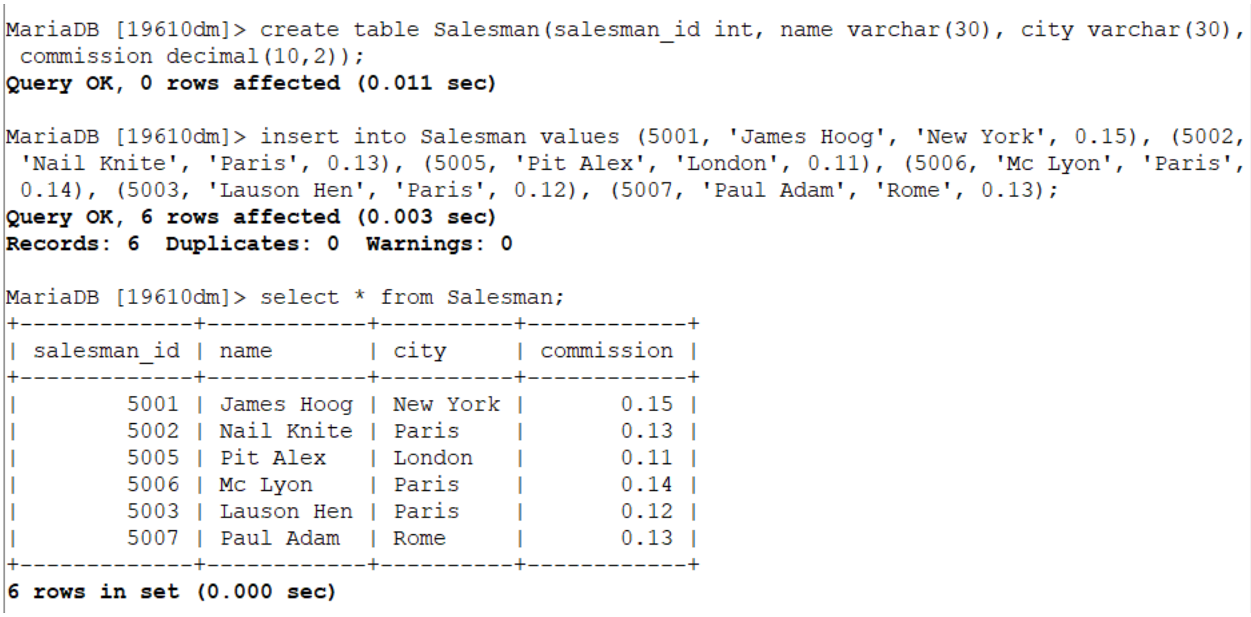
CS457L

12/17/2022

FINAL

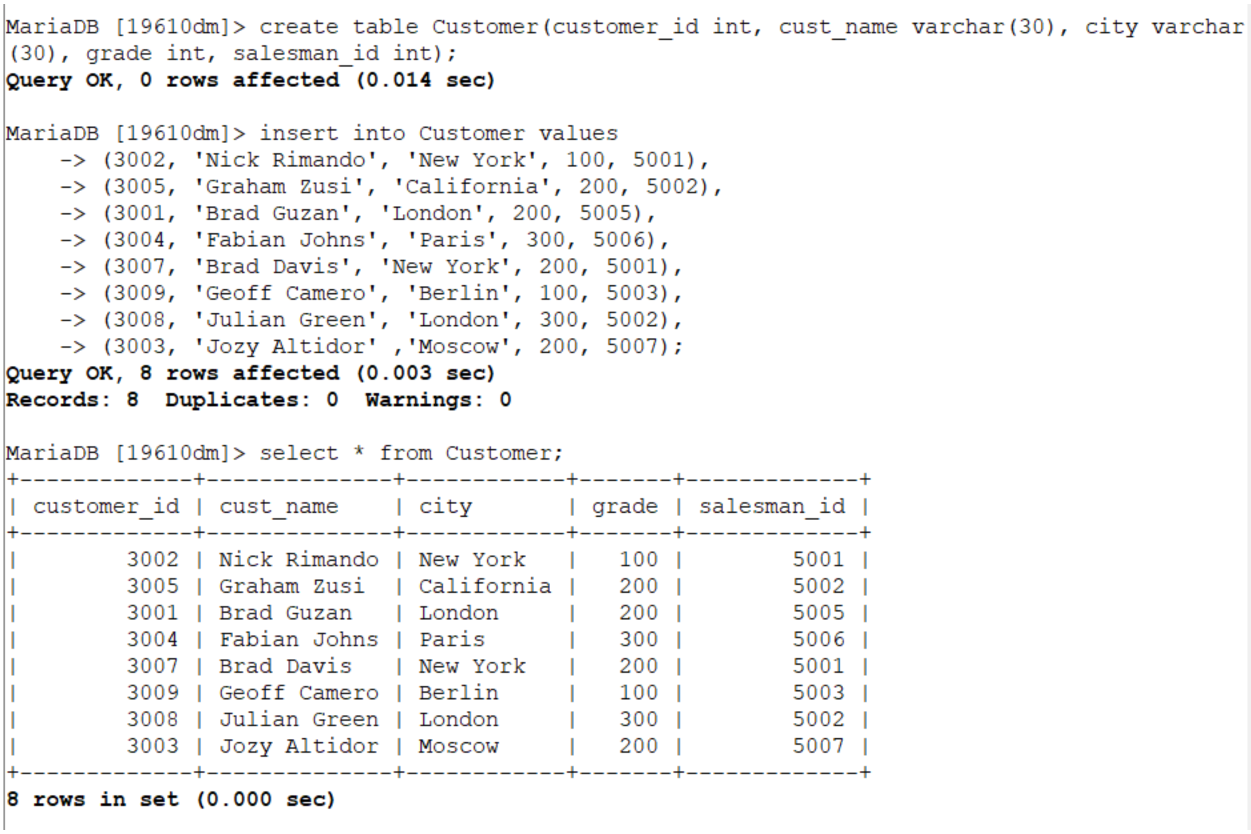
1.

Create table Salesman:



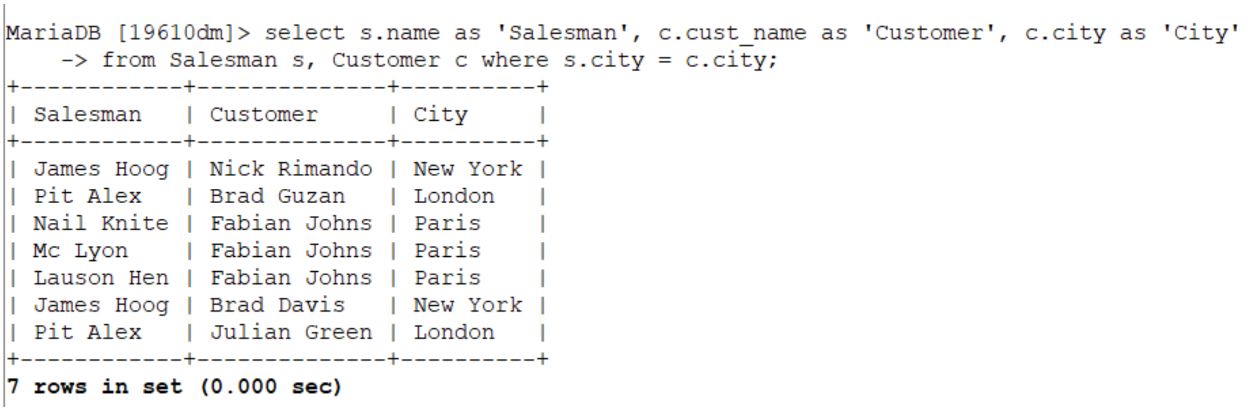
```
MariaDB [19610dm]> create table Salesman(salesman_id int, name varchar(30), city varchar(30),
 commission decimal(10,2));
Query OK, 0 rows affected (0.011 sec)

MariaDB [19610dm]> insert into Salesman values (5001, 'James Hoog', 'New York', 0.15), (5002,
 'Nail Knite', 'Paris', 0.13), (5005, 'Pit Alex', 'London', 0.11), (5006, 'Mc Lyon', 'Paris',
 0.14), (5003, 'Lauson Hen', 'Paris', 0.12), (5007, 'Paul Adam', 'Rome', 0.13);
Query OK, 6 rows affected (0.003 sec)
Records: 6  Duplicates: 0  Warnings: 0

MariaDB [19610dm]> select * from Salesman;
+-------------+------------+----------+------------+
| salesman_id | name       | city     | commission |
+-------------+------------+----------+------------+
|        5001 | James Hoog | New York |       0.15 |
|        5002 | Nail Knite | Paris    |       0.13 |
|        5005 | Pit Alex   | London   |       0.11 |
|        5006 | Mc Lyon    | Paris    |       0.14 |
|        5003 | Lauson Hen | Paris    |       0.12 |
|        5007 | Paul Adam  | Rome     |       0.13 |
+-------------+------------+----------+------------+
6 rows in set (0.000 sec)
```

Create table Customer:



```
MariaDB [19610dm]> create table Customer(customer_id int, cust_name varchar(30), city varchar
(30), grade int, salesman_id int);
Query OK, 0 rows affected (0.014 sec)

MariaDB [19610dm]> insert into Customer values
    -> (3002, 'Nick Rimando', 'New York', 100, 5001),
    -> (3005, 'Graham Zusi', 'California', 200, 5002),
    -> (3001, 'Brad Guzan', 'London', 200, 5005),
    -> (3004, 'Fabian Johns', 'Paris', 300, 5006),
    -> (3007, 'Brad Davis', 'New York', 200, 5001),
    -> (3009, 'Geoff Camero', 'Berlin', 100, 5003),
    -> (3008, 'Julian Green', 'London', 300, 5002),
    -> (3003, 'Jozy Altidor' ,'Moscow', 200, 5007);
Query OK, 8 rows affected (0.003 sec)
Records: 8  Duplicates: 0  Warnings: 0

MariaDB [19610dm]> select * from Customer;
+-------------+--------------+------------+-------+-------------+
| customer_id | cust_name    | city       | grade | salesman_id |
+-------------+--------------+------------+-------+-------------+
|        3002 | Nick Rimando | New York   |   100 |        5001 |
|        3005 | Graham Zusi  | California |   200 |        5002 |
|        3001 | Brad Guzan   | London     |   200 |        5005 |
|        3004 | Fabian Johns | Paris      |   300 |        5006 |
|        3007 | Brad Davis   | New York   |   200 |        5001 |
|        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        3008 | Julian Green | London     |   300 |        5002 |
|        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
+-------------+--------------+------------+-------+-------------+
8 rows in set (0.000 sec)
```

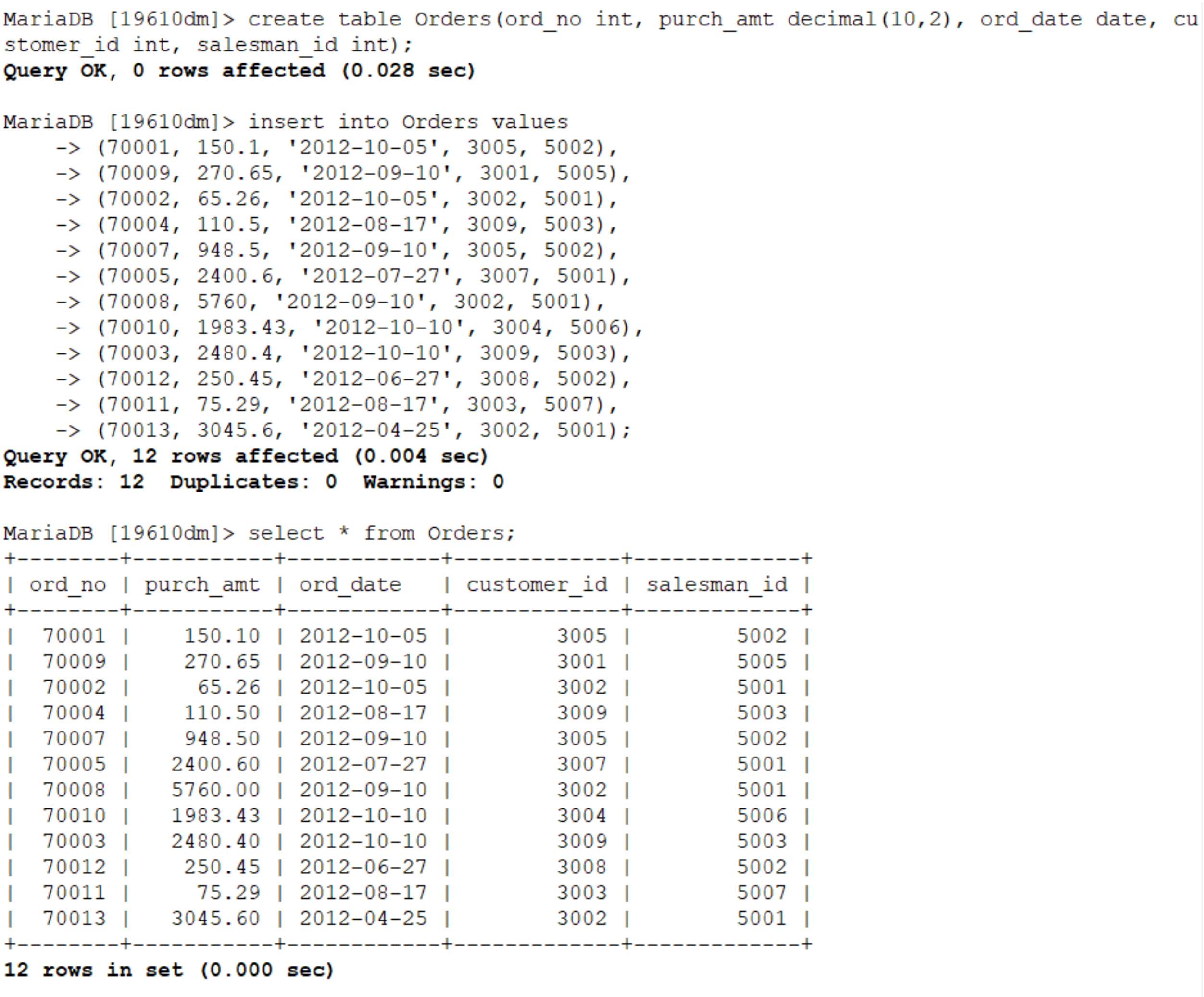
Write a SQL statement to prepare a list with the salesman's name, customer name, and their cities for the salesmen and customer who belongs to the same city.



```
MariaDB [19610dm]> select s.name as 'Salesman', c.cust_name as 'Customer', c.city as 'City'
    -> from Salesman s, Customer c where s.city = c.city;
+------------+--------------+----------+
| Salesman   | Customer     | City     |
+------------+--------------+----------+
| James Hoog | Nick Rimando | New York |
| Pit Alex   | Brad Guzan   | London   |
| Nail Knite | Fabian Johns | Paris    |
| Mc Lyon    | Fabian Johns | Paris    |
| Lauson Hen | Fabian Johns | Paris    |
| James Hoog | Brad Davis   | New York |
| Pit Alex   | Julian Green | London   |
+------------+--------------+----------+
7 rows in set (0.000 sec)
```

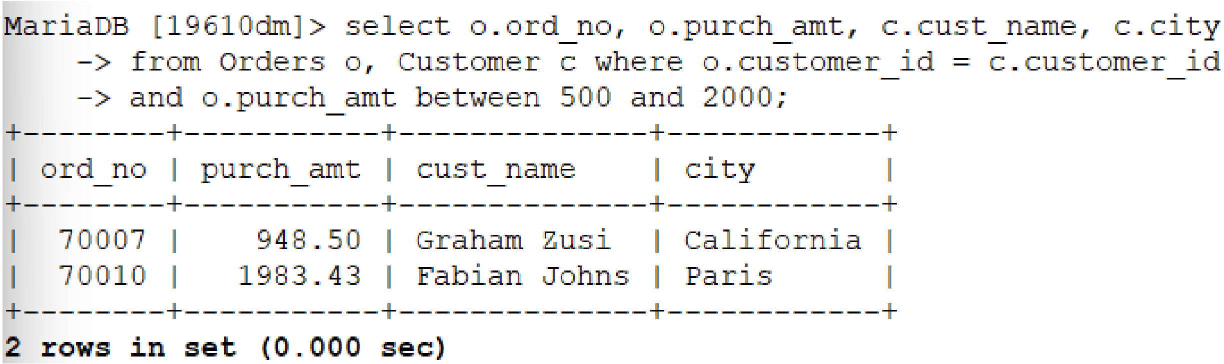
2.

Create table Orders:



```
MariaDB [19610dm]> create table Orders(ord_no int, purch_amt decimal(10,2), ord_date date, cu
stomer_id int, salesman_id int);
Query OK, 0 rows affected (0.028 sec)

MariaDB [19610dm]> insert into Orders values
    -> (70001, 150.1, '2012-10-05', 3005, 5002),
    -> (70009, 270.65, '2012-09-10', 3001, 5005),
    -> (70002, 65.26, '2012-10-05', 3002, 5001),
    -> (70004, 110.5, '2012-08-17', 3009, 5003),
    -> (70007, 948.5, '2012-09-10', 3005, 5002),
    -> (70005, 2400.6, '2012-07-27', 3007, 5001),
    -> (70008, 5760, '2012-09-10', 3002, 5001),
    -> (70010, 1983.43, '2012-10-10', 3004, 5006),
    -> (70003, 2480.4, '2012-10-10', 3009, 5003),
    -> (70012, 250.45, '2012-06-27', 3008, 5002),
    -> (70011, 75.29, '2012-08-17', 3003, 5007),
    -> (70013, 3045.6, '2012-04-25', 3002, 5001);
Query OK, 12 rows affected (0.004 sec)
Records: 12  Duplicates: 0  Warnings: 0

MariaDB [19610dm]> select * from Orders;
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
|  70001 |    150.10 | 2012-10-05 |        3005 |        5002 |
|  70009 |    270.65 | 2012-09-10 |        3001 |        5005 |
|  70002 |     65.26 | 2012-10-05 |        3002 |        5001 |
|  70004 |    110.50 | 2012-08-17 |        3009 |        5003 |
|  70007 |    948.50 | 2012-09-10 |        3005 |        5002 |
|  70005 |   2400.60 | 2012-07-27 |        3007 |        5001 |
|  70008 |   5760.00 | 2012-09-10 |        3002 |        5001 |
|  70010 |   1983.43 | 2012-10-10 |        3004 |        5006 |
|  70003 |   2480.40 | 2012-10-10 |        3009 |        5003 |
|  70012 |    250.45 | 2012-06-27 |        3008 |        5002 |
|  70011 |     75.29 | 2012-08-17 |        3003 |        5007 |
|  70013 |   3045.60 | 2012-04-25 |        3002 |        5001 |
+--------+-----------+------------+-------------+-------------+
12 rows in set (0.000 sec)
```

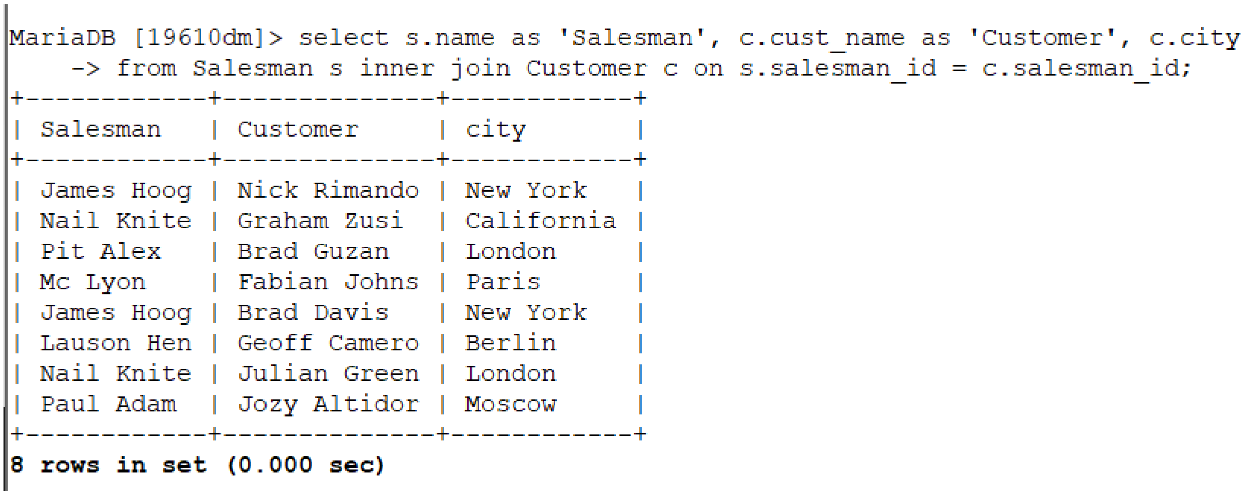
Write a SQL statement to make a list with the order number, purchase amount, customer name, and cities for orders that order between 500 and 2000.



```
MariaDB [19610dm]> select o.ord_no, o.purch_amt, c.cust_name, c.city
    -> from Orders o, Customer c where o.customer_id = c.customer_id
    -> and o.purch_amt between 500 and 2000;
+--------+-----------+--------------+------------+
| ord_no | purch_amt | cust_name    | city       |
+--------+-----------+--------------+------------+
|  70007 |    948.50 | Graham Zusi  | California |
|  70010 |   1983.43 | Fabian Johns | Paris      |
+--------+-----------+--------------+------------+
2 rows in set (0.000 sec)
```

3.

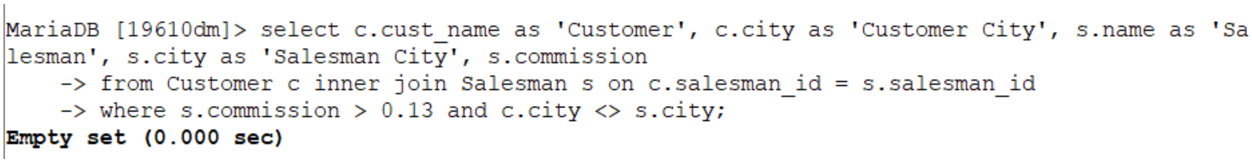
Write a SQL statement to know which salesman is working for which customer.



```
MariaDB [19610dm]> select s.name as 'Salesman', c.cust_name as 'Customer', c.city
    -> from Salesman s inner join Customer c on s.salesman_id = c.salesman_id;
+------------+--------------+------------+
| Salesman   | Customer     | city       |
+------------+--------------+------------+
| James Hoog | Nick Rimando | New York   |
| Nail Knite | Graham Zusi  | California |
| Pit Alex   | Brad Guzan   | London     |
| Mc Lyon    | Fabian Johns | Paris      |
| James Hoog | Brad Davis   | New York   |
| Lauson Hen | Geoff Camero | Berlin     |
| Nail Knite | Julian Green | London     |
| Paul Adam  | Jozy Altidor | Moscow     |
+------------+--------------+------------+
8 rows in set (0.000 sec)
```

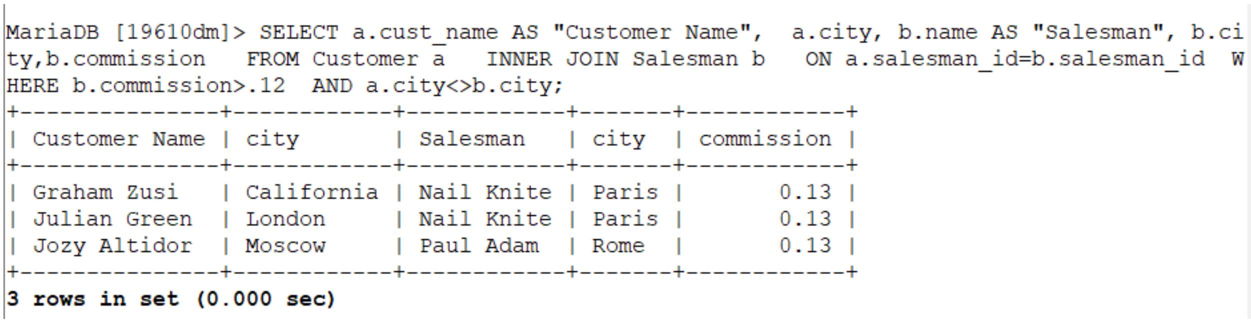
4.

Write a SQL statement to find the list of customers who appointed a salesman for their jobs who does not live in the same city where their customer lives, and gets a commission is above 13%.



```
MariaDB [19610dm]> select c.cust_name as 'Customer', c.city as 'Customer City', s.name as 'Sa
lesman', s.city as 'Salesman City', s.commission
    -> from Customer c inner join Salesman s on c.salesman_id = s.salesman_id
    -> where s.commission > 0.13 and c.city <> s.city;
Empty set (0.000 sec)
```

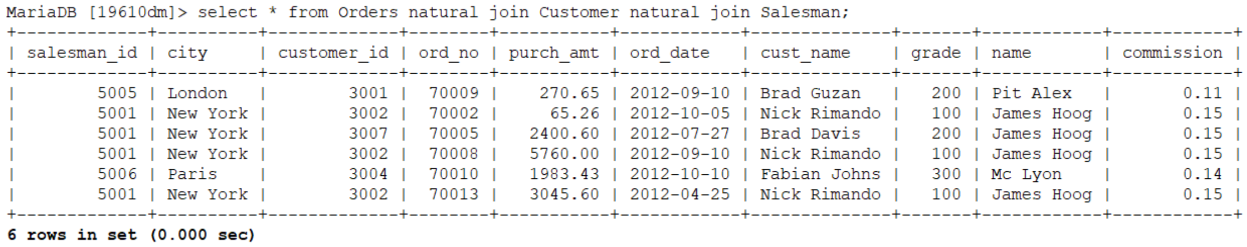
If the commission is set to be above 12%, we will have the below result:



```
MariaDB [19610dm]> SELECT a.cust_name AS "Customer Name",  a.city, b.name AS "Salesman", b.ci
ty,b.commission   FROM Customer a   INNER JOIN Salesman b   ON a.salesman_id=b.salesman_id  W
HERE b.commission>.12  AND a.city<>b.city;
+---------------+------------+------------+-------+------------+
| Customer Name | city       | Salesman   | city  | commission |
+---------------+------------+------------+-------+------------+
| Graham Zusi   | California | Nail Knite | Paris |       0.13 |
| Julian Green  | London     | Nail Knite | Paris |       0.13 |
| Jozy Altidor  | Moscow     | Paul Adam  | Rome  |       0.13 |
+---------------+------------+------------+-------+------------+
3 rows in set (0.000 sec)
```

5.

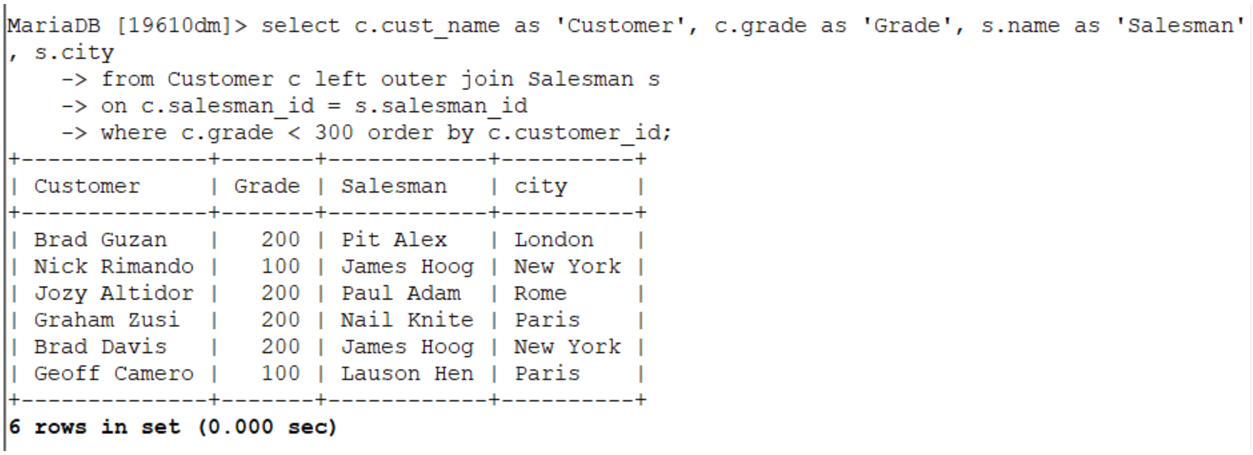
Write a SQL statement to make a join on the tables salesman, customer and orders in such a form that the same column of each table will appear once and only the relational rows will come.



```
MariaDB [19610dm]> select * from Orders natural join Customer natural join Salesman;
+-------------+----------+-------------+--------+-----------+------------+--------------+-------+------------+------------+
| salesman_id | city     | customer_id | ord_no | purch_amt | ord_date   | cust_name    | grade | name       | commission |
+-------------+----------+-------------+--------+-----------+------------+--------------+-------+------------+------------+
|        5005 | London   |        3001 |  70009 |    270.65 | 2012-09-10 | Brad Guzan   |   200 | Pit Alex   |       0.11 |
|        5001 | New York |        3002 |  70002 |     65.26 | 2012-10-05 | Nick Rimando |   100 | James Hoog |       0.15 |
|        5001 | New York |        3007 |  70005 |   2400.60 | 2012-07-27 | Brad Davis   |   200 | James Hoog |       0.15 |
|        5001 | New York |        3002 |  70008 |   5760.00 | 2012-09-10 | Nick Rimando |   100 | James Hoog |       0.15 |
|        5006 | Paris    |        3004 |  70010 |   1983.43 | 2012-10-10 | Fabian Johns |   300 | Mc Lyon    |       0.14 |
|        5001 | New York |        3002 |  70013 |   3045.60 | 2012-04-25 | Nick Rimando |   100 | James Hoog |       0.15 |
+-------------+----------+-------------+--------+-----------+------------+--------------+-------+------------+------------+
6 rows in set (0.000 sec)
```

6.

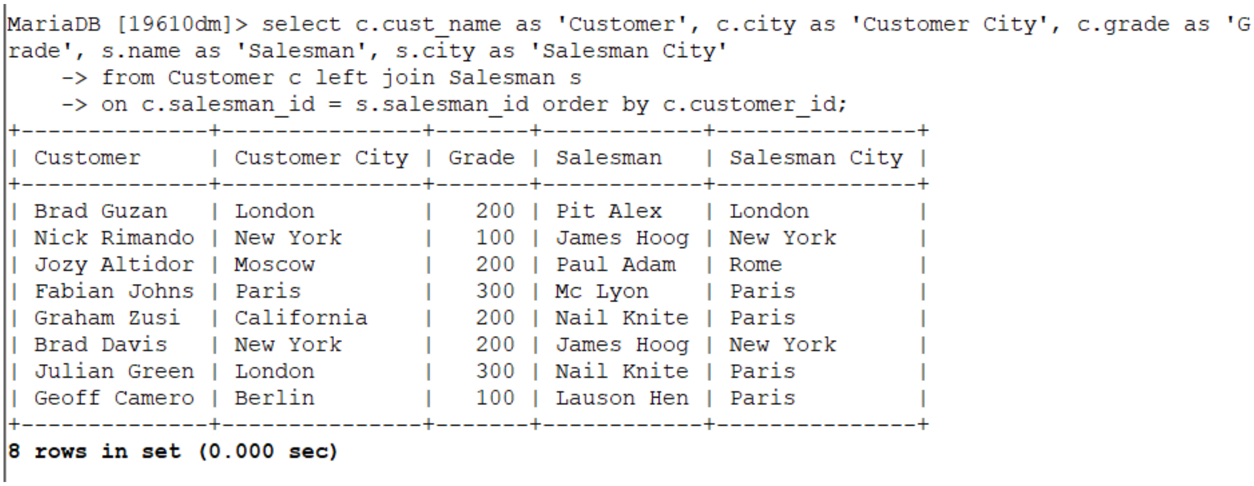
Write a SQL statement to make a list in ascending order for the customer who holds a grade less than 300 and works either through a salesman or on their own.



```
MariaDB [19610dm]> select c.cust_name as 'Customer', c.grade as 'Grade', s.name as 'Salesman'
, s.city
    -> from Customer c left outer join Salesman s
    -> on c.salesman_id = s.salesman_id
    -> where c.grade < 300 order by c.customer_id;
+--------------+-------+------------+----------+
| Customer     | Grade | Salesman   | city     |
+--------------+-------+------------+----------+
| Brad Guzan   |   200 | Pit Alex   | London   |
| Nick Rimando |   100 | James Hoog | New York |
| Jozy Altidor |   200 | Paul Adam  | Rome     |
| Graham Zusi  |   200 | Nail Knite | Paris    |
| Brad Davis   |   200 | James Hoog | New York |
| Geoff Camero |   100 | Lauson Hen | Paris    |
+--------------+-------+------------+----------+
6 rows in set (0.000 sec)
```

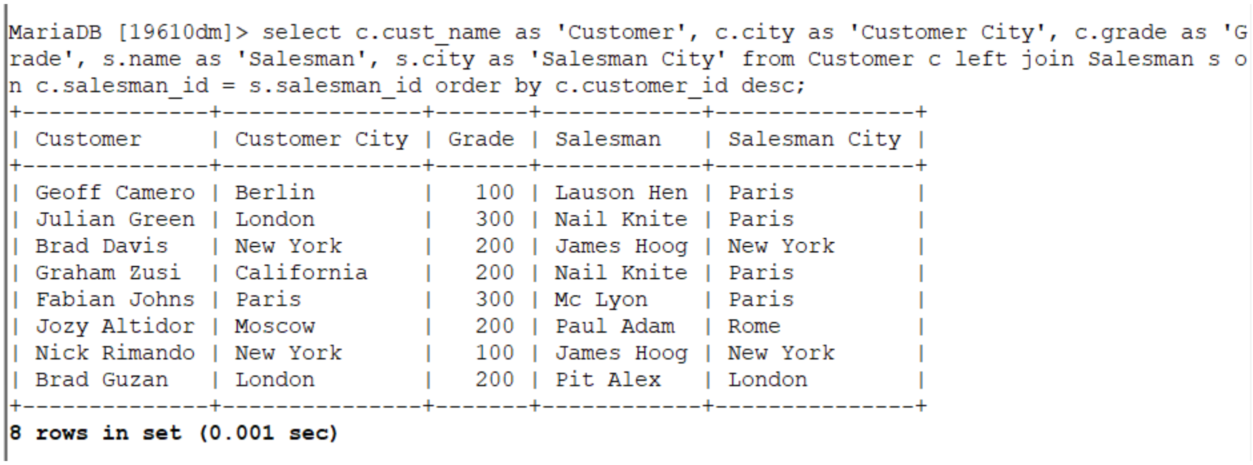
7.

a. Ascending order



```
MariaDB [19610dm]> select c.cust_name as 'Customer', c.city as 'Customer City', c.grade as 'G
rade', s.name as 'Salesman', s.city as 'Salesman City'
    -> from Customer c left join Salesman s
    -> on c.salesman_id = s.salesman_id order by c.customer_id;
+--------------+---------------+-------+------------+---------------+
| Customer     | Customer City | Grade | Salesman   | Salesman City |
+--------------+---------------+-------+------------+---------------+
| Brad Guzan   | London        |   200 | Pit Alex   | London        |
| Nick Rimando | New York      |   100 | James Hoog | New York      |
| Jozy Altidor | Moscow        |   200 | Paul Adam  | Rome          |
| Fabian Johns | Paris         |   300 | Mc Lyon    | Paris         |
| Graham Zusi  | California    |   200 | Nail Knite | Paris         |
| Brad Davis   | New York      |   200 | James Hoog | New York      |
| Julian Green | London        |   300 | Nail Knite | Paris         |
| Geoff Camero | Berlin        |   100 | Lauson Hen | Paris         |
+--------------+---------------+-------+------------+---------------+
8 rows in set (0.000 sec)
```

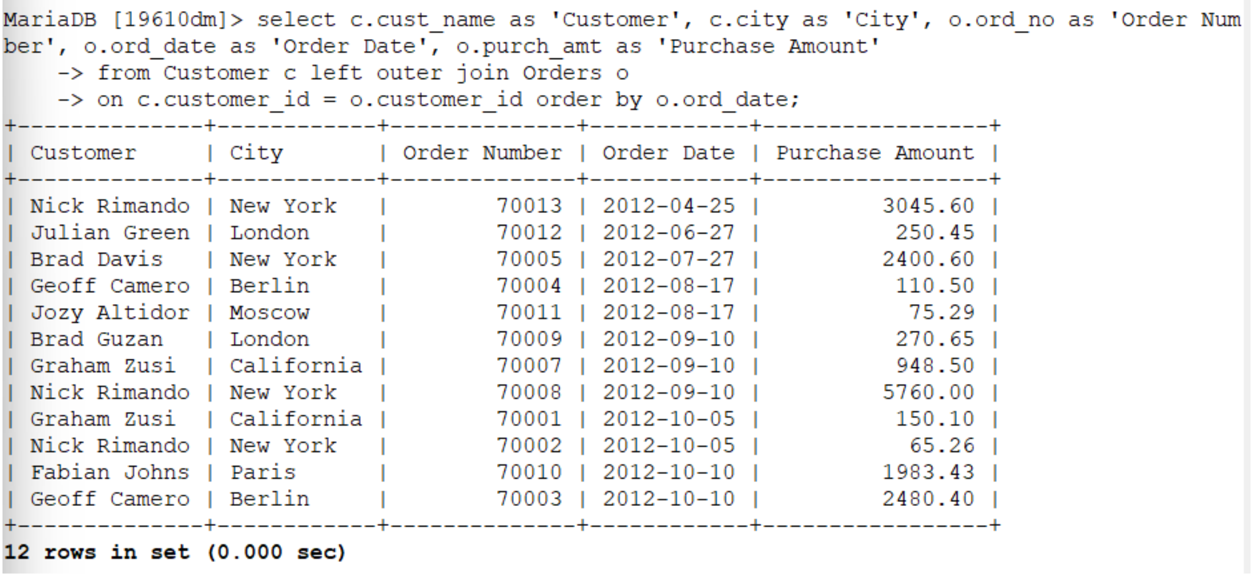
b. Descending order



```
MariaDB [19610dm]> select c.cust_name as 'Customer', c.city as 'Customer City', c.grade as 'G
rade', s.name as 'Salesman', s.city as 'Salesman City' from Customer c left join Salesman s o
n c.salesman_id = s.salesman_id order by c.customer_id desc;
+--------------+---------------+-------+------------+---------------+
| Customer     | Customer City | Grade | Salesman   | Salesman City |
+--------------+---------------+-------+------------+---------------+
| Geoff Camero | Berlin        |   100 | Lauson Hen | Paris         |
| Julian Green | London        |   300 | Nail Knite | Paris         |
| Brad Davis   | New York      |   200 | James Hoog | New York      |
| Graham Zusi  | California    |   200 | Nail Knite | Paris         |
| Fabian Johns | Paris         |   300 | Mc Lyon    | Paris         |
| Jozy Altidor | Moscow        |   200 | Paul Adam  | Rome          |
| Nick Rimando | New York      |   100 | James Hoog | New York      |
| Brad Guzan   | London        |   200 | Pit Alex   | London        |
+--------------+---------------+-------+------------+---------------+
8 rows in set (0.001 sec)
```

8.

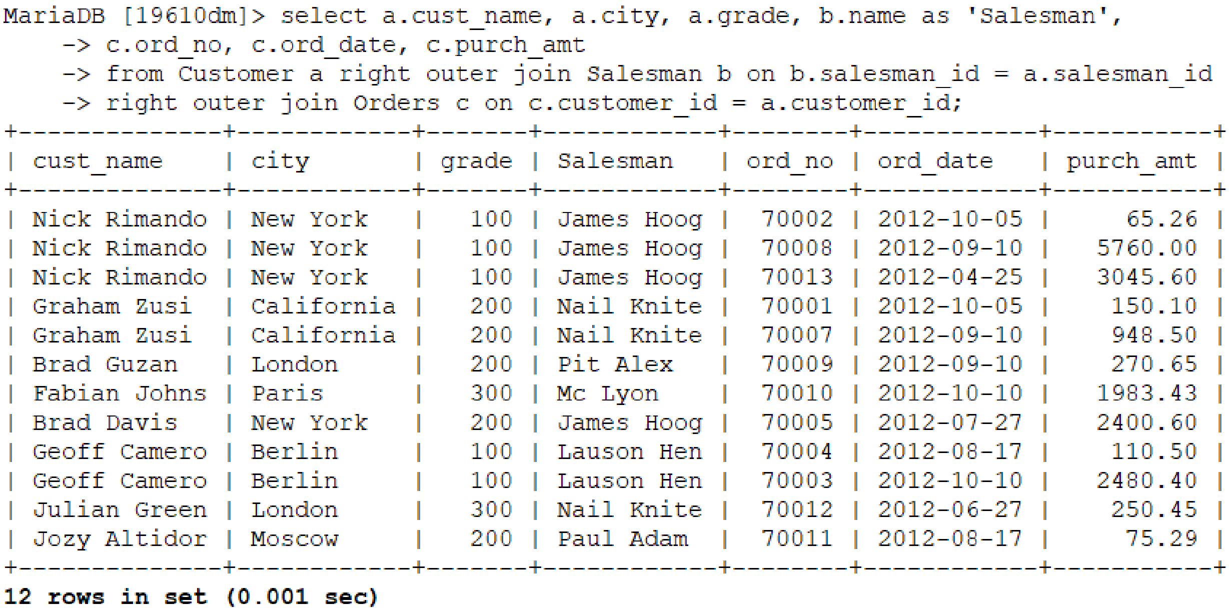
Write a SQL statement to make a report with customer name, city, order number, order date, and order amount in ascending order according to the order date to find that either any of the existing customers have placed no order or placed one or more orders



```
MariaDB [19610dm]> select c.cust_name as 'Customer', c.city as 'City', o.ord_no as 'Order Num
ber', o.ord_date as 'Order Date', o.purch_amt as 'Purchase Amount'
    -> from Customer c left outer join Orders o
    -> on c.customer_id = o.customer_id order by o.ord_date;
+--------------+------------+--------------+------------+-----------------+
| Customer     | City       | Order Number | Order Date | Purchase Amount |
+--------------+------------+--------------+------------+-----------------+
| Nick Rimando | New York   |        70013 | 2012-04-25 |         3045.60 |
| Julian Green | London     |        70012 | 2012-06-27 |          250.45 |
| Brad Davis   | New York   |        70005 | 2012-07-27 |         2400.60 |
| Geoff Camero | Berlin     |        70004 | 2012-08-17 |          110.50 |
| Jozy Altidor | Moscow     |        70011 | 2012-08-17 |           75.29 |
| Brad Guzan   | London     |        70009 | 2012-09-10 |          270.65 |
| Graham Zusi  | California |        70007 | 2012-09-10 |          948.50 |
| Nick Rimando | New York   |        70008 | 2012-09-10 |         5760.00 |
| Graham Zusi  | California |        70001 | 2012-10-05 |          150.10 |
| Nick Rimando | New York   |        70002 | 2012-10-05 |           65.26 |
| Fabian Johns | Paris      |        70010 | 2012-10-10 |         1983.43 |
| Geoff Camero | Berlin     |        70003 | 2012-10-10 |         2480.40 |
+--------------+------------+--------------+------------+-----------------+
12 rows in set (0.000 sec)
```

9.

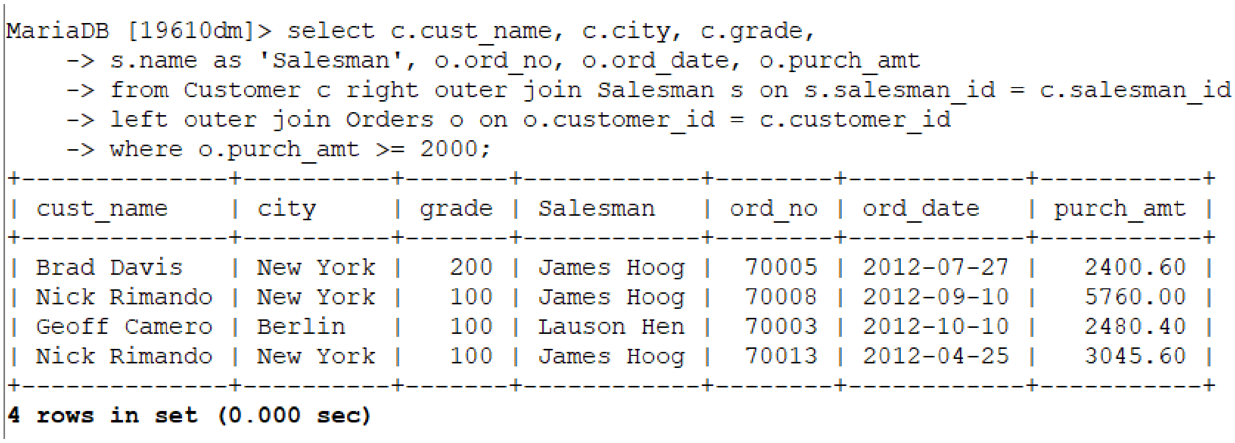
Write a SQL statement to make a list for the salesmen who works either for one or more customer or not yet join under any of the customers who placed either one or more orders or no order to their supplier.



```
MariaDB [19610dm]> select a.cust_name, a.city, a.grade, b.name as 'Salesman',
    -> c.ord_no, c.ord_date, c.purch_amt
    -> from Customer a right outer join Salesman b on b.salesman_id = a.salesman_id
    -> right outer join Orders c on c.customer_id = a.customer_id;
+--------------+------------+-------+------------+--------+------------+-----------+
| cust_name    | city       | grade | Salesman   | ord_no | ord_date   | purch_amt |
+--------------+------------+-------+------------+--------+------------+-----------+
| Nick Rimando | New York   |   100 | James Hoog |  70002 | 2012-10-05 |     65.26 |
| Nick Rimando | New York   |   100 | James Hoog |  70008 | 2012-09-10 |   5760.00 |
| Nick Rimando | New York   |   100 | James Hoog |  70013 | 2012-04-25 |   3045.60 |
| Graham Zusi  | California |   200 | Nail Knite |  70001 | 2012-10-05 |    150.10 |
| Graham Zusi  | California |   200 | Nail Knite |  70007 | 2012-09-10 |    948.50 |
| Brad Guzan   | London     |   200 | Pit Alex   |  70009 | 2012-09-10 |    270.65 |
| Fabian Johns | Paris      |   300 | Mc Lyon    |  70010 | 2012-10-10 |   1983.43 |
| Brad Davis   | New York   |   200 | James Hoog |  70005 | 2012-07-27 |   2400.60 |
| Geoff Camero | Berlin     |   100 | Lauson Hen |  70004 | 2012-08-17 |    110.50 |
| Geoff Camero | Berlin     |   100 | Lauson Hen |  70003 | 2012-10-10 |   2480.40 |
| Julian Green | London     |   300 | Nail Knite |  70012 | 2012-06-27 |    250.45 |
| Jozy Altidor | Moscow     |   200 | Paul Adam  |  70011 | 2012-08-17 |     75.29 |
+--------------+------------+-------+------------+--------+------------+-----------+
12 rows in set (0.001 sec)
```

10.

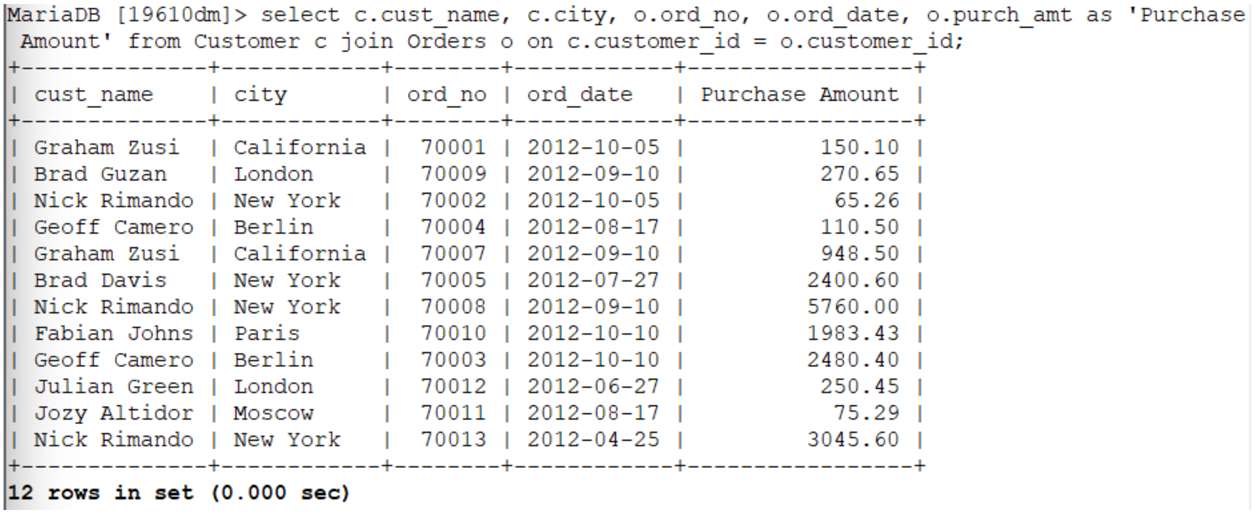
Write a SQL statement to make a list of the salesmen who either work for one or more customers or have yet to join any of the customers. The customer may have placed, either one or more orders on or above order amount 2000 and must have a grade, or he may not have placed any order to the associated supplier.



```
MariaDB [19610dm]> select c.cust_name, c.city, c.grade,
    -> s.name as 'Salesman', o.ord_no, o.ord_date, o.purch_amt
    -> from Customer c right outer join Salesman s on s.salesman_id = c.salesman_id
    -> left outer join Orders o on o.customer_id = c.customer_id
    -> where o.purch_amt >= 2000;
+--------------+----------+-------+------------+--------+------------+-----------+
| cust_name    | city     | grade | Salesman   | ord_no | ord_date   | purch_amt |
+--------------+----------+-------+------------+--------+------------+-----------+
| Brad Davis   | New York |   200 | James Hoog |  70005 | 2012-07-27 |   2400.60 |
| Nick Rimando | New York |   100 | James Hoog |  70008 | 2012-09-10 |   5760.00 |
| Geoff Camero | Berlin   |   100 | Lauson Hen |  70003 | 2012-10-10 |   2480.40 |
| Nick Rimando | New York |   100 | James Hoog |  70013 | 2012-04-25 |   3045.60 |
+--------------+----------+-------+------------+--------+------------+-----------+
4 rows in set (0.000 sec)
```

11.

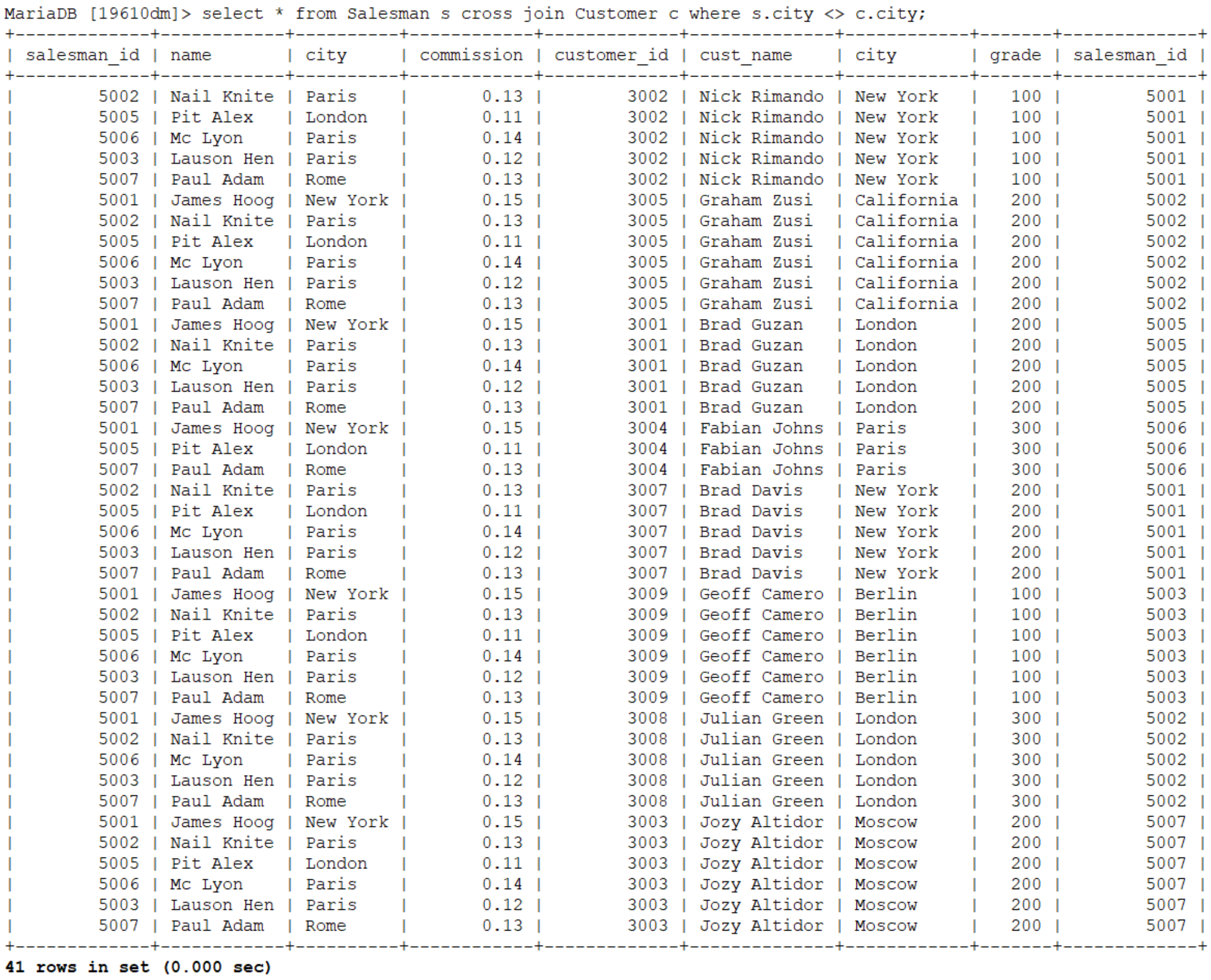
Write a SQL statement to make a report with customer name, city, order no. order date, purchase amount for only those customers on the list who must have a grade and placed one or more orders or which order(s) have been placed by the customer who is neither in the list not have a grade



```
MariaDB [19610dm]> select c.cust_name, c.city, o.ord_no, o.ord_date, o.purch_amt as 'Purchase
 Amount' from Customer c join Orders o on c.customer_id = o.customer_id;
+--------------+------------+--------+------------+-----------------+
| cust_name    | city       | ord_no | ord_date   | Purchase Amount |
+--------------+------------+--------+------------+-----------------+
| Graham Zusi  | California |  70001 | 2012-10-05 |          150.10 |
| Brad Guzan   | London     |  70009 | 2012-09-10 |          270.65 |
| Nick Rimando | New York   |  70002 | 2012-10-05 |           65.26 |
| Geoff Camero | Berlin     |  70004 | 2012-08-17 |          110.50 |
| Graham Zusi  | California |  70007 | 2012-09-10 |          948.50 |
| Brad Davis   | New York   |  70005 | 2012-07-27 |         2400.60 |
| Nick Rimando | New York   |  70008 | 2012-09-10 |         5760.00 |
| Fabian Johns | Paris      |  70010 | 2012-10-10 |         1983.43 |
| Geoff Camero | Berlin     |  70003 | 2012-10-10 |         2480.40 |
| Julian Green | London     |  70012 | 2012-06-27 |          250.45 |
| Jozy Altidor | Moscow     |  70011 | 2012-08-17 |           75.29 |
| Nick Rimando | New York   |  70013 | 2012-04-25 |         3045.60 |
+--------------+------------+--------+------------+-----------------+
12 rows in set (0.000 sec)
```

12.

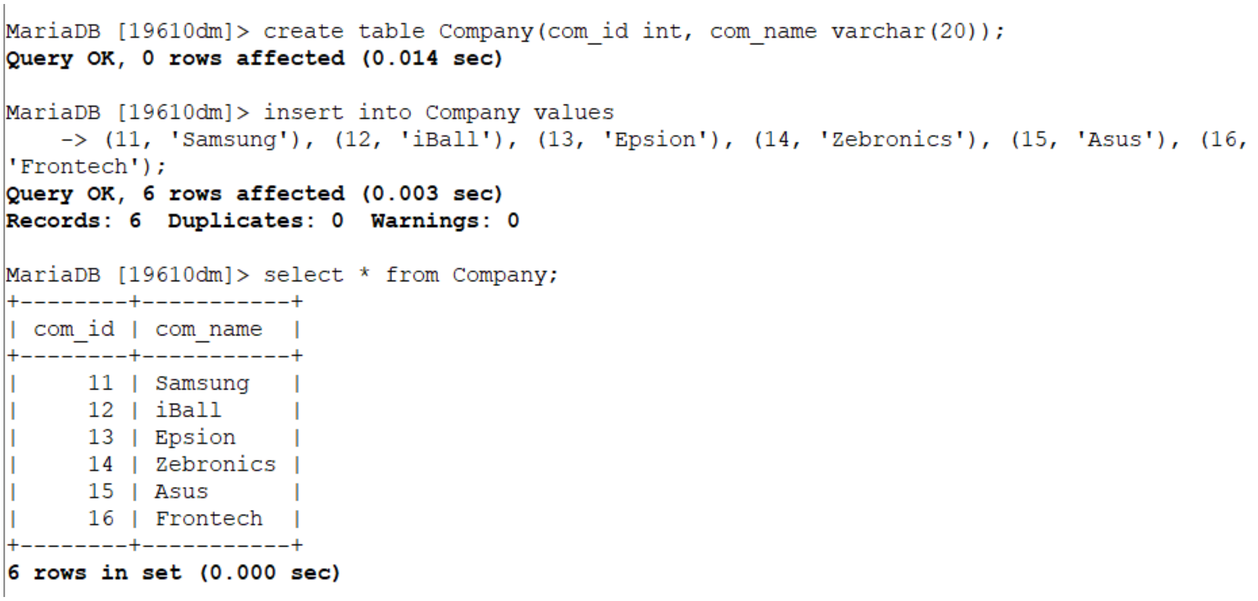
Cartesian products:



```
MariaDB [19610dm]> select * from Salesman s cross join Customer c where s.city <> c.city;
+-------------+------------+----------+------------+-------------+--------------+------------+-------+-------------+
| salesman_id | name       | city     | commission | customer_id | cust_name    | city       | grade | salesman_id |
+-------------+------------+----------+------------+-------------+--------------+------------+-------+-------------+
|        5002 | Nail Knite | Paris    |       0.13 |        3002 | Nick Rimando | New York   |   100 |        5001 |
|        5005 | Pit Alex   | London   |       0.11 |        3002 | Nick Rimando | New York   |   100 |        5001 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3002 | Nick Rimando | New York   |   100 |        5001 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3002 | Nick Rimando | New York   |   100 |        5001 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3002 | Nick Rimando | New York   |   100 |        5001 |
|        5001 | James Hoog | New York |       0.15 |        3005 | Graham Zusi  | California |   200 |        5002 |
|        5002 | Nail Knite | Paris    |       0.13 |        3005 | Graham Zusi  | California |   200 |        5002 |
|        5005 | Pit Alex   | London   |       0.11 |        3005 | Graham Zusi  | California |   200 |        5002 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3005 | Graham Zusi  | California |   200 |        5002 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3005 | Graham Zusi  | California |   200 |        5002 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3005 | Graham Zusi  | California |   200 |        5002 |
|        5001 | James Hoog | New York |       0.15 |        3001 | Brad Guzan   | London     |   200 |        5005 |
|        5002 | Nail Knite | Paris    |       0.13 |        3001 | Brad Guzan   | London     |   200 |        5005 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3001 | Brad Guzan   | London     |   200 |        5005 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3001 | Brad Guzan   | London     |   200 |        5005 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3001 | Brad Guzan   | London     |   200 |        5005 |
|        5001 | James Hoog | New York |       0.15 |        3004 | Fabian Johns | Paris      |   300 |        5006 |
|        5005 | Pit Alex   | London   |       0.11 |        3004 | Fabian Johns | Paris      |   300 |        5006 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3004 | Fabian Johns | Paris      |   300 |        5006 |
|        5002 | Nail Knite | Paris    |       0.13 |        3007 | Brad Davis   | New York   |   200 |        5001 |
|        5005 | Pit Alex   | London   |       0.11 |        3007 | Brad Davis   | New York   |   200 |        5001 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3007 | Brad Davis   | New York   |   200 |        5001 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3007 | Brad Davis   | New York   |   200 |        5001 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3007 | Brad Davis   | New York   |   200 |        5001 |
|        5001 | James Hoog | New York |       0.15 |        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        5002 | Nail Knite | Paris    |       0.13 |        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        5005 | Pit Alex   | London   |       0.11 |        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3009 | Geoff Camero | Berlin     |   100 |        5003 |
|        5001 | James Hoog | New York |       0.15 |        3008 | Julian Green | London     |   300 |        5002 |
|        5002 | Nail Knite | Paris    |       0.13 |        3008 | Julian Green | London     |   300 |        5002 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3008 | Julian Green | London     |   300 |        5002 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3008 | Julian Green | London     |   300 |        5002 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3008 | Julian Green | London     |   300 |        5002 |
|        5001 | James Hoog | New York |       0.15 |        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
|        5002 | Nail Knite | Paris    |       0.13 |        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
|        5005 | Pit Alex   | London   |       0.11 |        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
|        5006 | Mc Lyon    | Paris    |       0.14 |        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
|        5003 | Lauson Hen | Paris    |       0.12 |        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
|        5007 | Paul Adam  | Rome     |       0.13 |        3003 | Jozy Altidor | Moscow     |   200 |        5007 |
+-------------+------------+----------+------------+-------------+--------------+------------+-------+-------------+
41 rows in set (0.000 sec)
```

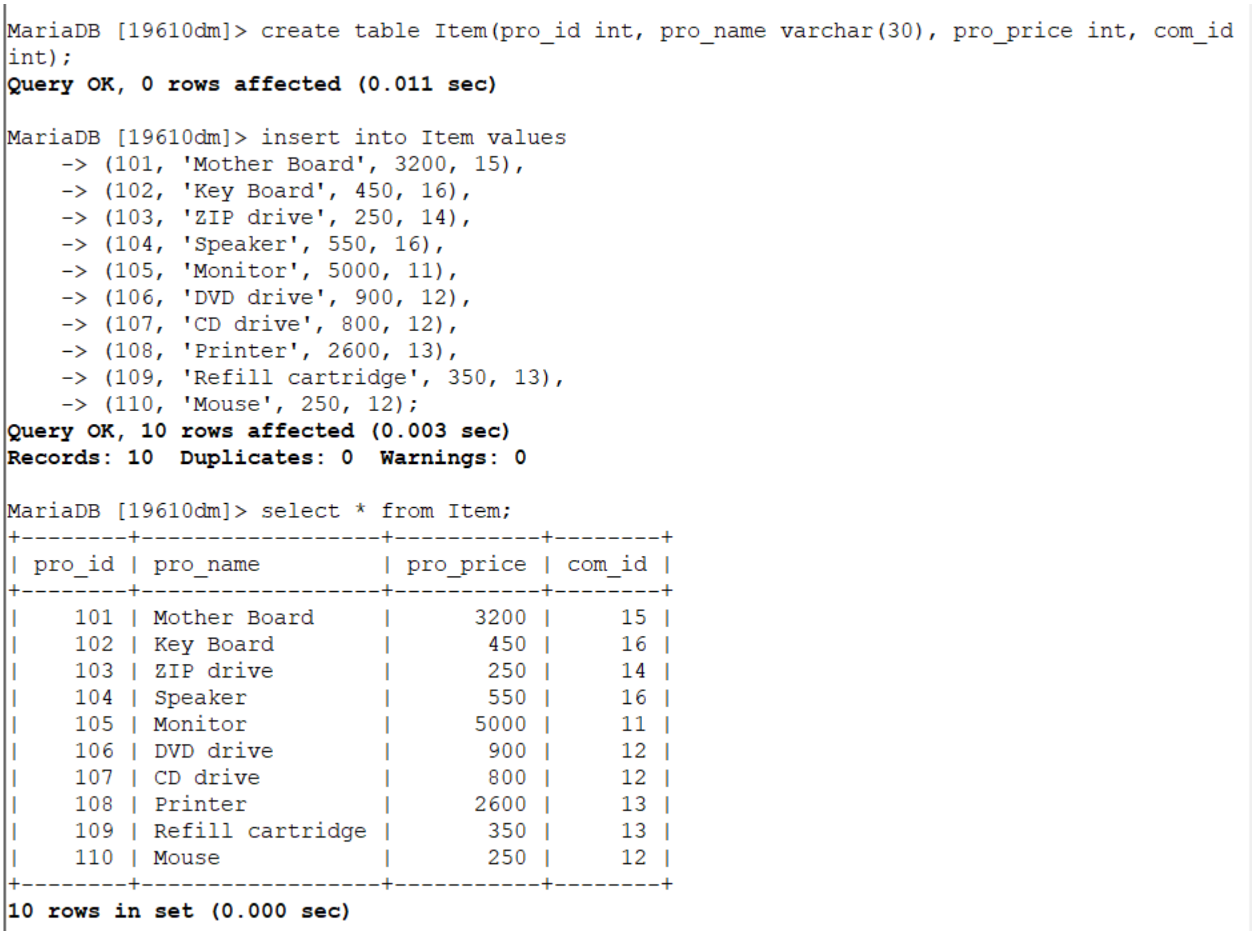
13.

Create table Company:

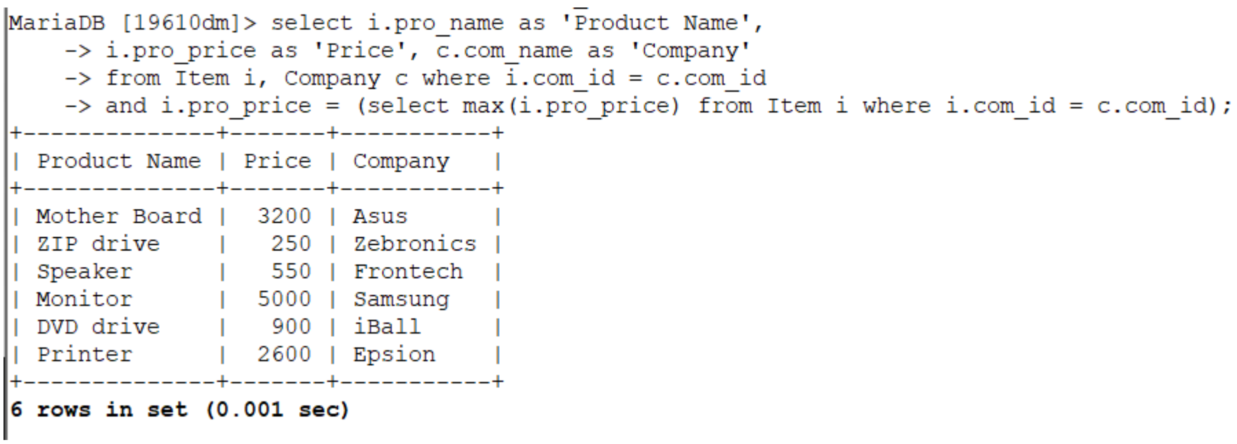


```
MariaDB [19610dm]> create table Company(com_id int, com_name varchar(20));
Query OK, 0 rows affected (0.014 sec)

MariaDB [19610dm]> insert into Company values
    -> (11, 'Samsung'), (12, 'iBall'), (13, 'Epsion'), (14, 'Zebronics'), (15, 'Asus'), (16,
'Frontech');
Query OK, 6 rows affected (0.003 sec)
Records: 6  Duplicates: 0  Warnings: 0

MariaDB [19610dm]> select * from Company;
+--------+-----------+
| com_id | com_name  |
+--------+-----------+
|     11 | Samsung   |
|     12 | iBall     |
|     13 | Epsion    |
|     14 | Zebronics |
|     15 | Asus      |
|     16 | Frontech  |
+--------+-----------+
6 rows in set (0.000 sec)
```

Create table Item:



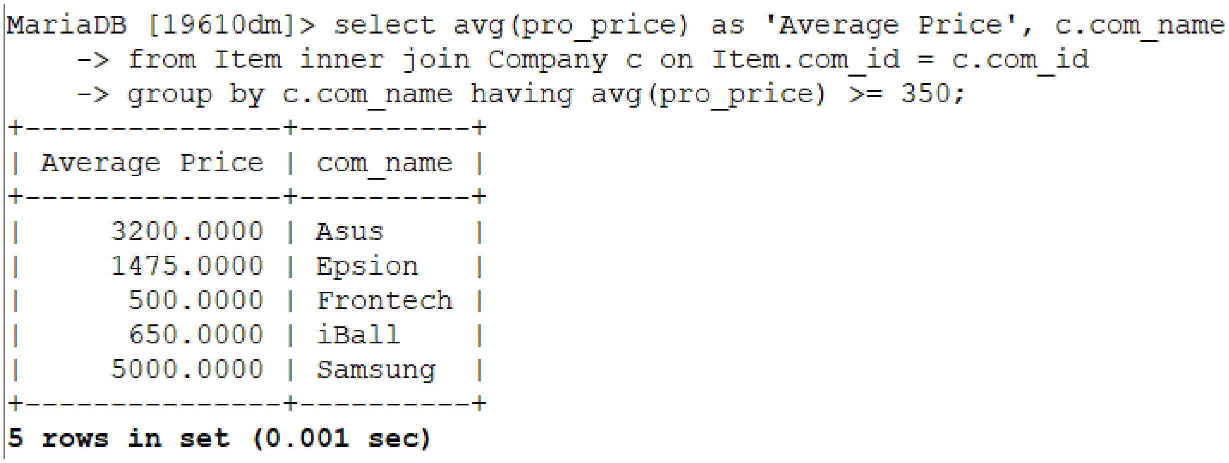
```
MariaDB [19610dm]> create table Item(pro_id int, pro_name varchar(30), pro_price int, com_id
int);
Query OK, 0 rows affected (0.011 sec)

MariaDB [19610dm]> insert into Item values
    -> (101, 'Mother Board', 3200, 15),
    -> (102, 'Key Board', 450, 16),
    -> (103, 'ZIP drive', 250, 14),
    -> (104, 'Speaker', 550, 16),
    -> (105, 'Monitor', 5000, 11),
    -> (106, 'DVD drive', 900, 12),
    -> (107, 'CD drive', 800, 12),
    -> (108, 'Printer', 2600, 13),
    -> (109, 'Refill cartridge', 350, 13),
    -> (110, 'Mouse', 250, 12);
Query OK, 10 rows affected (0.003 sec)
Records: 10  Duplicates: 0  Warnings: 0

MariaDB [19610dm]> select * from Item;
+--------+------------------+-----------+--------+
| pro_id | pro_name         | pro_price | com_id |
+--------+------------------+-----------+--------+
|    101 | Mother Board     |      3200 |     15 |
|    102 | Key Board        |       450 |     16 |
|    103 | ZIP drive        |       250 |     14 |
|    104 | Speaker          |       550 |     16 |
|    105 | Monitor          |      5000 |     11 |
|    106 | DVD drive        |       900 |     12 |
|    107 | CD drive         |       800 |     12 |
|    108 | Printer          |      2600 |     13 |
|    109 | Refill cartridge |       350 |     13 |
|    110 | Mouse            |       250 |     12 |
+--------+------------------+-----------+--------+
10 rows in set (0.000 sec)
```



```
MariaDB [19610dm]> select i.pro_name as 'Product Name',
    -> i.pro_price as 'Price', c.com_name as 'Company'
    -> from Item i, Company c where i.com_id = c.com_id
    -> and i.pro_price = (select max(i.pro_price) from Item i where i.com_id = c.com_id);
+--------------+-------+-----------+
| Product Name | Price | Company   |
+--------------+-------+-----------+
| Mother Board |  3200 | Asus      |
| ZIP drive    |   250 | Zebronics |
| Speaker      |   550 | Frontech  |
| Monitor      |  5000 | Samsung   |
| DVD drive    |   900 | iBall     |
| Printer      |  2600 | Epsion    |
+--------------+-------+-----------+
6 rows in set (0.001 sec)
```

14.

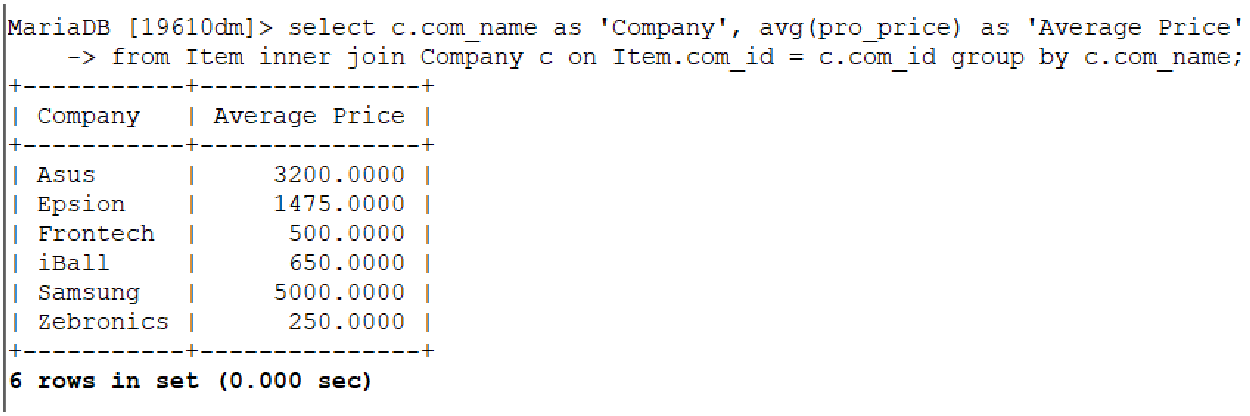
Write a SQL query to display the names of the company whose products have an average price larger than or equal to 350



```
MariaDB [19610dm]> select avg(pro_price) as 'Average Price', c.com_name
    -> from Item inner join Company c on Item.com_id = c.com_id
    -> group by c.com_name having avg(pro_price) >= 350;
+---------------+----------+
| Average Price | com_name |
+---------------+----------+
|     3200.0000 | Asus     |
|     1475.0000 | Epsion   |
|      500.0000 | Frontech |
|      650.0000 | iBall    |
|     5000.0000 | Samsung  |
+---------------+----------+
5 rows in set (0.001 sec)
```

15.

Write a SQL query to display the average price of items of each company, showing the name of the company



```
MariaDB [19610dm]> select c.com_name as 'Company', avg(pro_price) as 'Average Price'
    -> from Item inner join Company c on Item.com_id = c.com_id group by c.com_name;
+-----------+---------------+
| Company   | Average Price |
+-----------+---------------+
| Asus      |     3200.0000 |
| Epsion    |     1475.0000 |
| Frontech  |      500.0000 |
| iBall     |      650.0000 |
| Samsung   |     5000.0000 |
| Zebronics |      250.0000 |
+-----------+---------------+
6 rows in set (0.000 sec)
```