Assignment-20

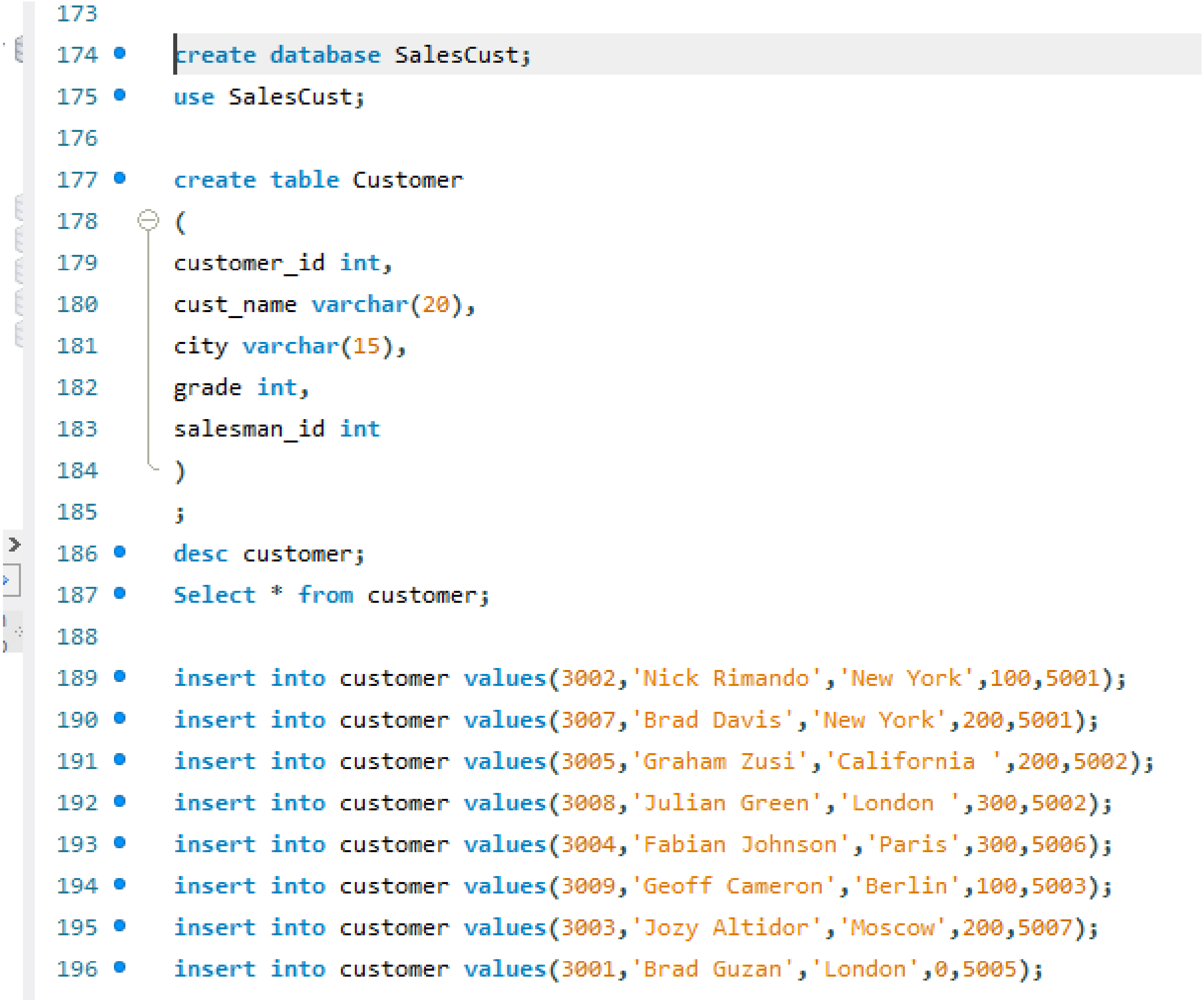
Where Clause

Kaviya C

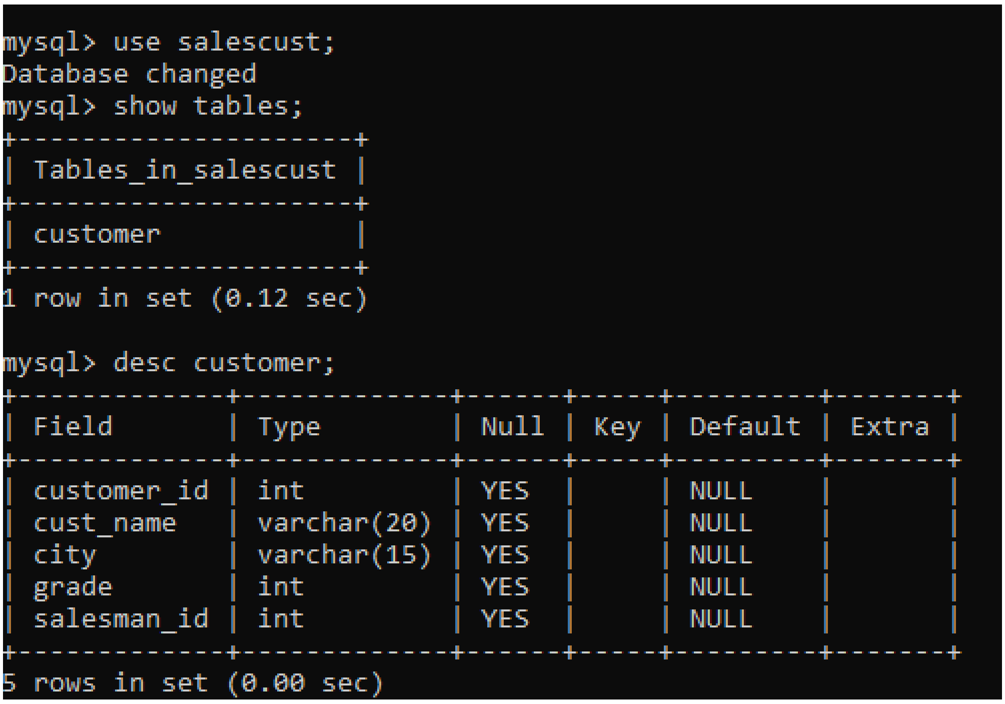
1. From the following table, write a SQL query to find the details of the customers who have a gradevalue above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

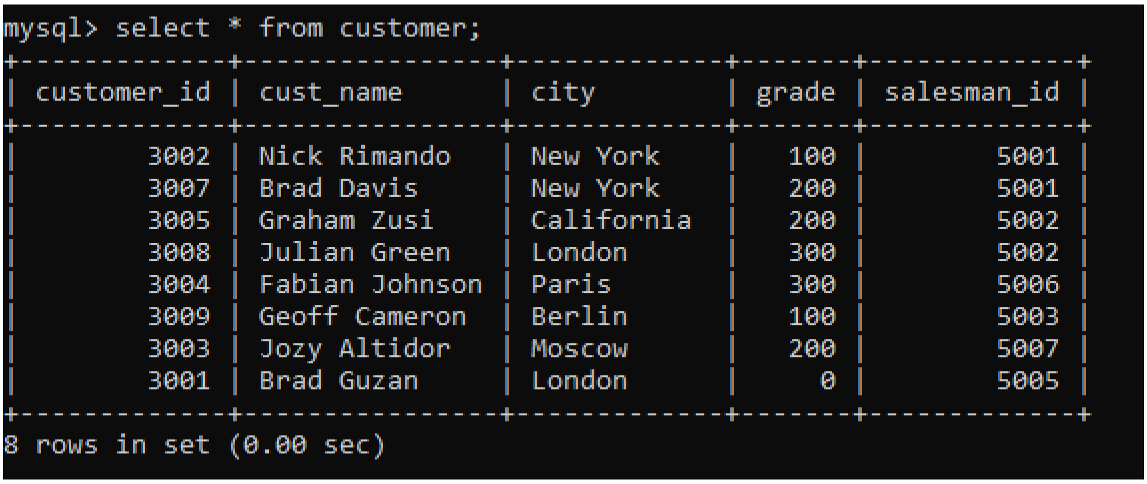
```
customer_id |   cust_name    |    city    | grade | 
salesman_id
-------------+----------------+------------+-------+----------
---
        3002 | Nick Rimando   | New York   |   100 | 
5001
        3007 | Brad Davis     | New York   |   200 | 
5001
        3005 | Graham Zusi    | California |   200 | 
5002
        3008 | Julian Green   | London     |   300 | 
5002
        3004 | Fabian Johnson | Paris      |   300 | 
5006
        3009 | Geoff Cameron  | Berlin     |   100 | 
5003
        3003 | Jozy Altidor   | Moscow     |   200 | 
5007
        3001 | Brad Guzan     | London     |       | 
5005
```



```sql

create database SalesCust;
use SalesCust;

create table Customer
(
customer_id int,
cust_name varchar(20),
city varchar(15),
grade int,
salesman_id int
)
;
desc customer;
Select * from customer;

insert into customer values(3002,'Nick Rimando','New York',100,5001);
insert into customer values(3007,'Brad Davis','New York',200,5001);
insert into customer values(3005,'Graham Zusi','California ',200,5002);
insert into customer values(3008,'Julian Green','London ',300,5002);
insert into customer values(3004,'Fabian Johnson','Paris',300,5006);
insert into customer values(3009,'Geoff Cameron','Berlin',100,5003);
insert into customer values(3003,'Jozy Altidor','Moscow',200,5007);
insert into customer values(3001,'Brad Guzan','London',0,5005);
```



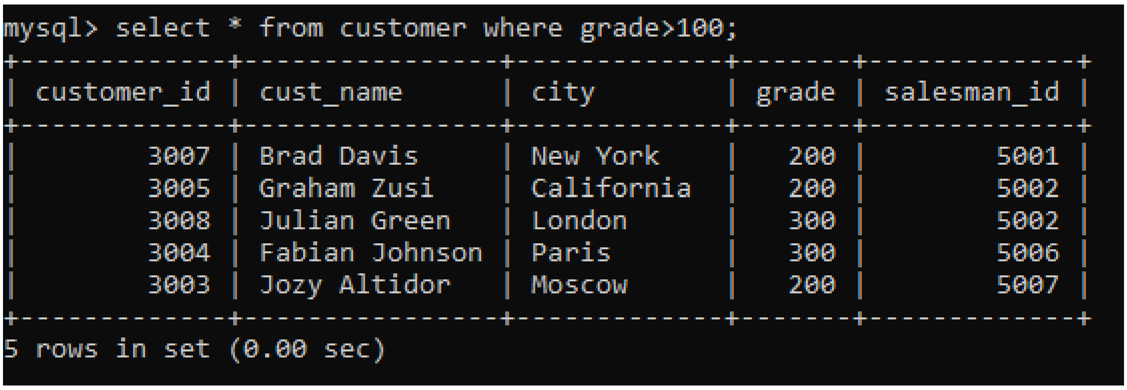
```
mysql> use salescust;
Database changed
mysql> show tables;
+---------------------+
| Tables_in_salescust |
+---------------------+
| customer            |
+---------------------+
1 row in set (0.12 sec)

mysql> desc customer;
+-------------+-------------+------+-----+---------+-------+
| Field       | Type        | Null | Key | Default | Extra |
+-------------+-------------+------+-----+---------+-------+
| customer_id | int         | YES  |     | NULL    |       |
| cust_name   | varchar(20) | YES  |     | NULL    |       |
| city        | varchar(15) | YES  |     | NULL    |       |
| grade       | int         | YES  |     | NULL    |       |
| salesman_id | int         | YES  |     | NULL    |       |
+-------------+-------------+------+-----+---------+-------+
5 rows in set (0.00 sec)
```



```
mysql> select * from customer;
+-------------+----------------+------------+-------+-------------+
| customer_id | cust_name      | city       | grade | salesman_id |
+-------------+----------------+------------+-------+-------------+
|        3002 | Nick Rimando   | New York   |   100 |        5001 |
|        3007 | Brad Davis     | New York   |   200 |        5001 |
|        3005 | Graham Zusi    | California |   200 |        5002 |
|        3008 | Julian Green   | London     |   300 |        5002 |
|        3004 | Fabian Johnson | Paris      |   300 |        5006 |
|        3009 | Geoff Cameron  | Berlin     |   100 |        5003 |
|        3003 | Jozy Altidor   | Moscow     |   200 |        5007 |
|        3001 | Brad Guzan     | London     |     0 |        5005 |
+-------------+----------------+------------+-------+-------------+
8 rows in set (0.00 sec)
```

Query:

ANS:



```
mysql> select * from customer where grade>100;
+-------------+----------------+------------+-------+-------------+
| customer_id | cust_name      | city       | grade | salesman_id |
+-------------+----------------+------------+-------+-------------+
|        3007 | Brad Davis     | New York   |   200 |        5001 |
|        3005 | Graham Zusi    | California |   200 |        5002 |
|        3008 | Julian Green   | London     |   300 |        5002 |
|        3004 | Fabian Johnson | Paris      |   300 |        5006 |
|        3003 | Jozy Altidor   | Moscow     |   200 |        5007 |
+-------------+----------------+------------+-------+-------------+
5 rows in set (0.00 sec)
```

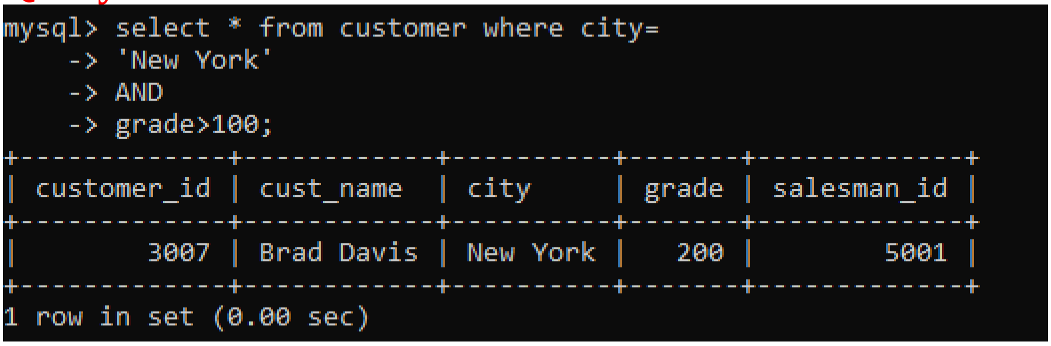
2. From the following table, write a SQL query to find all the customers in ‘New York’ city who have a grade value above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

```
customer_id |   cust_name     |    city     | grade |
salesman_id
-------------+----------------+------------+-------+----------
---
        3002 | Nick Rimando   | New York   |   100 |
5001
        3007 | Brad Davis     | New York   |   200 |
5001
        3005 | Graham Zusi    | California |   200 |
5002
        3008 | Julian Green   | London     |   300 |
5002
        3004 | Fabian Johnson | Paris      |   300 |
5006
```

```
        3009 | Geoff Cameron  | Berlin      |   100 |
5003
        3003 | Jozy Altidor   | Moscow      |   200 |
5007
        3001 | Brad Guzan     | London      |       |
5005
```

Query:



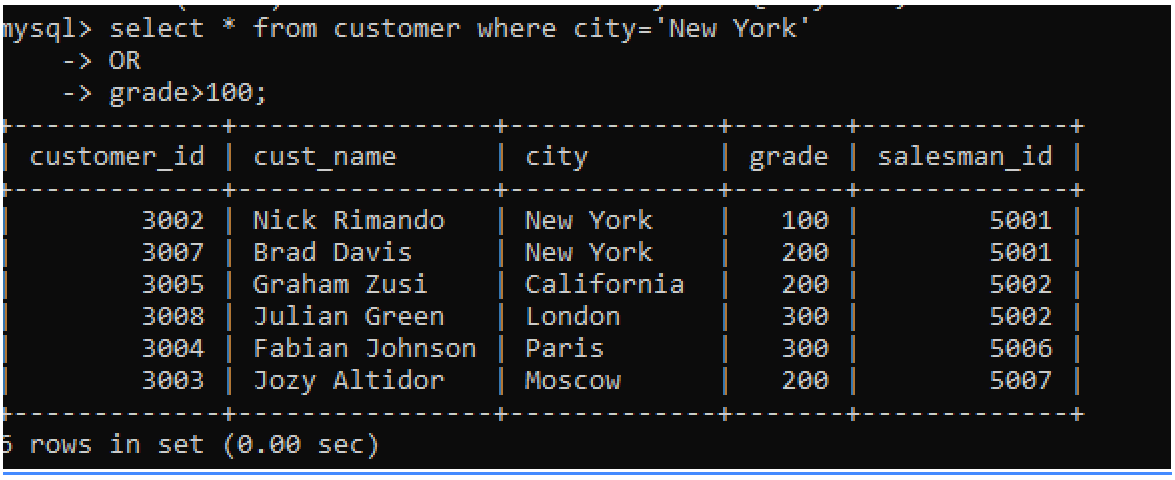
```
mysql> select * from customer where city=
    -> 'New York'
    -> AND
    -> grade>100;
+-------------+------------+----------+-------+-------------+
| customer_id | cust_name  | city     | grade | salesman_id |
+-------------+------------+----------+-------+-------------+
|        3007 | Brad Davis | New York |   200 |        5001 |
+-------------+------------+----------+-------+-------------+
1 row in set (0.00 sec)
```

3. From the following table, write a SQL query to find the customers who belong to either the city ‘New York’ or have a grade above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

```
customer_id |   cust_name    |    city     | grade |
salesman_id
-------------+----------------+-------------+-------+----------
---
        3002 | Nick Rimando   | New York    |   100 |
5001
        3007 | Brad Davis     | New York    |   200 |
5001
        3005 | Graham Zusi    | California  |   200 |
5002
        3008 | Julian Green   | London      |   300 |
5002
        3004 | Fabian Johnson | Paris       |   300 |
5006
        3009 | Geoff Cameron  | Berlin      |   100 |
5003
        3003 | Jozy Altidor   | Moscow      |   200 |
5007
        3001 | Brad Guzan     | London      |       |
5005
```

Query:



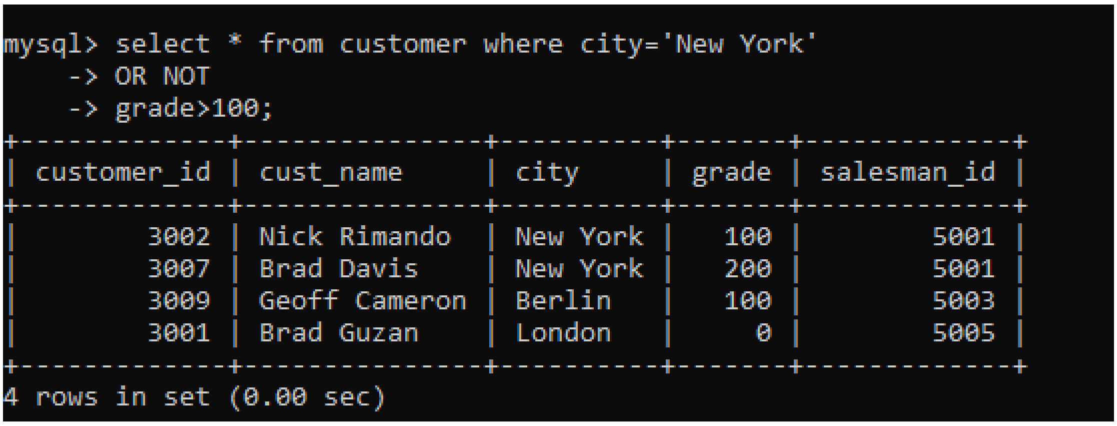
```
mysql> select * from customer where city='New York'
    -> OR
    -> grade>100;
+-------------+----------------+------------+-------+-------------+
| customer_id | cust_name      | city       | grade | salesman_id |
+-------------+----------------+------------+-------+-------------+
|        3002 | Nick Rimando   | New York   |   100 |        5001 |
|        3007 | Brad Davis     | New York   |   200 |        5001 |
|        3005 | Graham Zusi    | California |   200 |        5002 |
|        3008 | Julian Green   | London     |   300 |        5002 |
|        3004 | Fabian Johnson | Paris      |   300 |        5006 |
|        3003 | Jozy Altidor   | Moscow     |   200 |        5007 |
+-------------+----------------+------------+-------+-------------+
6 rows in set (0.00 sec)
```

4. From the following table, write a SQL query to find the customers who belong to either the city ‘New York’ or not have a grade above 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

```
customer_id |   cust_name    |    city    | grade | 
salesman_id
-------------+----------------+------------+-------+----------
---
        3002 | Nick Rimando   | New York   |   100 | 
5001
        3007 | Brad Davis     | New York   |   200 | 
5001
        3005 | Graham Zusi    | California |   200 | 
5002
        3008 | Julian Green   | London     |   300 | 
5002
        3004 | Fabian Johnson | Paris      |   300 | 
5006
        3009 | Geoff Cameron  | Berlin     |   100 | 
5003
        3003 | Jozy Altidor   | Moscow     |   200 | 
5007
        3001 | Brad Guzan     | London     |       | 
5005
```

Query:



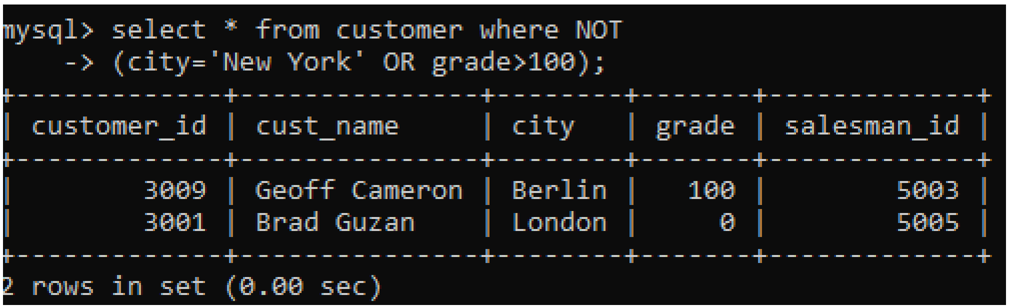
```
mysql> select * from customer where city='New York'
    -> OR NOT
    -> grade>100;
+-------------+----------------+----------+-------+-------------+
| customer_id | cust_name      | city     | grade | salesman_id |
+-------------+----------------+----------+-------+-------------+
|        3002 | Nick Rimando   | New York |   100 |        5001 |
|        3007 | Brad Davis     | New York |   200 |        5001 |
|        3009 | Geoff Cameron  | Berlin   |   100 |        5003 |
|        3001 | Brad Guzan     | London   |     0 |        5005 |
+-------------+----------------+----------+-------+-------------+
4 rows in set (0.00 sec)
```

5. From the following table, write a SQL query to find those customers who belong to neither the ‘New York’ city nor their grade value exceeds 100. Return customer_id, cust_name, city, grade, and salesman_id.

Sample table: customer

```
customer_id |   cust_name     |    city     | grade |
salesman_id
-------------+----------------+------------+-------+----------
---
        3002 | Nick Rimando   | New York   |   100 |
5001
        3007 | Brad Davis     | New York   |   200 |
5001
        3005 | Graham Zusi    | California |   200 |
5002
        3008 | Julian Green   | London     |   300 |
5002
        3004 | Fabian Johnson | Paris      |   300 |
5006
        3009 | Geoff Cameron  | Berlin     |   100 |
5003
        3003 | Jozy Altidor   | Moscow     |   200 |
5007
        3001 | Brad Guzan     | London     |       |
5005
```

Query:

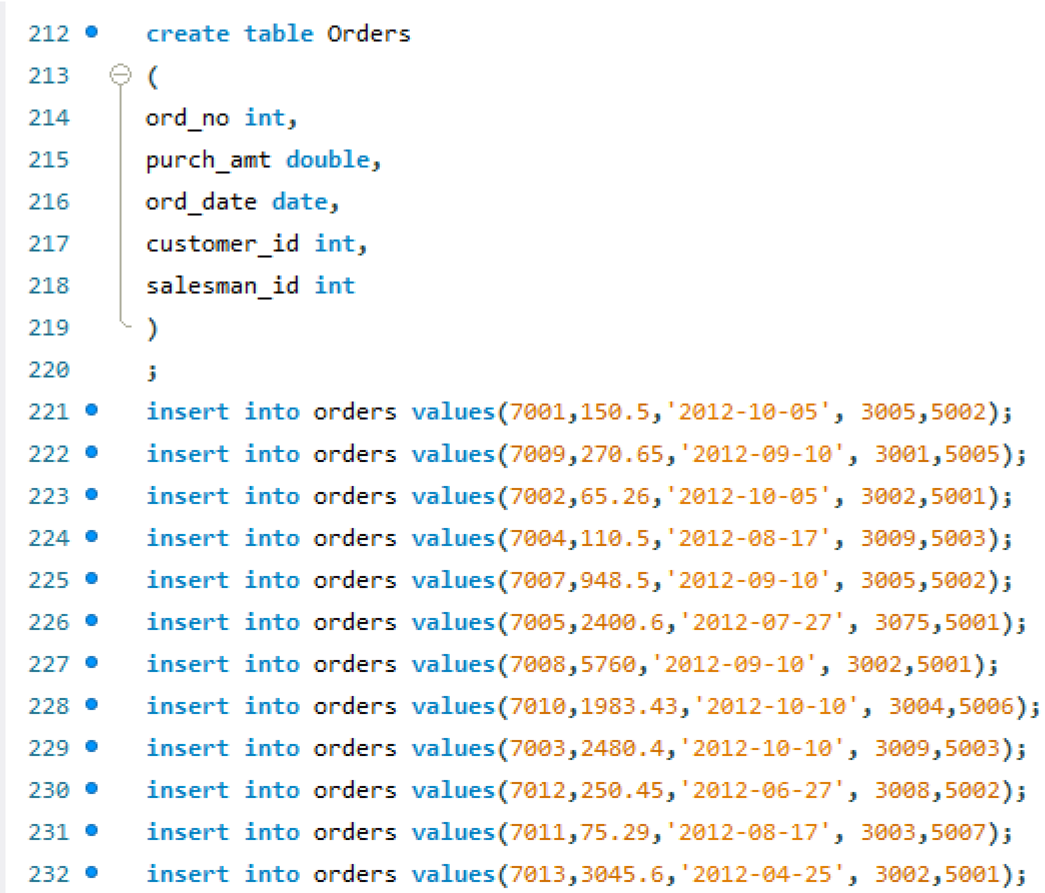


```
mysql> select * from customer where NOT
    -> (city='New York' OR grade>100);
+-------------+--------------+--------+-------+-------------+
| customer_id | cust_name    | city   | grade | salesman_id |
+-------------+--------------+--------+-------+-------------+
|        3009 | Geoff Cameron | Berlin |   100 |        5003 |
|        3001 | Brad Guzan   | London |     0 |        5005 |
+-------------+--------------+--------+-------+-------------+
2 rows in set (0.00 sec)
```

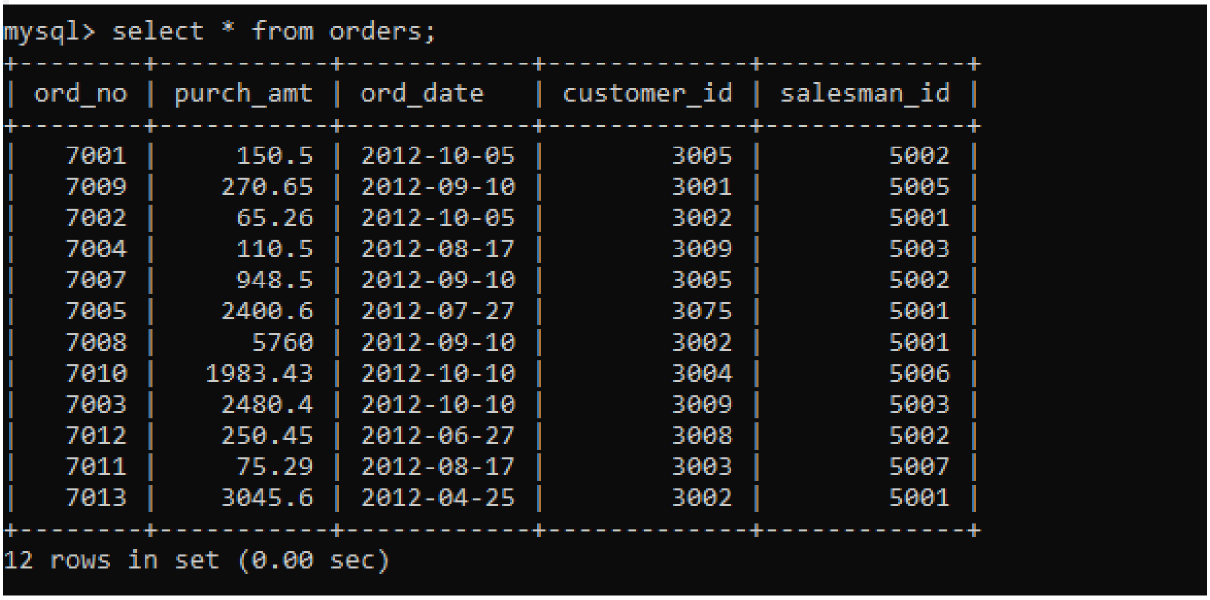
6. From the following table, write a SQL query to find details of all order excluding combination of ord_date equal to '2012-09-10' and salesman_id higher than 5005 or purch_amt greater than 1000. Return ord_no, purch_amt, ord_date, customer_id and salesman_id.

Sample table : orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	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



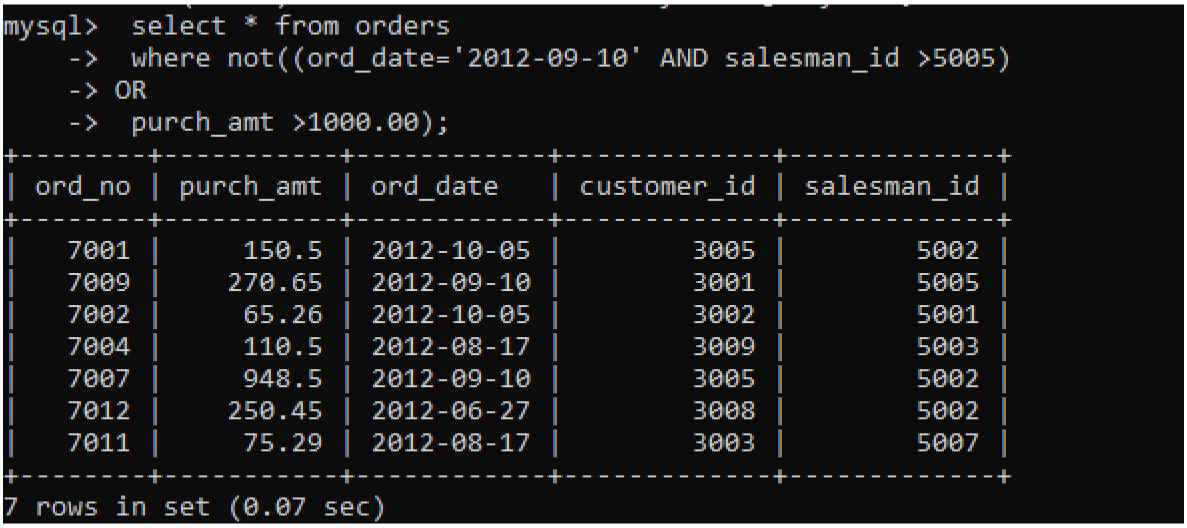
```sql
create table Orders
(
ord_no int,
purch_amt double,
ord_date date,
customer_id int,
salesman_id int
)
;
insert into orders values(7001,150.5,'2012-10-05', 3005,5002);
insert into orders values(7009,270.65,'2012-09-10', 3001,5005);
insert into orders values(7002,65.26,'2012-10-05', 3002,5001);
insert into orders values(7004,110.5,'2012-08-17', 3009,5003);
insert into orders values(7007,948.5,'2012-09-10', 3005,5002);
insert into orders values(7005,2400.6,'2012-07-27', 3075,5001);
insert into orders values(7008,5760,'2012-09-10', 3002,5001);
insert into orders values(7010,1983.43,'2012-10-10', 3004,5006);
insert into orders values(7003,2480.4,'2012-10-10', 3009,5003);
insert into orders values(7012,250.45,'2012-06-27', 3008,5002);
insert into orders values(7011,75.29,'2012-08-17', 3003,5007);
insert into orders values(7013,3045.6,'2012-04-25', 3002,5001);
```



```
mysql> select * from orders;
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
|   7001 |     150.5 | 2012-10-05 |        3005 |        5002 |
|   7009 |    270.65 | 2012-09-10 |        3001 |        5005 |
|   7002 |     65.26 | 2012-10-05 |        3002 |        5001 |
|   7004 |     110.5 | 2012-08-17 |        3009 |        5003 |
|   7007 |     948.5 | 2012-09-10 |        3005 |        5002 |
|   7005 |    2400.6 | 2012-07-27 |        3075 |        5001 |
|   7008 |      5760 | 2012-09-10 |        3002 |        5001 |
|   7010 |   1983.43 | 2012-10-10 |        3004 |        5006 |
|   7003 |    2480.4 | 2012-10-10 |        3009 |        5003 |
|   7012 |    250.45 | 2012-06-27 |        3008 |        5002 |
|   7011 |     75.29 | 2012-08-17 |        3003 |        5007 |
|   7013 |    3045.6 | 2012-04-25 |        3002 |        5001 |
+--------+-----------+------------+-------------+-------------+
12 rows in set (0.00 sec)
```

Query:

ANS:

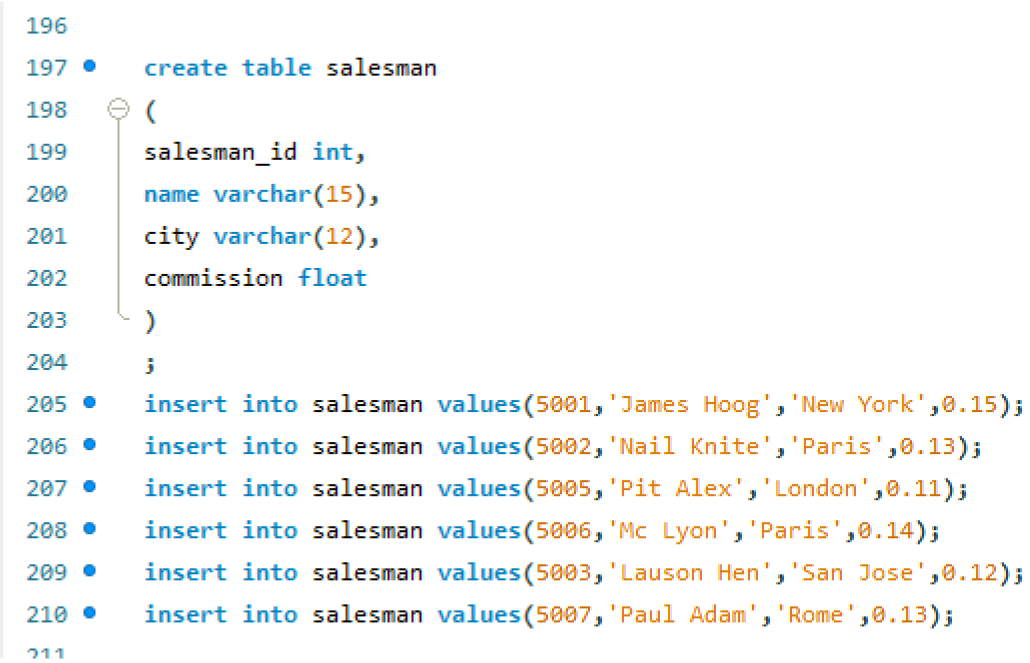


```
mysql>  select * from orders
    ->  where not((ord_date='2012-09-10' AND salesman_id >5005)
    -> OR
    ->  purch_amt >1000.00);
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
|   7001 |     150.5 | 2012-10-05 |        3005 |        5002 |
|   7009 |    270.65 | 2012-09-10 |        3001 |        5005 |
|   7002 |     65.26 | 2012-10-05 |        3002 |        5001 |
|   7004 |     110.5 | 2012-08-17 |        3009 |        5003 |
|   7007 |     948.5 | 2012-09-10 |        3005 |        5002 |
|   7012 |    250.45 | 2012-06-27 |        3008 |        5002 |
|   7011 |     75.29 | 2012-08-17 |        3003 |        5007 |
+--------+-----------+------------+-------------+-------------+
7 rows in set (0.07 sec)
```

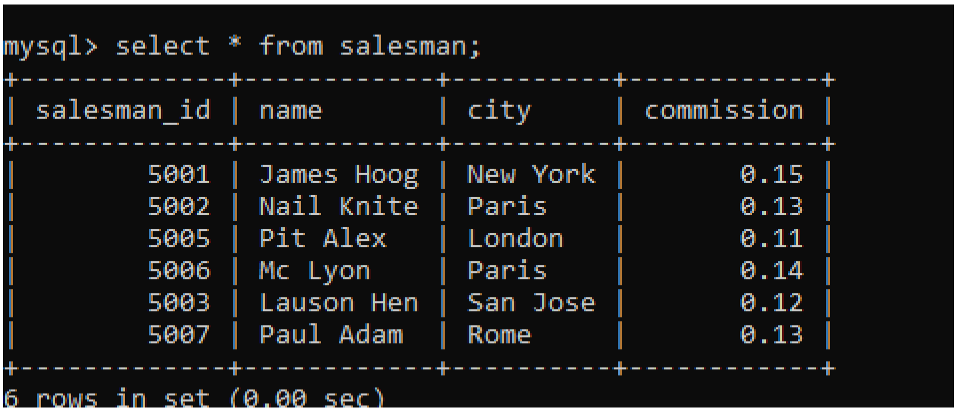
7. From the following table, write a SQL query to find the details of those salespeople whose commissions range from 0.10 to0.12. Return salesman_id, name, city, and commission.

Sample table : 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
        5007 | Paul Adam  | Rome     |       0.13
        5003 | Lauson Hen | San Jose |       0.12
```



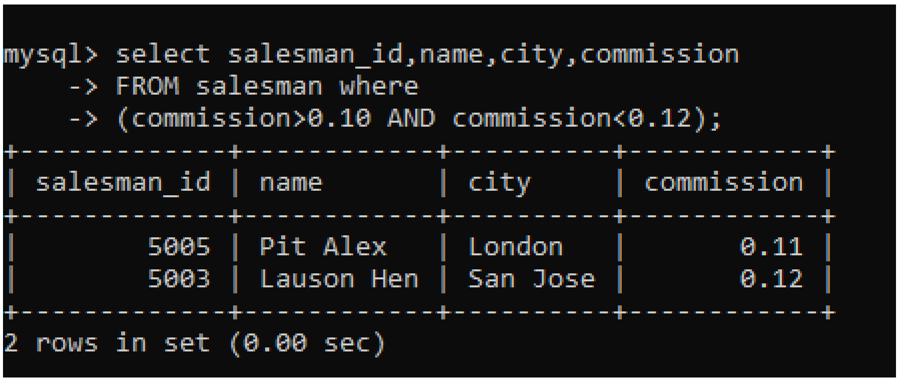
```
create table salesman
(
salesman_id int,
name varchar(15),
city varchar(12),
commission float
)
;
insert into salesman values(5001,'James Hoog','New York',0.15);
insert into salesman values(5002,'Nail Knite','Paris',0.13);
insert into salesman values(5005,'Pit Alex','London',0.11);
insert into salesman values(5006,'Mc Lyon','Paris',0.14);
insert into salesman values(5003,'Lauson Hen','San Jose',0.12);
insert into salesman values(5007,'Paul Adam','Rome',0.13);
```



```
mysql> 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 | San Jose |       0.12 |
|        5007 | Paul Adam  | Rome     |       0.13 |
+-------------+------------+----------+------------+
6 rows in set (0.00 sec)
```

Query:

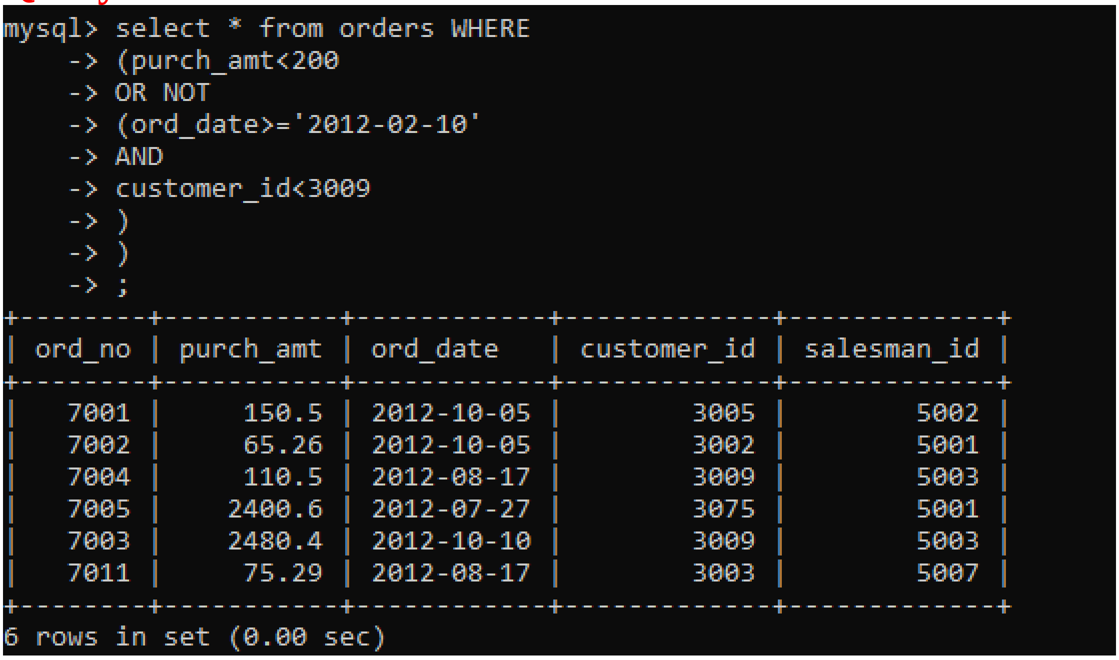
Ans:



```
mysql> select salesman_id,name,city,commission
    -> FROM salesman where
    -> (commission>0.10 AND commission<0.12);
+-------------+------------+----------+------------+
| salesman_id | name       | city     | commission |
+-------------+------------+----------+------------+
|        5005 | Pit Alex   | London   |       0.11 |
|        5003 | Lauson Hen | San Jose |       0.12 |
+-------------+------------+----------+------------+
2 rows in set (0.00 sec)
```

8. From the following table, write a SQL query to find details of all order where purchase amount less than 200 or excluding combination of order date greater than or equal to '2012-02-10' and customer ID less than 3009. Return ord_no, purch_amt, ord_date, customer_id and salesman_id.

Query:



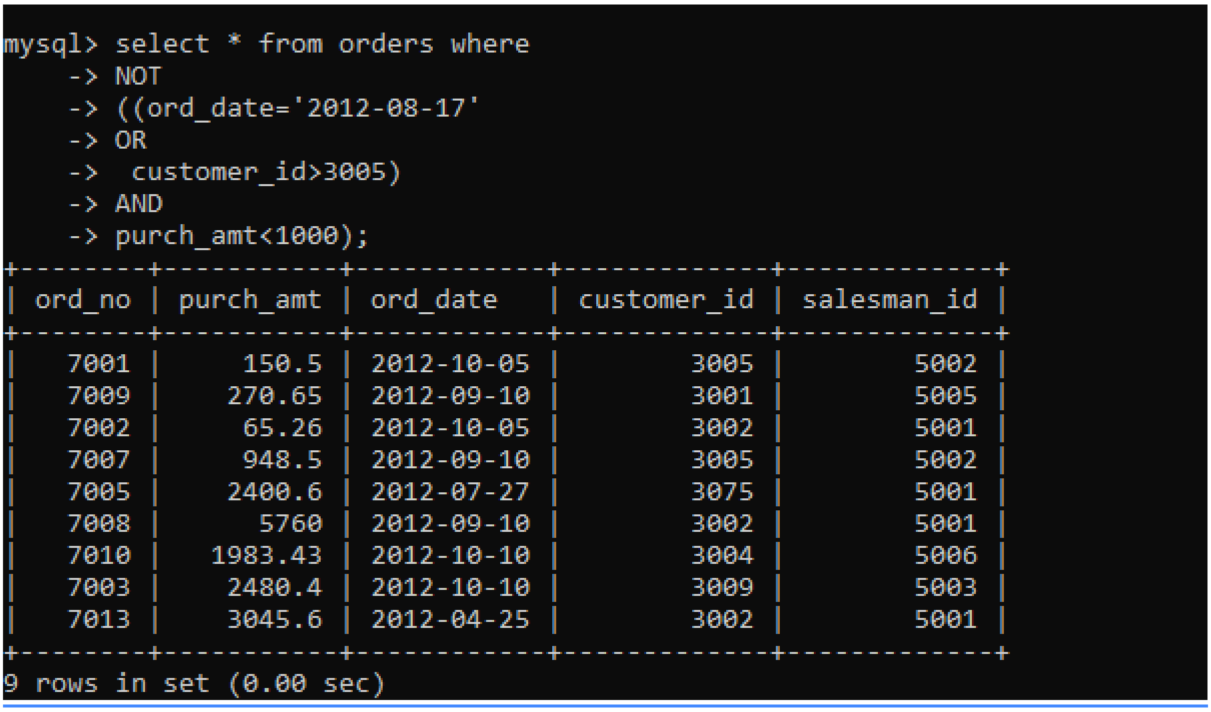
```
mysql> select * from orders WHERE
    -> (purch_amt<200
    -> OR NOT
    -> (ord_date>='2012-02-10'
    -> AND
    -> customer_id<3009
    -> )
    -> )
    -> ;
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
|   7001 |     150.5 | 2012-10-05 |        3005 |        5002 |
|   7002 |     65.26 | 2012-10-05 |        3002 |        5001 |
|   7004 |     110.5 | 2012-08-17 |        3009 |        5003 |
|   7005 |    2400.6 | 2012-07-27 |        3075 |        5001 |
|   7003 |    2480.4 | 2012-10-10 |        3009 |        5003 |
|   7011 |     75.29 | 2012-08-17 |        3003 |        5007 |
+--------+-----------+------------+-------------+-------------+
6 rows in set (0.00 sec)
```

9. From the following table, write a SQL query to find all orders subject to following conditions. Exclude combination of order date equal to '2012-08-17' or customer ID higher than 3005 and purchase amount less than 1000.

Sample table : orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	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:



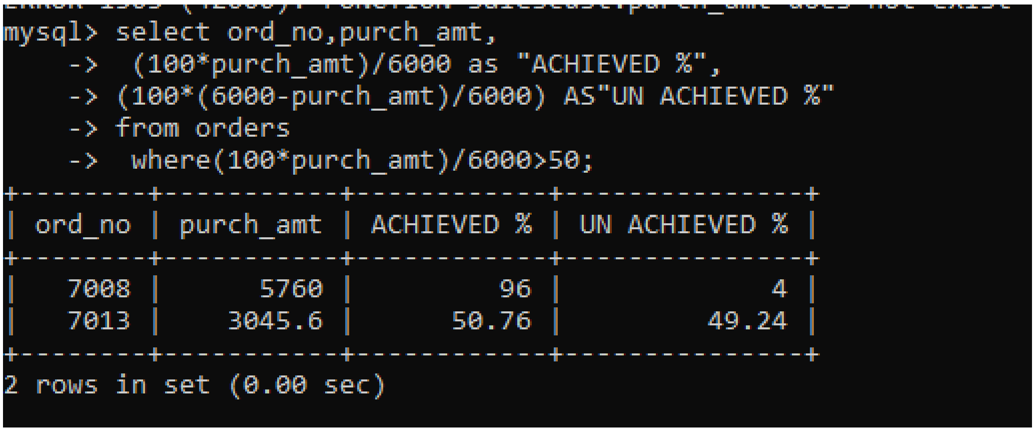
```
mysql> select * from orders where
    -> NOT
    -> ((ord_date='2012-08-17'
    -> OR
    ->  customer_id>3005)
    -> AND
    -> purch_amt<1000);
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
|   7001 |     150.5 | 2012-10-05 |        3005 |        5002 |
|   7009 |    270.65 | 2012-09-10 |        3001 |        5005 |
|   7002 |     65.26 | 2012-10-05 |        3002 |        5001 |
|   7007 |     948.5 | 2012-09-10 |        3005 |        5002 |
|   7005 |    2400.6 | 2012-07-27 |        3075 |        5001 |
|   7008 |      5760 | 2012-09-10 |        3002 |        5001 |
|   7010 |   1983.43 | 2012-10-10 |        3004 |        5006 |
|   7003 |    2480.4 | 2012-10-10 |        3009 |        5003 |
|   7013 |    3045.6 | 2012-04-25 |        3002 |        5001 |
+--------+-----------+------------+-------------+-------------+
9 rows in set (0.00 sec)
```

10. Write a SQL query to display order number, purchase amount, achieved, the unachieved percentage for those order which exceeds the 50% of the target value of 6000.

Sample table: orders

ord_no	purch_amt	ord_date	customer_id	salesman_id
70001	150.5	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:

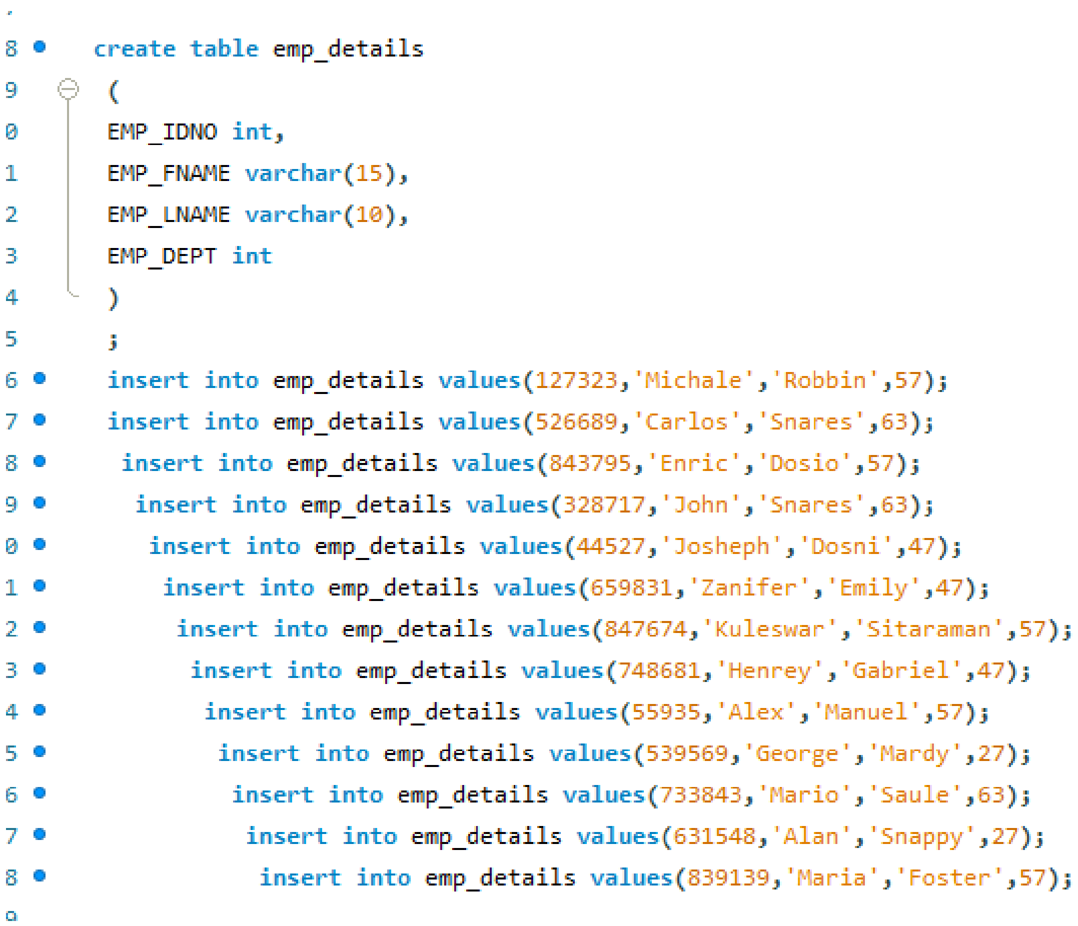


```
mysql> select ord_no,purch_amt,
    ->  (100*purch_amt)/6000 as "ACHIEVED %",
    -> (100*(6000-purch_amt)/6000) AS"UN ACHIEVED %"
    -> from orders
    ->  where(100*purch_amt)/6000>50;
+--------+-----------+------------+---------------+
| ord_no | purch_amt | ACHIEVED % | UN ACHIEVED % |
+--------+-----------+------------+---------------+
|   7008 |      5760 |         96 |             4 |
|   7013 |    3045.6 |      50.76 |         49.24 |
+--------+-----------+------------+---------------+
2 rows in set (0.00 sec)
```

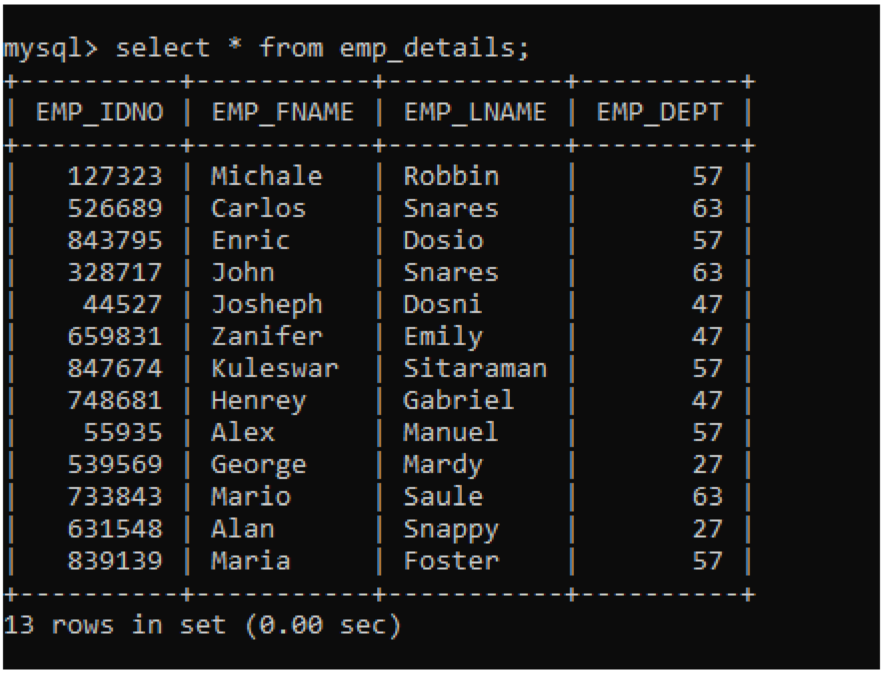
11. From the following table, write a SQL query to find the details of all employees whose last name is ‘Dosni’ or ‘Mardy’. Return emp_idno, emp_fname, emp_lname, and emp_dept.

Sample table : emp_details

```
EMP_IDNO EMP_FNAME       EMP_LNAME         EMP_DEPT
--------- --------------- --------------- ----------
   127323 Michale         Robbin                  57
   526689 Carlos          Snares                  63
   843795 Enric           Dosio                   57
   328717 Jhon            Snares                  63
   444527 Joseph          Dosni                   47
   659831 Zanifer         Emily                   47
   847674 Kuleswar        Sitaraman               57
   748681 Henrey          Gabriel                 47
   555935 Alex            Manuel                  57
   539569 George          Mardy                   27
   733843 Mario           Saule                   63
   631548 Alan            Snappy                  27
   839139 Maria           Foster                  57
```

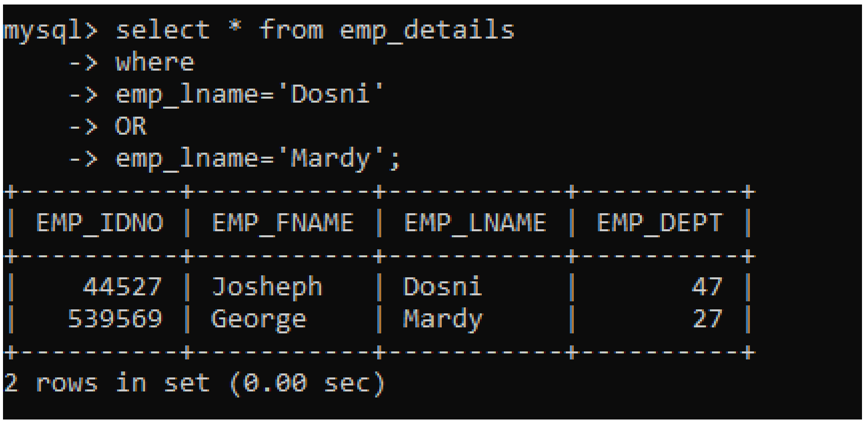


```
create table emp_details
 (
 EMP_IDNO int,
 EMP_FNAME varchar(15),
 EMP_LNAME varchar(10),
 EMP_DEPT int
 )
 ;
 insert into emp_details values(127323,'Michale','Robbin',57);
 insert into emp_details values(526689,'Carlos','Snares',63);
  insert into emp_details values(843795,'Enric','Dosio',57);
   insert into emp_details values(328717,'John','Snares',63);
    insert into emp_details values(44527,'Josheph','Dosni',47);
     insert into emp_details values(659831,'Zanifer','Emily',47);
      insert into emp_details values(847674,'Kuleswar','Sitaraman',57);
       insert into emp_details values(748681,'Henrey','Gabriel',47);
        insert into emp_details values(55935,'Alex','Manuel',57);
         insert into emp_details values(539569,'George','Mardy',27);
          insert into emp_details values(733843,'Mario','Saule',63);
           insert into emp_details values(631548,'Alan','Snappy',27);
            insert into emp_details values(839139,'Maria','Foster',57);
```



```
mysql> select * from emp_details;
+----------+-----------+-----------+----------+
| EMP_IDNO | EMP_FNAME | EMP_LNAME | EMP_DEPT |
+----------+-----------+-----------+----------+
|   127323 | Michale   | Robbin    |       57 |
|   526689 | Carlos    | Snares    |       63 |
|   843795 | Enric     | Dosio     |       57 |
|   328717 | John      | Snares    |       63 |
|    44527 | Josheph   | Dosni     |       47 |
|   659831 | Zanifer   | Emily     |       47 |
|   847674 | Kuleswar  | Sitaraman |       57 |
|   748681 | Henrey    | Gabriel   |       47 |
|    55935 | Alex      | Manuel    |       57 |
|   539569 | George    | Mardy     |       27 |
|   733843 | Mario     | Saule     |       63 |
|   631548 | Alan      | Snappy    |       27 |
|   839139 | Maria     | Foster    |       57 |
+----------+-----------+-----------+----------+
13 rows in set (0.00 sec)
```

Query:



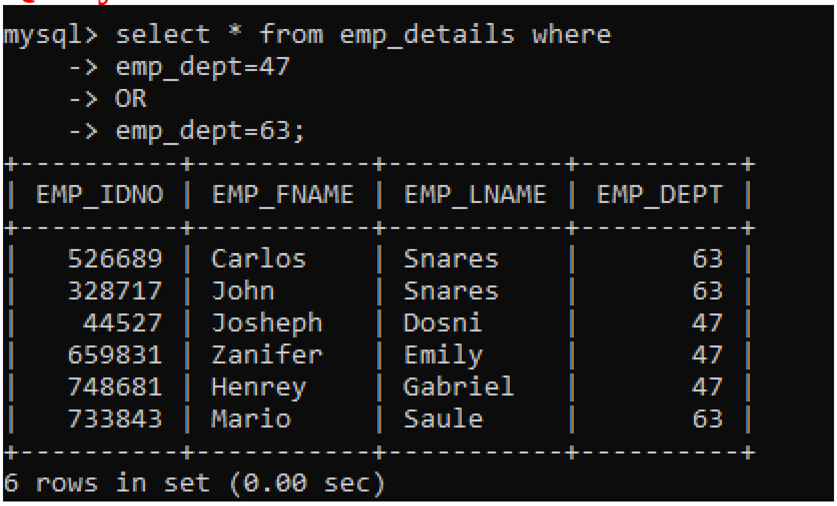
```
mysql> select * from emp_details
    -> where
    -> emp_lname='Dosni'
    -> OR
    -> emp_lname='Mardy';
+----------+-----------+-----------+----------+
| EMP_IDNO | EMP_FNAME | EMP_LNAME | EMP_DEPT |
+----------+-----------+-----------+----------+
|    44527 | Josheph   | Dosni     |       47 |
|   539569 | George    | Mardy     |       27 |
+----------+-----------+-----------+----------+
2 rows in set (0.00 sec)
```

12. From the following table, write a SQL query to find the employees who works at depart 47 or 63. Return emp_idno, emp_fname, emp_lname, and emp_dept.

Sample table : emp_details

EMP_IDNO	EMP_FNAME	EMP_LNAME	EMP_DEPT
127323	Michale	Robbin	57
526689	Carlos	Snares	63
843795	Enric	Dosio	57
328717	Jhon	Snares	63
444527	Joseph	Dosni	47
659831	Zanifer	Emily	47
847674	Kuleswar	Sitaraman	57
748681	Henrey	Gabriel	47
555935	Alex	Manuel	57
539569	George	Mardy	27
733843	Mario	Saule	63
631548	Alan	Snappy	27
839139	Maria	Foster	57

Query:



```
mysql> select * from emp_details where
    -> emp_dept=47
    -> OR
    -> emp_dept=63;
+----------+-----------+-----------+----------+
| EMP_IDNO | EMP_FNAME | EMP_LNAME | EMP_DEPT |
+----------+-----------+-----------+----------+
|   526689 | Carlos    | Snares    |       63 |
|   328717 | John      | Snares    |       63 |
|    44527 | Josheph   | Dosni     |       47 |
|   659831 | Zanifer   | Emily     |       47 |
|   748681 | Henrey    | Gabriel   |       47 |
|   733843 | Mario     | Saule     |       63 |
+----------+-----------+-----------+----------+
6 rows in set (0.00 sec)
```