Name:- RESHMA KEESARI

Roll no. :- 36

PRACTICAL - 01

Aim:- DDL operations on Relational Schema

1. Display name and commission for all the salesman.

```
mysql> select name , commission from salesman;
+------------+------------+
| name       | commission |
+------------+------------+
| James Hoog |       0.15 |
| Nail Knite |       0.13 |
| Lauson Hen |       0.12 |
| Pit Alex   |       0.11 |
| Mc Lyon    |       0.14 |
| Paul Adam  |       0.13 |
+------------+------------+
6 rows in set (0.00 sec)
```

2. Retrieve salesman id of all salesmen from orders table without any repeats.

```
mysql> select distinct salesman_id from orders;
+-------------+
| salesman_id |
+-------------+
|        NULL |
|        5001 |
|        5002 |
|        5006 |
|        5007 |
+-------------+
5 rows in set (0.00 sec)
```

3. Display names and city of salesman, who belongs to the city of Paris.

```
mysql> select name, city from salesman where city='Paris';
+------------+-------+
| name       | city  |
+------------+-------+
| Nail Knite | Paris |
| Mc Lyon    | Paris |
+------------+-------+
2 rows in set (0.00 sec)
```

4. Display all the information for those customers with a grade of 200.

```
mysql> select * from customer where grade= 200;
+-------------+---------------+------------+-------+-------------+
| customer_id | customer_name | city       | grade | salesman_id |
+-------------+---------------+------------+-------+-------------+
|        3003 | Jozy Altidor  | Moncow     |   200 |        5007 |
|        3005 | Graham Zusi   | California |   200 |        5002 |
|        3007 | Brad Davis    | New York   |   200 |        5001 |
+-------------+---------------+------------+-------+-------------+
3 rows in set (0.00 sec)
```

5. Display the order number, order date and the purchase amount for order(s) which will be delivered by the salesman with ID 5001 .

```
mysql> select order_no, order_date, purch_amt from orders where salesman_id = 5001;
+----------+------------+-----------+
| order_no | order_date | purch_amt |
+----------+------------+-----------+
|    70002 | 2016-10-05 |        65 |
|    70005 | 2016-07-27 |      2401 |
|    70008 | 2016-09-10 |      5760 |
+----------+------------+-----------+
3 rows in set (0.00 sec)
```

6. Display all the customers, who are either belongs to the city New York or not had a grade above 100.

```
mysql> select * from customer where city= 'New York' and grade <= 100;
+-------------+---------------+----------+-------+-------------+
| customer_id | customer_name | city     | grade | salesman_id |
+-------------+---------------+----------+-------+-------------+
|        3002 | Nick Rimando  | New York |   100 |        5001 |
+-------------+---------------+----------+-------+-------------+
1 row in set (0.00 sec)
```

7. Find those salesmen with all information who gets the commission within a range of 0.12 and 0.14.

```
mysql> select * from salesman where commission between 0.12 and 0.14;
+-------------+------------+-------+------------+
| salesman_id | name       | city  | commission |
+-------------+------------+-------+------------+
|        5002 | Nail Knite | Paris |       0.13 |
|        5007 | Paul Adam  | Rome  |       0.13 |
+-------------+------------+-------+------------+
2 rows in set (0.00 sec)
```

8. Find all those customers with all information whose names are ending with the letter 'n'.

```
mysql> select * from customer where customer_name like '%n';
+-------------+---------------+--------+-------+-------------+
| customer_id | customer_name | city   | grade | salesman_id |
+-------------+---------------+--------+-------+-------------+
|        3001 | Brad Guzan    | London |  NULL |        NULL |
|        3008 | Julian Green  | London |   300 |        5002 |
+-------------+---------------+--------+-------+-------------+
2 rows in set (0.00 sec)
```

9. Find those salesmen with all information whose name containing the 1st character is 'N' and the 4th character is 'l' and rests may be any character.

```
mysql> select * from salesman where name like 'N__l%';
+-------------+------------+-------+------------+
| salesman_id | name       | city  | commission |
+-------------+------------+-------+------------+
|        5002 | Nail Knite | Paris |       0.13 |
+-------------+------------+-------+------------+
1 row in set (0.00 sec)
```

10. Find that customer with all information who does not get any grade except NULL.

```
mysql> select * from customer where grade is null;
+-------------+---------------+--------+-------+-------------+
| customer_id | customer_name | city   | grade | salesman_id |
+-------------+---------------+--------+-------+-------------+
|        3001 | Brad Guzan    | London |  NULL |        NULL |
+-------------+---------------+--------+-------+-------------+
1 row in set (0.00 sec)
```

11. Find the total purchase amount of all orders.

```
mysql> select sum(purch_amt) as TotalPurchaseAmount from orders;
+---------------------+
| TotalPurchaseAmount |
+---------------------+
|               14496 |
+---------------------+
1 row in set (0.00 sec)
```

12. Find the number of salesman currently listing for all of their customers.

```
mysql>  select count(salesman_id) from orders;
+--------------------+
| count(salesman_id) |
+--------------------+
|                  8 |
+--------------------+
1 row in set (0.00 sec)

mysql> select count(Distinct salesman_id) from orders;
+-----------------------------+
| count(Distinct salesman_id) |
+-----------------------------+
|                           4 |
+-----------------------------+
1 row in set (0.00 sec)
```

13. Find the highest grade for each of the cities of the customers.

```
mysql> select city, max(grade) as highest_grade from customer group by city;
+------------+---------------+
| city       | highest_grade |
+------------+---------------+
| London     |           300 |
| New York   |           200 |
| Moncow     |           200 |
| Paris      |           300 |
| California |           200 |
| Berlin     |           100 |
+------------+---------------+
6 rows in set (0.01 sec)
```

14. Find the highest purchase amount ordered by each customer with their ID and highest purchase amount.

```
mysql> select customer_id, max(purch_amt) as highest_purchaseAmount from orders group by customer_id;
+-------------+------------------------+
| customer_id | highest_purchaseAmount |
+-------------+------------------------+
|        3001 |                    271 |
|        3002 |                   5760 |
|        3003 |                     75 |
|        3004 |                   1983 |
|        3005 |                    949 |
|        3007 |                   2401 |
|        3008 |                    250 |
|        3009 |                   2480 |
+-------------+------------------------+
8 rows in set (0.00 sec)
```

15. Find the highest purchase amount ordered by each customer on a particular date with their ID, order date and highest purchase amount.

```
mysql> select customer_id,order_date,max(purch_amt) from orders group by customer_id,order_date;
+-------------+------------+-----------------+
| customer_id | order_date | max(purch_amt) |
+-------------+------------+-----------------+
|        3005 | 2016-10-05 |             151 |
|        3002 | 2016-10-05 |              65 |
|        3009 | 2016-10-10 |            2480 |
|        3009 | 2016-08-17 |             111 |
|        3007 | 2016-07-27 |            2401 |
|        3005 | 2016-09-10 |             949 |
|        3002 | 2016-09-10 |            5760 |
|        3001 | 2016-09-10 |             271 |
|        3004 | 2016-10-10 |            1983 |
|        3003 | 2016-08-17 |              75 |
|        3008 | 2016-06-27 |             250 |
+-------------+------------+-----------------+
11 rows in set (0.00 sec)
```

16. Find the highest purchase amount on a date '2016-08-17' for each salesman with their ID.

```
mysql> select salesman_id, max(purch_amt) as highest_purchaseAmount from orders where order_date='2016-08-17' group by s
alesman_id;
+-------------+------------------------+
| salesman_id | highest_purchaseAmount |
+-------------+------------------------+
|        NULL |                    111 |
|        5007 |                     75 |
+-------------+------------------------+
2 rows in set (0.00 sec)
```

17. Find the highest purchase amount with their customer ID and order date, for only those customers who have the highest purchase amount in a day is more than 2000.

```
mysql> select customer_id,order_date,max(purch_amt) from orders group by customer_id,order_date having max(purch_amt)>20
00;
+-------------+------------+-----------------+
| customer_id | order_date | max(purch_amt) |
+-------------+------------+-----------------+
|        3009 | 2016-10-10 |            2480 |
|        3007 | 2016-07-27 |            2401 |
|        3002 | 2016-09-10 |            5760 |
+-------------+------------+-----------------+
3 rows in set (0.01 sec)
```

18. Write a SQL statement that counts all orders for a date August 17th, 2016.

```
mysql> select count(*) from orders where order_date='2016-08-17';
+----------+
| count(*) |
+----------+
|        2 |
+----------+
1 row in set (0.00 sec)
```