Practical-1 : DDL operations on Relational Schema

```
>create database prac1
>use prac1
Database changed

mysql> CREATE TABLE salesman(salesman_id INTEGER PRIMARY KEY,name TEXT,city TEXT,comission float);
Query OK, 0 rows affected (0.01 sec)

mysql> INSERT INTO salesman VALUES (5001, 'James Hoog', 'New York', 0.15);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO salesman VALUES (5002, 'Nail Knite', 'Paris' ,0.13);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO salesman VALUES (5005, 'Pit Alex','London', 0.11);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO salesman VALUES (5006,  'Mc Lyon', 'Paris' ,0.14);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO salesman VALUES (5003,  'Lauson Hen', '' ,0.12);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO salesman VALUES (5007,  'Paul Adam', 'Rome' , 0.13);
Query OK, 1 row affected (0.00 sec)

mysql> select * from salesman;
+-------------+------------+----------+-----------+
| salesman_id | name       | city     | comission |
+-------------+------------+----------+-----------+
|        5001 | James Hoog | New York |      0.15 |
|        5002 | Nail Knite | Paris    |      0.13 |
```

```
|       5003 | Lauson Hen |          |      0.12 |
|       5005 | Pit Alex   | London   |      0.11 |
|       5006 | Mc Lyon    | Paris    |      0.14 |
|       5007 | Paul Adam  | Rome     |      0.13 |
+-------------+------------+----------+-----------+
```

6 rows in set (0.00 sec)

```
CREATE TABLE customer(customer_id int primary key,customer_name text,city text,grade int,salesman_id
int,foreign key(salesman_id) references salesman(salesman_id));
```

Query OK, 0 rows affected (0.02 sec)

```
mysql> INSERT INTO customer VALUES ( 3002, 'Nick Rimando' ,'New York',  100 ,5001);
```

Query OK, 1 row affected (0.01 sec)

```
mysql> INSERT INTO customer VALUES (3005, 'Graham Zusi', 'California' ,200 ,5002);
```

Query OK, 1 row affected (0.00 sec)

```
mysql> INSERT INTO customer VALUES (3001, 'Brad Guzan', 'London',null,null);
```

Query OK, 1 row affected (0.00 sec)

```
mysql> INSERT INTO customer VALUES (3004, 'Fabian Johnson' ,'Paris' , 300, 5006);
```

Query OK, 1 row affected (0.00 sec)

```
mysql> INSERT INTO customer VALUES (3007, 'Brad Davis',  'New York' , 200, 5001);
```

Query OK, 1 row affected (0.01 sec)

```
mysql> INSERT INTO customer VALUES (3009, 'Geoff Cameron', 'Berlin',  100, null);
```

Query OK, 1 row affected (0.00 sec)

```
mysql> INSERT INTO customer VALUES (3008 ,'Julian Green', 'London' , 300, 5002);
```

Query OK, 1 row affected (0.00 sec)

```
mysql> INSERT INTO customer VALUES (3003 ,'Jozy Altidor', 'Moscow', 200,  5007);
Query OK, 1 row affected (0.00 sec)

mysql> select * from customer;
+-------------+---------------+------------+-------+-------------+
| customer_id | customer_name  | city       | grade | salesman_id |
+-------------+---------------+------------+-------+-------------+
|       3001 | Brad Guzan     | London     |  NULL |       NULL |
|       3002 | Nick Rimando   | New York   |   100 |       5001 |
|       3003 | Jozy Altidor   | Moscow     |   200 |       5007 |
|       3004 | Fabian Johnson | Paris      |   300 |       5006 |
|       3005 | Graham Zusi    | California |   200 |       5002 |
|       3007 | Brad Davis     | New York   |   200 |       5001 |
|       3008 | Julian Green   | London     |   300 |       5002 |
|       3009 | Geoff Cameron  | Berlin     |   100 |       NULL |
+-------------+---------------+------------+-------+-------------+
```

```
CREATE TABLE orders (ord_no INTEGER PRIMARY KEY,purch_amt float,ord_date date,customer_id
INTEGER,foreign key(customer_id) references customer(customer_id),salesman_id INTEGER,foreign
key(salesman_id) references salesman(salesman_id));
Query OK, 0 rows affected (0.02 sec)

mysql> INSERT INTO orders VALUES (70001, 150.5, '2012-10-05', 3005, 5002);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70009, 270.65, '2012-09-10', 3001, null);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70002, 65.26 ,'2012-10-05' ,3002, 5001);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70004, 110.5, '2012-08-17', 3009, null);
```

```
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70007, 948.5, '2012-09-10', 3005, 5002);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70005, 2400.6, '2012-07-27', 3007, 5001);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70008, 5760, '2012-09-10', 3002, 5001);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70010, 1983.43, '2012-10-10', 3004, 5006);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70003, 2480.4, '2012-10-10', 3009, null);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70012, 250.45, '2012-06-27', 3008, 5002);
Query OK, 1 row affected (0.00 sec)

mysql> INSERT INTO orders VALUES (70011, 75.29, '2012-08-17', 3003, 5007);
Query OK, 1 row affected (0.00 sec)

mysql> select * from orders;
+--------+-----------+------------+-------------+-------------+
| ord_no | purch_amt | ord_date   | customer_id | salesman_id |
+--------+-----------+------------+-------------+-------------+
| 70001 |    150.5 | 2012-10-05 |       3005 |       5002 |
| 70002 |    65.26 | 2012-10-05 |       3002 |       5001 |
| 70003 |   2480.4 | 2012-10-10 |       3009 |       NULL |
| 70004 |    110.5 | 2012-08-17 |       3009 |       NULL |
| 70005 |   2400.6 | 2012-07-27 |       3007 |       5001 |
| 70007 |    948.5 | 2012-09-10 |       3005 |       5002 |
```

| 70008 | 5760 | 2012-09-10 | 3002 | 5001 |

| 70009 | 270.65 | 2012-09-10 | 3001 | NULL |

| 70010 | 1983.43 | 2012-10-10 | 3004 | 5006 |

| 70011 | 75.29 | 2012-08-17 | 3003 | 5007 |

| 70012 | 250.45 | 2012-06-27 | 3008 | 5002 |

+--------+-----------+------------+-------------+-------------+

11 rows in set (0.00 sec)

Queries:-

1. Display name and commission for all the salesmen

```
mysql> select name , comission from salesman;
+------------+-----------+
| name       | comission |
+------------+-----------+
| 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 (0.01 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 (0.01 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 (0.02 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  | Moscow     |   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 ord_no , ord_date , purch_amt from orders where salesman_id = 5001;
+--------+------------+-----------+
| ord_no | ord_date   | purch_amt |
+--------+------------+-----------+
|  70002 | 2012-10-05 |     65.26 |
|  70005 | 2012-07-27 |    2400.6 |
|  70008 | 2012-09-10 |      5760 |
+--------+------------+-----------+
```

3 rows in set (0.00 sec)

12. 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' or not grade >100;
+-------------+---------------+----------+-------+-------------+
| customer_id | customer_name | city     | grade | salesman_id |
+-------------+---------------+----------+-------+-------------+
|        3002 | Nick Rimando  | New York |   100 |        5001 |
|        3007 | Brad Davis    | New York |   200 |        5001 |
|        3009 | Geoff Cameron | Berlin   |   100 |        NULL |
+-------------+---------------+----------+-------+-------------+
3 rows in set (0.06 sec)
```

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

```
mysql> select * from salesman where comission between 0.12 AND 0.14;
+-------------+------------+-------+-----------+
| salesman_id | name       | city  | comission |
+-------------+------------+-------+-----------+
|        5002 | Nail Knite | Paris |      0.13 |
|        5007 | Paul Adam  | Rome  |      0.13 |
+-------------+------------+-------+-----------+
2 rows in set (0.01 sec)
```

14. 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 |
|        3004 | Fabian Johnson | Paris  |   300 |        5006 |
|        3008 | Julian Green   | London |   300 |        5002 |
|        3009 | Geoff Cameron  | Berlin |   100 |        NULL |
+-------------+---------------+--------+-------+-------------+
4 rows in set (0.02 sec)
```

15. 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  | comission |
+-------------+------------+-------+-----------+
|        5002 | Nail Knite | Paris |      0.13 |
+-------------+------------+-------+-----------+
1 row in set (0.00 sec)
```

16. 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)
```

17. Find the total purchase amount of all orders

```
mysql> select sum(purch_amt) from orders;
+--------------------+
| sum(purch_amt)     |
+--------------------+
| 14495.580047607422 |
+--------------------+
1 row in set (0.03 sec)

mysql>
```

18. 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 (0.01 sec)
```

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

```
mysql> select city,max(grade) from customer Group by city;
+------------+------------+
| city       | max(grade) |
+------------+------------+
| Berlin     |        100 |
| California |        200 |
| London     |        300 |
| Moscow     |        200 |
| New York   |        200 |
| Paris      |        300 |
+------------+------------+
6 rows in set (0.08 sec)
```

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

```
mysql> select customer_id,max(purch_amt) from orders group by customer_id;
+-------------+----------------+
| customer_id | max(purch_amt) |
+-------------+----------------+
|        3001 |         270.65 |
|        3002 |           5760 |
|        3003 |          75.29 |
|        3004 |        1983.43 |
|        3005 |          948.5 |
|        3007 |         2400.6 |
|        3008 |         250.45 |
|        3009 |         2480.4 |
```

```
+-------------+----------------+
8 rows in set (0.00 sec)
```

21. 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,ord_date,max(purch_amt) from orders group by customer_id,ord_date;
+-------------+------------+----------------+
| customer_id | ord_date   | max(purch_amt) |
+-------------+------------+----------------+
|        3005 | 2012-10-05 |          150.5 |
|        3002 | 2012-10-05 |          65.26 |
|        3009 | 2012-10-10 |         2480.4 |
|        3009 | 2012-08-17 |          110.5 |
|        3007 | 2012-07-27 |         2400.6 |
|        3005 | 2012-09-10 |          948.5 |
|        3002 | 2012-09-10 |           5760 |
|        3001 | 2012-09-10 |         270.65 |
|        3004 | 2012-10-10 |        1983.43 |
|        3003 | 2012-08-17 |          75.29 |
|        3008 | 2012-06-27 |         250.45 |
+-------------+------------+----------------+
11 rows in set (0.00 sec)
```

22. Find the highest purchase amount on a date '2012-08-17' for each salesman with their ID.

```
mysql> select salesman_id,max(purch_amt) from orders where ord_date='2012-08-17' group by salesman_id;
+-------------+----------------+
| salesman_id | max(purch_amt) |
+-------------+----------------+
|        NULL |          110.5 |
|        5007 |          75.29 |
```

```
+-------------+----------------+
2 rows in set (0.00 sec)
```

23. 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,ord_date,max(purch_amt) from orders group by customer_id,ord_date having max(purch_amt)>2000;

```
+-------------+------------+----------------+
| customer_id | ord_date   | max(purch_amt) |
+-------------+------------+----------------+
|       3009 | 2012-10-10 |        2480.4 |
|       3007 | 2012-07-27 |        2400.6 |
|       3002 | 2012-09-10 |         5760 |
+-------------+------------+----------------+
3 rows in set (0.00 sec)
```

24. Write a SQL statement that counts all orders for a date August 17th, 2012.

```
mysql> select count(*) from orders where ord_date='2012-08-17';
+----------+
| count(*) |
+----------+
|       2 |
+----------+
1 row in set (0.00 sec)
```