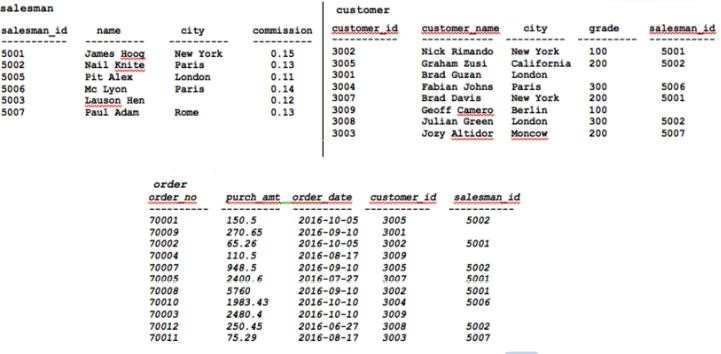
**PRACTICAL N0 – 1**

**Aim:** DDL operations on Relational Schema

Design the following schema and execute the following queries on it:



**Code:**

create database salesman; use salesman

CREATE TABLE salesman(salesman\_id INT NOT NULL AUTO\_INCREMENT PRIMARY key,

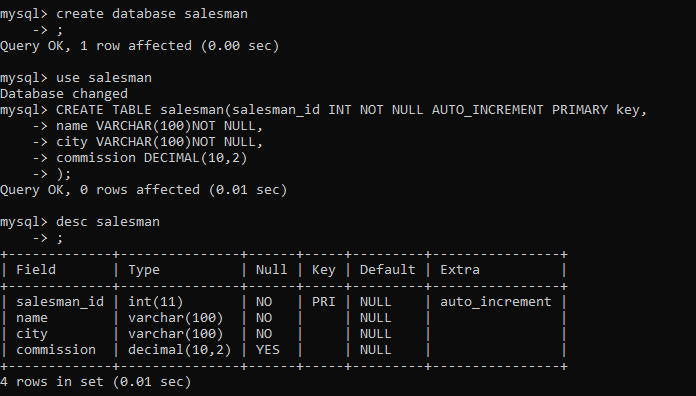
name VARCHAR(100)NOT NULL, city VARCHAR(100)NOT NULL,

commission DECIMAL(10,2)

);

desc salesman;

**Output:**

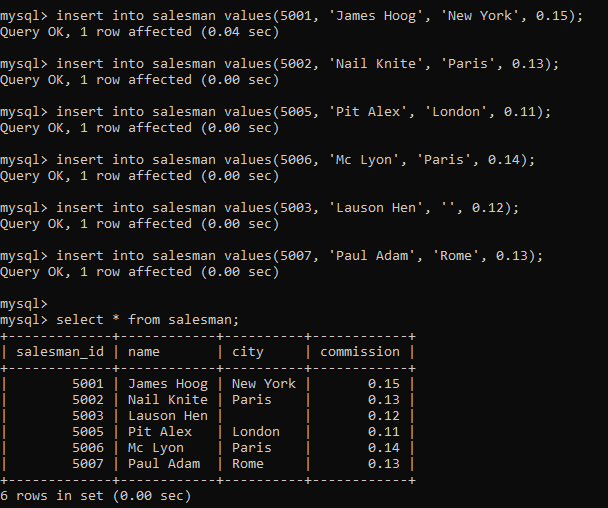
****

**Code:**

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', '', 0.12); insert into salesman values(5007, 'Paul Adam', 'Rome', 0.13); select \* from salesman;

**Output:**

****

**Code:**

CREATE TABLE customer(customer\_id INT NOT NULL AUTO\_INCREMENT PRIMARY key,

customer\_name VARCHAR(100)NOT NULL, city VARCHAR(100)NOT NULL,

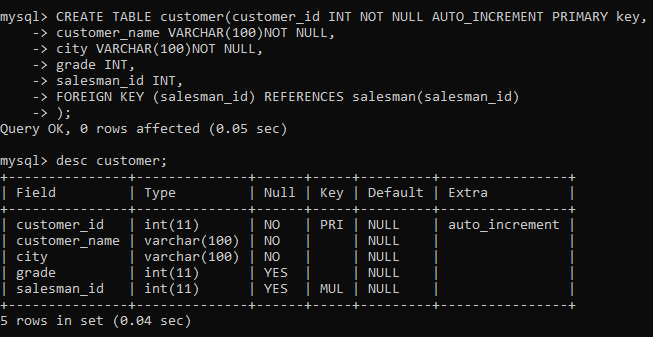
grade INT, salesman\_id INT,

FOREIGN KEY (salesman\_id) REFERENCES salesman(salesman\_id)

);

desc customer;

**Output:**

****

**Code:**

insert into customer values(3002, 'Nick Rimando', 'New York', 100, 5001);

insert into customer values(3005, 'Graham Zusi', 'California', 200, 5002);

insert into customer values(3001, 'Brad Guzan', 'Londan', 100, 5005);

insert into customer values(3004, 'Fabian Johns', 'Paris', 300, 5006);

insert into customer values(3007, 'Brad Davis', 'New York', 200, 5001);

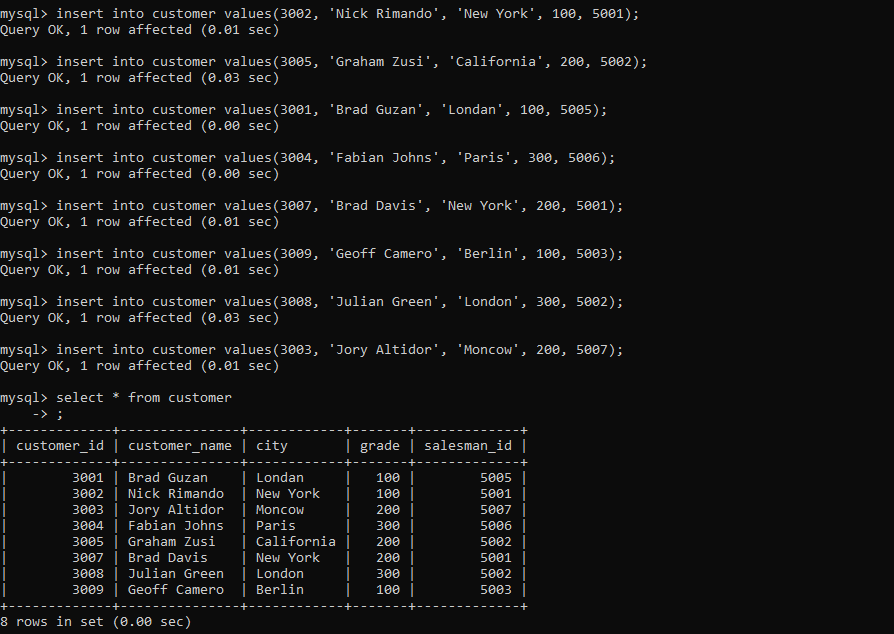
insert into customer values(3009, 'Geoff Camero', 'Berlin', 100, 5003);

insert into customer values(3008, 'Julian Green', 'London', 300, 5002);

insert into customer values(3003, 'Jory Altidor', 'Moncow', 200, 5007);

select \* from customer;

**Output:**

****

**Code:**

CREATE TABLE orders(order\_no INT NOT NULL AUTO\_INCREMENT PRIMARY key,

purch\_amt DECIMAL(10,2) NOT NULL, order\_date DATE NOT NULL, customer\_id INT,

salesman\_id INT,

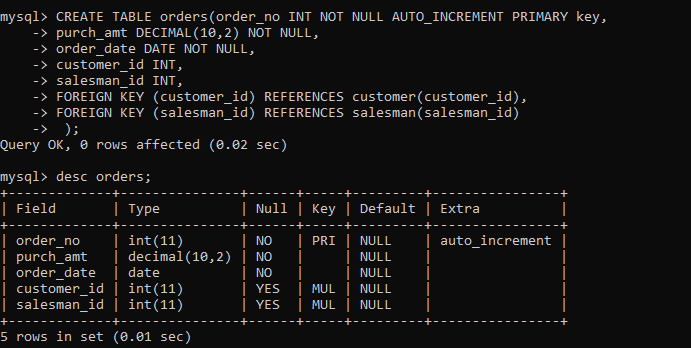
FOREIGN KEY (customer\_id) REFERENCES customer(customer\_id),

FOREIGN KEY (salesman\_id) REFERENCES salesman(salesman\_id)

);

desc orders;

**Output:**

****

**Code:**

insert into orders values(70001, 150.5, '2016-10-05', 3005, 5002);

insert into orders values(70009, 270.65, '2016-09-10', 3001, NULL);

insert into orders values(70002, 65.26, '2016-10-05', 3002, 5001);

insert into orders values(70004, 110.5, '2016-08-17', 3009, NULL);

insert into orders values(70007, 948.5, '2016-09-10', 3005, 5002);

insert into orders values(70005, 2400.6, '2016-07-27', 3007, 5001);

insert into orders values(70008, 5760, '2016-09-10', 3002, 5001);

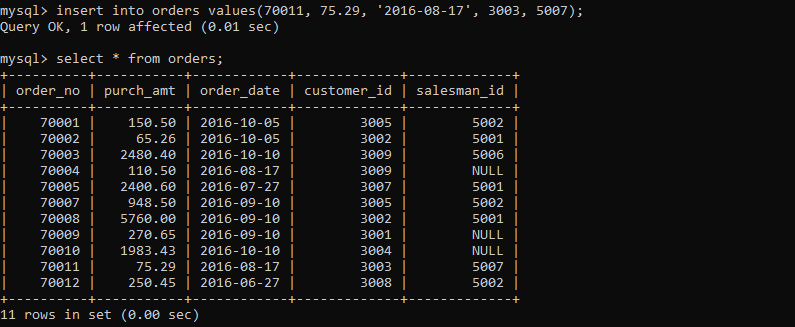
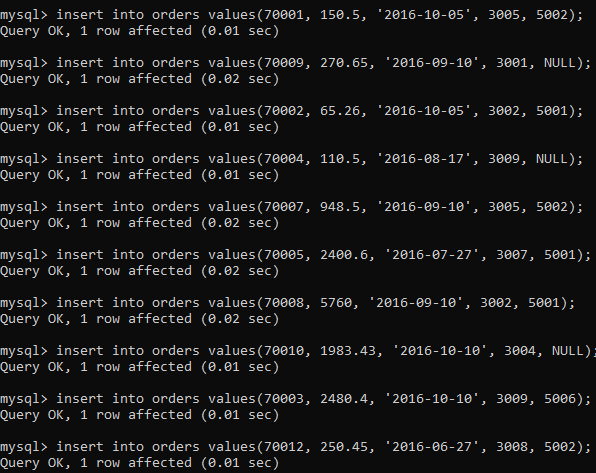
insert into orders values(70010, 1983.43, '2016-10-10', 3004, NULL);

insert into orders values(70003, 2480.4, '2016-10-10', 3009, 5006);

insert into orders values(70012, 250.45, '2016-06-27', 3008, 5002);

insert into orders values(70011, 75.29, '2016-08-17', 3003, 5007); select \* from orders;

**Output:**

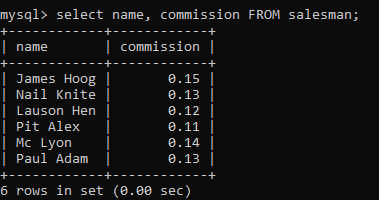
****

1. Display name and commission for all the salesmen.

**Code:**

select name, commission FROM salesman;

**Output:**

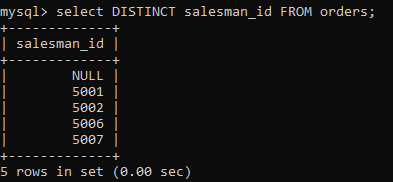
****

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

**Code:**

select DISTINCT salesman\_id FROM orders;

**Output:**

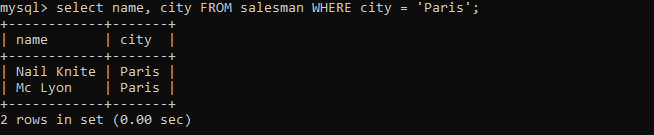
****

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

**Code:**

select name, city FROM salesman WHERE city = 'Paris';

**Output:**

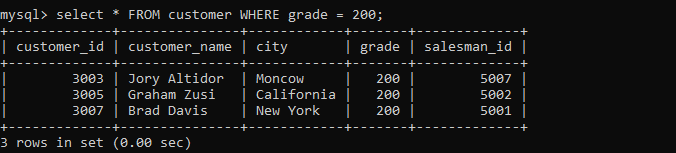
****

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

**Code:**

select \* FROM customer WHERE grade = 200;

**Output:**

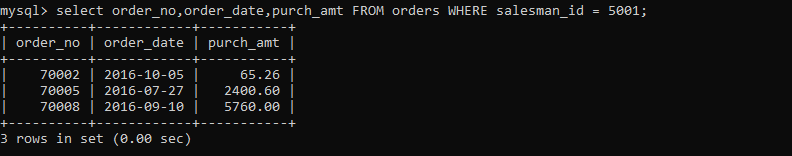


1. Display the order number, order date and the purchase amount for order(s) which will be delivered by the salesman with ID 5001

**Code:**

select order\_no,order\_date,purch\_amt FROM orders WHERE salesman\_id = 5001;

**Output:**

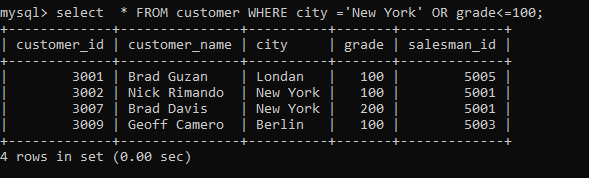


1. Display all the customers, who are either belongs to the city New York or not had a grade above 100.

**Code:**

select \* FROM customer WHERE city ='New York' OR grade<=100;

**Output:**

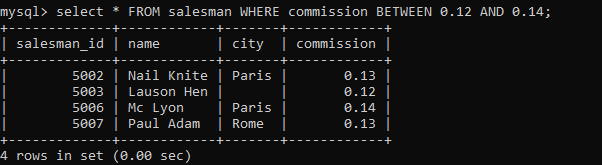


1. Find those salesmen with all information who gets the commission within a range of 0.12 and 0.14.

**Code:**

select \* FROM salesman WHERE commission BETWEEN 0.12 AND 0.14;

**Output:**

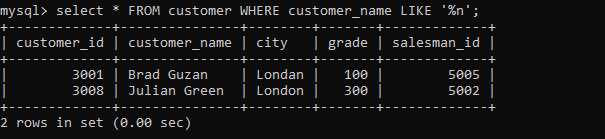


1. Find all those customers with all information whose names are ending with the letter 'n'.

**Code:**

select \* FROM customer WHERE customer\_name LIKE '%n';

**Output:**

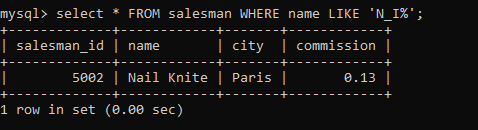


1. Find those salesmen with all information whose name containing the 1st character is 'N' and the 4th character is 'l' and rests may be any character.

**Code:**

select \* FROM salesman WHERE name LIKE 'N\_I%';

**Output:**



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

**Code:**

select \* FROM customer WHERE grade is NULL;

**Output:**

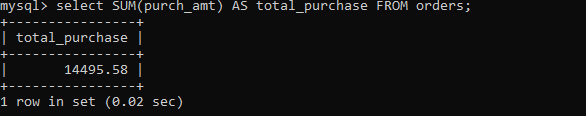


1. Find the total purchase amount of all orders.

**Code:**

select SUM(purch\_amt) AS total\_purchase FROM orders;

**Output:**

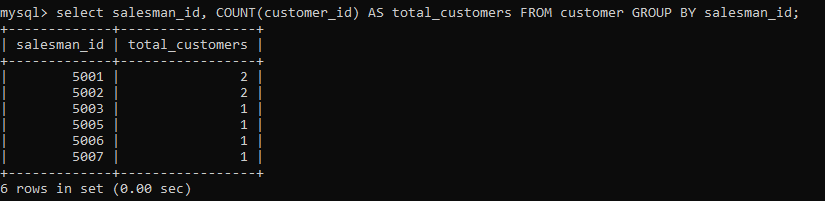


1. Find the number of salesman currently listing for all of their customers.

**Code:**

select salesman\_id, COUNT(customer\_id) AS total\_customers FROM customer GROUP BY salesman\_id;

**Output:**

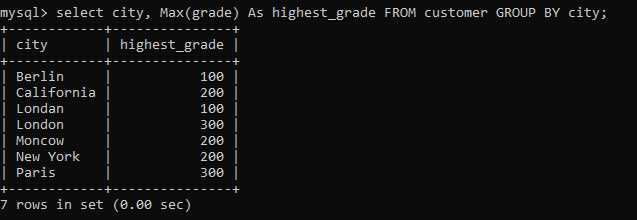


1. Find the highest grade for each of the cities of the customers.

**Code:**

select city, Max(grade) As highest\_grade FROM customer GROUP BY city;

**Output:**

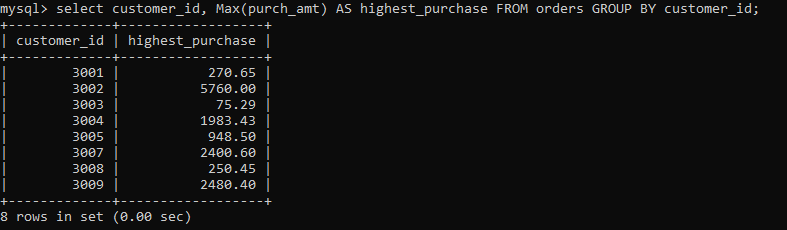


1. Find the highest purchase amount ordered by each customer with their ID and highest purchase amount.

**Code:**

select customer\_id, Max(purch\_amt) AS highest\_purchase FROM orders GROUP BY customer\_id;

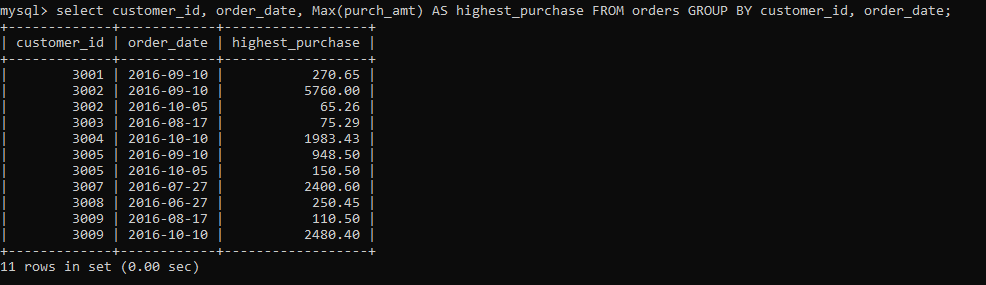
**Output:**



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

**Code:**

select customer\_id, order\_date, Max(purch\_amt) AS highest\_purchase FROM orders GROUP BY customer\_id, order\_date; **Output:**



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

**Code:**

select salesman\_id, MAX(purch\_amt) AS highest\_purchase FROM orders WHERE order\_date = 2012-08-17 GROUP BY salesman\_id; **Output:**

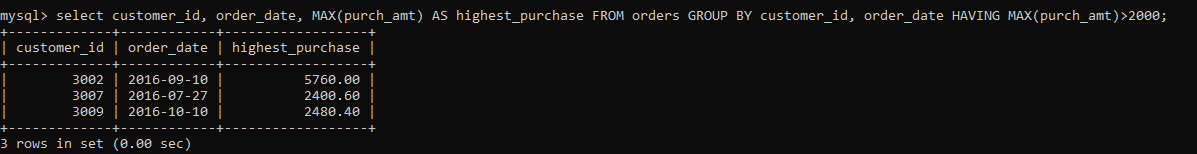


1. Find the highest purchase amount with their customer ID and order date, for only those customers who have the highest purchase amount in a day is more than 2000.

**Code:**

select customer\_id, order\_date, MAX(purch\_amt) AS highest\_purchase FROM orders GROUP BY customer\_id, order\_date HAVING MAX(purch\_amt)>2000;

**Output:**



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

**Code:**

select COUNT(\*) AS total\_orders FROM orders WHERE order\_date = 2012-08-17;

**Output:**

