**PRACTICAL NO. 1**

**AIM: DDL operations on Relational Schema**

create table salesman(

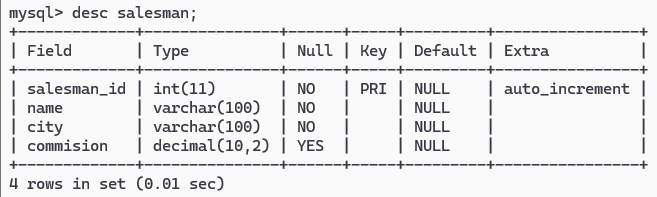
-> salesman\_id INT NOT NULL AUTO\_INCREMENT PRIMARY KEY,

-> name VARCHAR(100) NOT NULL,

-> city VARCHAR(100) NOT NULL,

-> commision DECIMAL(10,2)

-> );



create table customer(

-> customer\_id INT AUTO\_INCREMENT PRIMARY KEY,

-> customer\_name VARCHAR(100) NOT NULL,

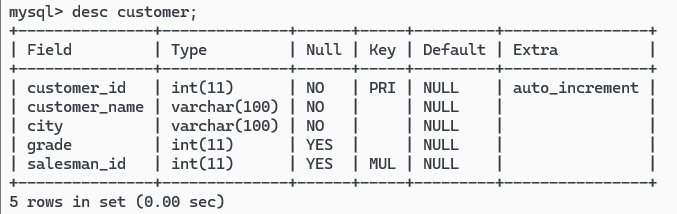
-> city VARCHAR(100) NOT NULL,

-> grade INT,

-> salesman\_id INT,

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

-> );



create table orders(

-> order\_no INT AUTO\_INCREMENT PRIMARY KEY,

-> purch\_amt DECIMAL(10,2) NOT NULL,

-> order\_date DATE NOT NULL,

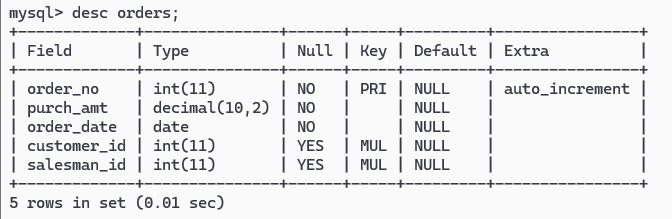
-> customer\_id INT,

-> salesman\_id INT,

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

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

-> );



**Values of salesman**

insert into salesman values(5001, 'James Hoog', 'New York', 0.15);

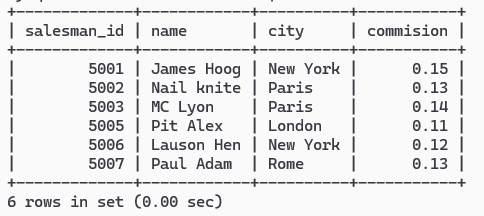
insert into salesman values(5002, 'Nail knite', 'Paris', 0.13);

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);



**Values of customer**

insert into customer values(3001, 'Brad Guzan', 'London', NULL,5003);

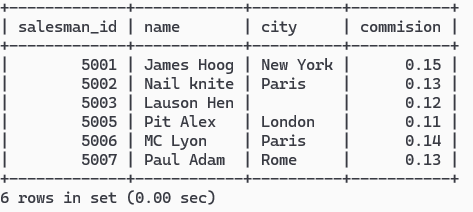
insert into customer values(3004, 'Fabian Johns', 'Paris', 300,5006);

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, 'Jozy Altidor', 'Moncow', 200,5007);



**Values of orders**

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

insert into orders values(70009, 270.65, '2016-09-10',3001,5003);

insert into orders values(70002, 65.26, '2016-10-15',3002,5001);

insert into orders values(70004, 110.5, '2016-08-17',3009,5003);

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

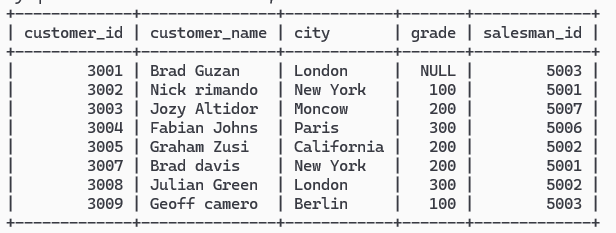
insert into orders values(70005, 2400.6, '2016-07-27',3007,5001);

insert into orders values(700010, 1983.43, '2016-10-10',3004,5006);

insert into orders values(70003, 2480.4, '2016-10-10',3009,5003);

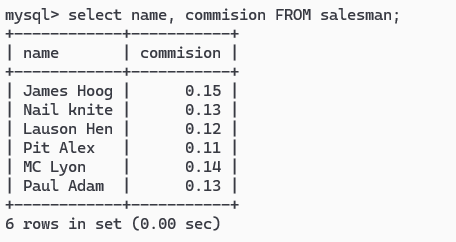
insert into orders values(700012, 250.45, '2016-06-27',3008,5002);

insert into orders values(700011, 75.29, '2016-08-17',3003,5007);



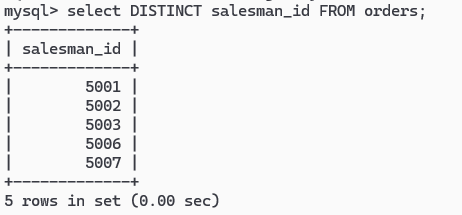
**1. Display name and commission for all the salesmen.**

select name, commision FROM salesman;



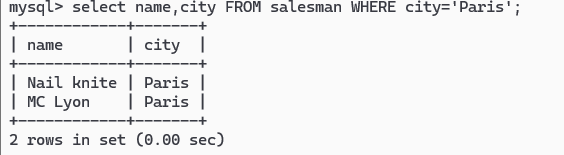
**2. Retrieve salesman id of all salesmen from orders table without any repeats.**

select DISTINCT salesman\_id FROM orders;



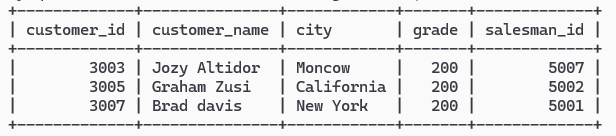
**3. Display names and city of salesman, who belongs to the city of Paris.**

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



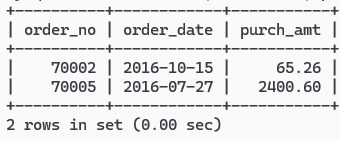
**4. Display all the information for those customers with a grade of 200.**

select \* from customer WHERE grade=200;



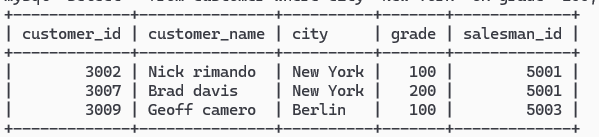
**5. Display the order number, order date and the purchase amount for order(s) which will be delivered by the salesman with ID 5001.**

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



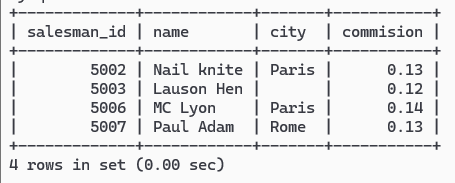
**6. Display all the customers, who are either belongs to the city New York or not had a grade above 100.**

select \* from customer where city='New York' OR grade<=100;



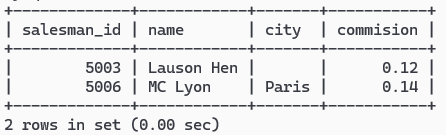
**7. Find those salesmen with all information who gets the commission within a range of 0.12 and 0.14.**

select \* from salesman WHERE commision BETWEEN 0.12 AND 0.14;



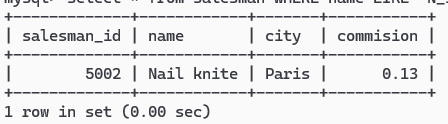
**8. Find all those customers with all information whose names are ending with the letter 'n'.**

select \* from salesman WHERE name LIKE '%n';



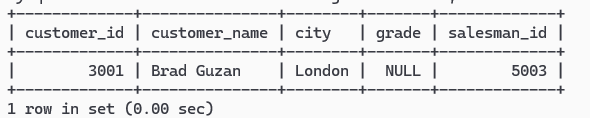
**9. Find those salesmen with all information whose name containing the 1st character is 'N' and the 4th character is 'l' and rests may be any character.**

select \* from salesman WHERE name LIKE 'N\_i%';



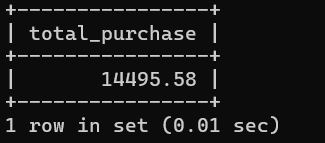
**10. Find that customer with all information who does not get any grade except NULL.**

select \* from customer WHERE grade is NULL;



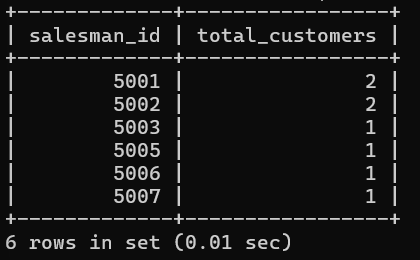
**11. Find the total purchase amount of all orders.**

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



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

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

****

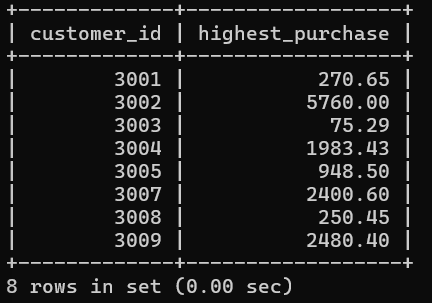
**13. Find the highest grade for each of the cities of the customers.**

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



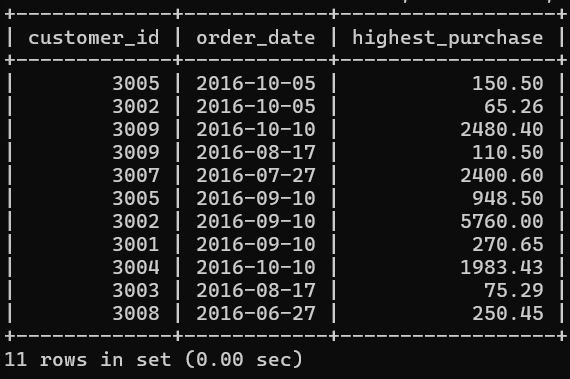
**14. Find the highest purchase amount ordered by each customer with their ID and highest purchase amount.**

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



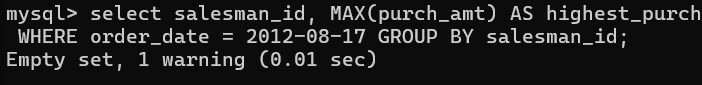
**15. Find the highest purchase amount ordered by each customer on a particular date with their ID, order date and highest purchase amount.**

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



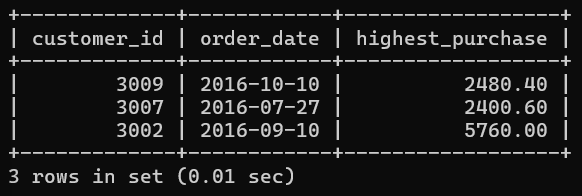
**16. Find the highest purchase amount on a date '2012-08-17' for each salesman with their ID.**

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



**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.**

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

****

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

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

