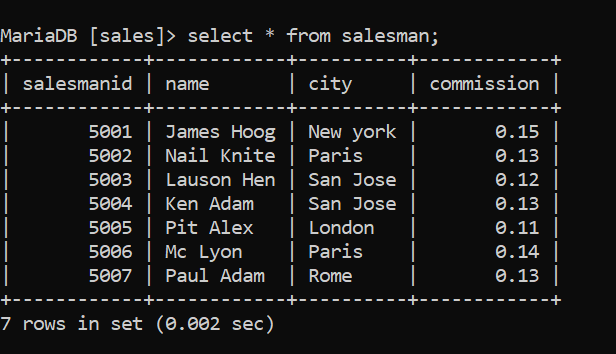
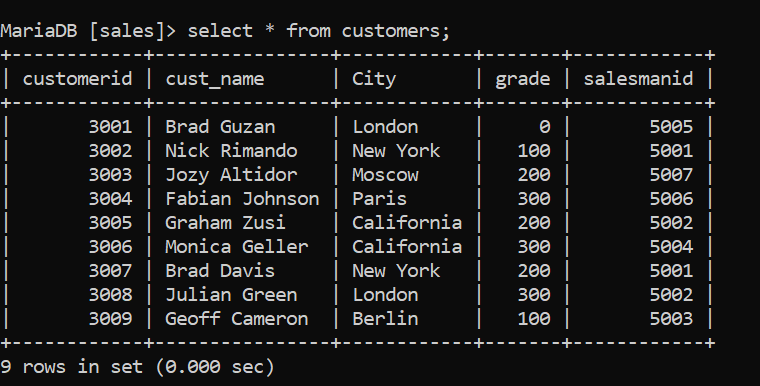
**SQL PROJECT**

**(Janhavi Bagul)**

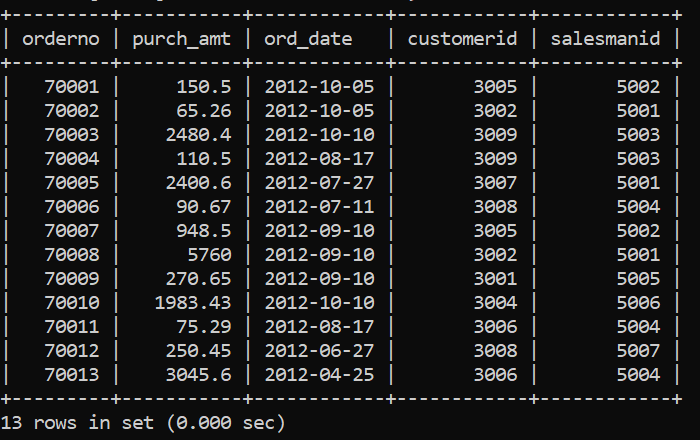
**Salesman Table**



**Customers Table**

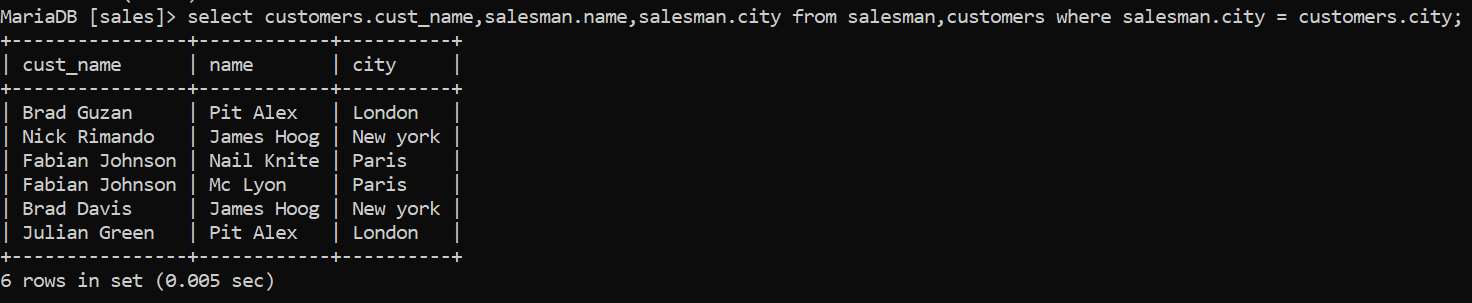


**Orders Table**



1. From the following tables, write a SQL query to find the salespeople and customers who live in the same city. Return customer name, salesperson name and salesperson city.

Ans- Select customers.cust\_name, salesman.name, salesman.city from salesman, customers where salesman.city = customers.city;



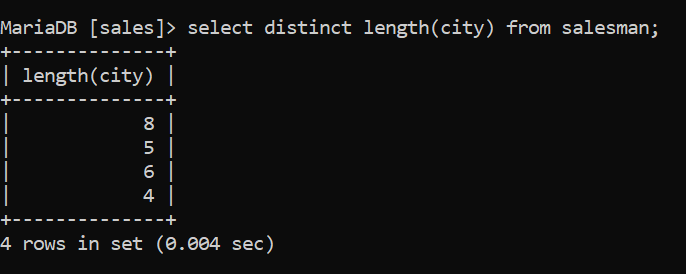
1. Write an SQL query to print all Customers details from the Customers table order by cust\_name Ascending.

Ans- Select \* from customers order by cust\_name asc;



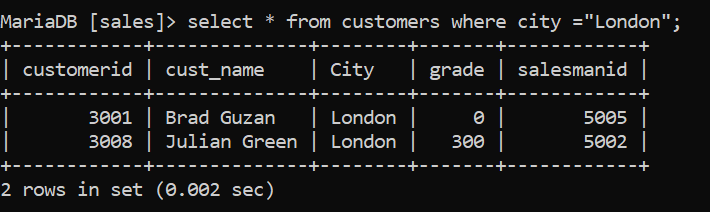
1. Write an SQL query that fetches the unique values of City from Salesman table and prints its length.

Ans- Select DISTINCT Length(city) from salesman;



1. Write an SQL query to print details of Customers with CITY name as “London”.

Ans- Select \* from Customers WHERE City =”London”;



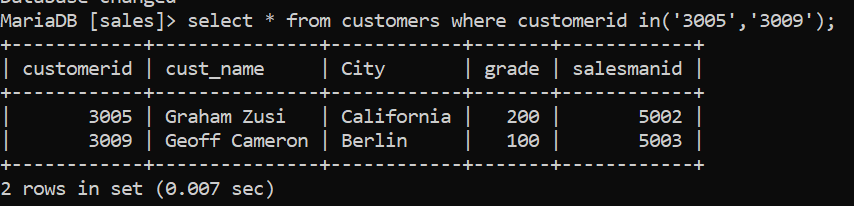
1. Write an SQL query to print details of the Salesman whose Name contains ‘a’.

Ans- Select \* from Salesman WHERE name like ’%a%’;



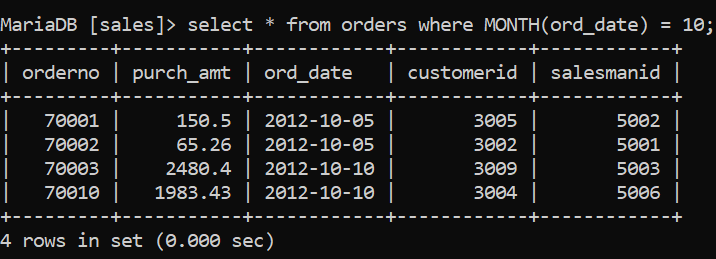
1. Write SQL query to print details of Customers with ID 3005 or 3009

Ans- Select \* from Customers WHERE customerid IN(‘3005,’3009’);



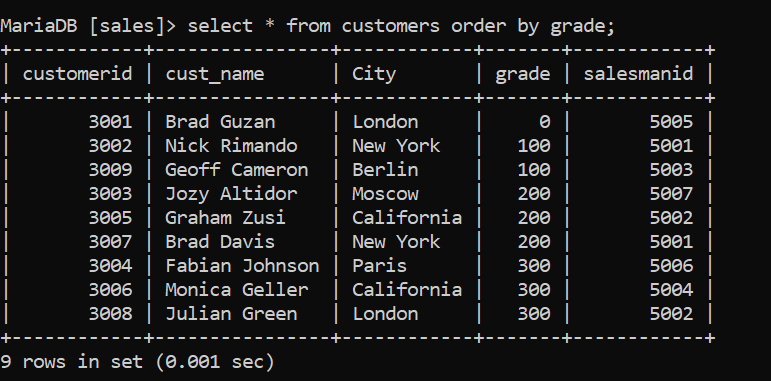
1. Write SQL query to print details of orders who ordered in the month of OCTOBER.

Ans- Select \* from orders WHERE MONTH(ord\_date) = 10;



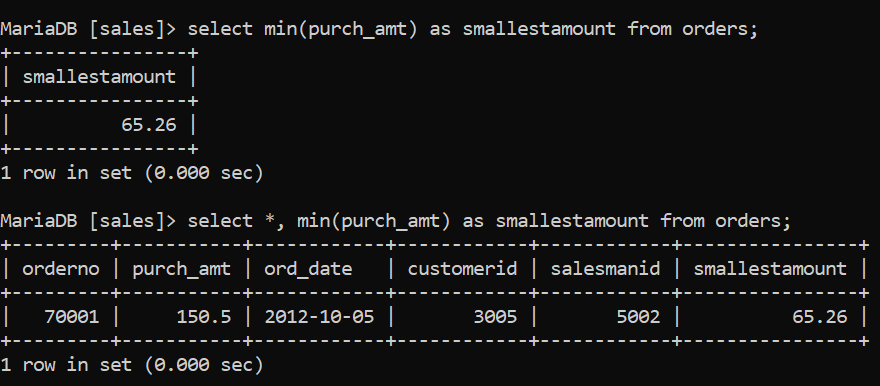
1. Write SQL query to print the Details of Customers in the descending order of the Grade.

Ans- Select \* from Customers ORDER BY grade;



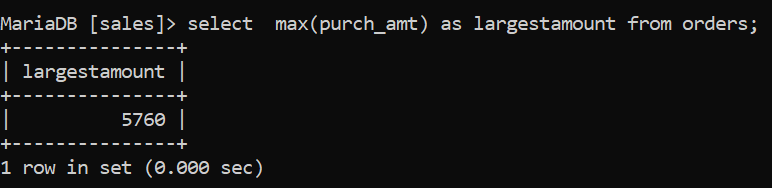
1. Write SQL query to find cheapest purchase amount of the orders.

Ans- Select MIN(purch\_amt) as smallestamount from orders;



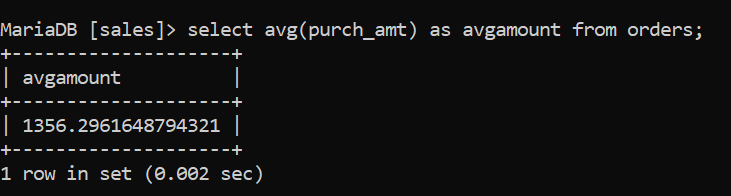
1. Write SQL query to print the largest purchase amount of the orders.

Ans – Select MAX(purch\_amt) as largestamount from orders;



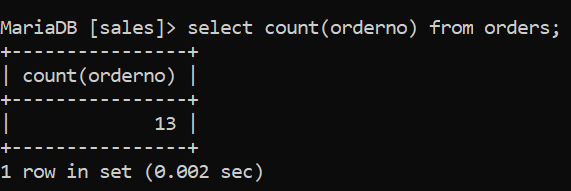
1. Write SQL query to print the average amount of the orders.

Ans- Select avg(purch\_amt) as avgamount from orders;



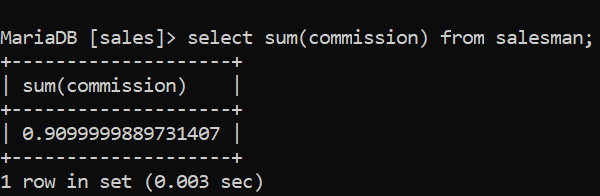
1. Write SQL query to find the number of orders

Ans Select COUNT(orderno) from orders;



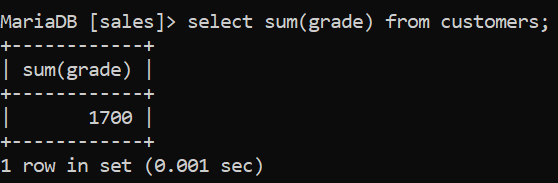
1. Write SQL query finds the sum of the "Commission" fields in the "Salesman" table.

Ans- Select sum(commission) from salesman;



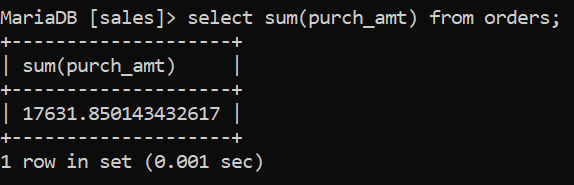
1. Write SQL query finds the sum of the "Grade" fields in the "Customers" table.

Ans – Select sum(grade) from customers;



1. Write SQL query finds the sum of the "Amount" fields in the "Orders" table.

Ans- Select sum(purch\_amt) from orders;



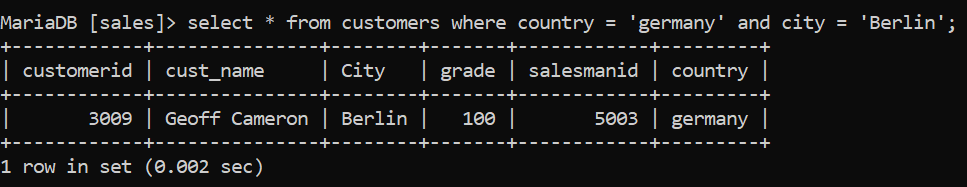
1. Alter Table add column Country from Customers Table and Update it.

Ans-



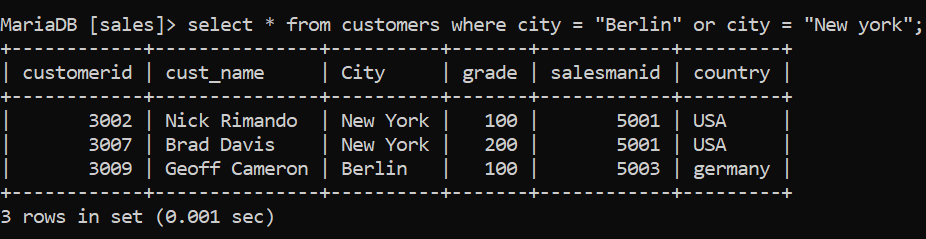
1. Write SQL query to selects all fields from "Customers" where country is "Germany" AND city is "Berlin".

Ans- Select \* from Customers where country = “Germany” AND city = “Berlin”;



1. Write SQL query to selects all fields from "Customers" where city is "Berlin" OR "New York"

Ans- Select \* from Customers WHERE City = “Berlin” or city = “new York”;



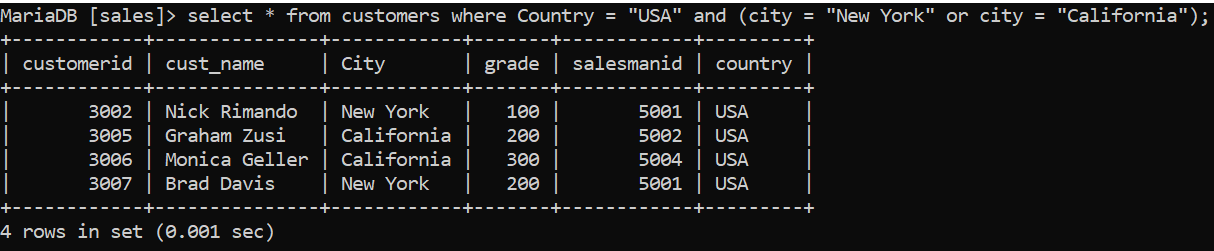
1. Write SQL query to selects all fields from "Customers" where country is NOT "USA"

Ans- Select \* from Customers WHERE NOT country = “USA”;



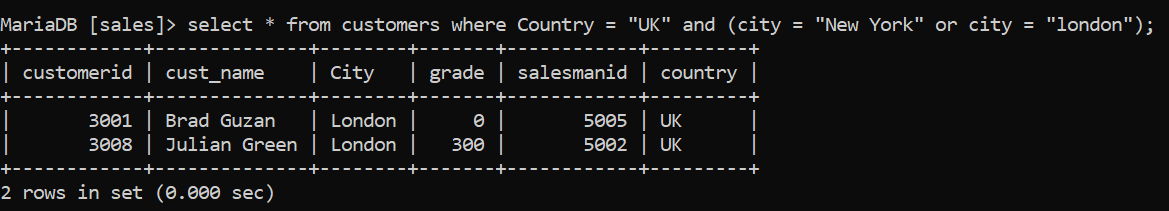
1. Write SQL query to selects all fields from "Customers" where country is "USA" AND city must be "New York " OR "California”

Ans- Select \* from Customers WHERE Country = “USA” AND (City = “New York” OR City = “California”);



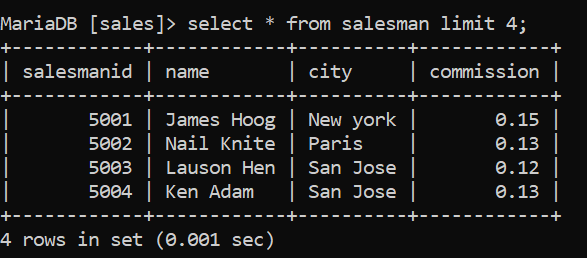
1. Write SQL query to selects all fields from "Customers" where country is "UK" AND city must be "New York " OR "London”

Ans- - Select \* from Customers WHERE Country = “UK” AND (City = “New York” OR City = “London”);



1. Write SQL Query to select all fields from Salesman with limit 4.

Ans Select \* from Salesman limit 4;



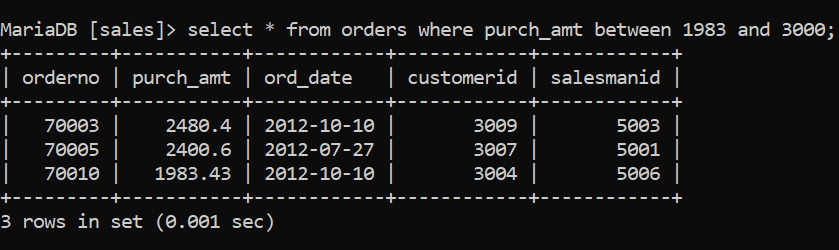
1. Write SQL query to selects all customers with a Salesman Name that does NOT start with "p".

Ans- Select \* from Salesman WHERE name NOT LIKE ‘p%’;



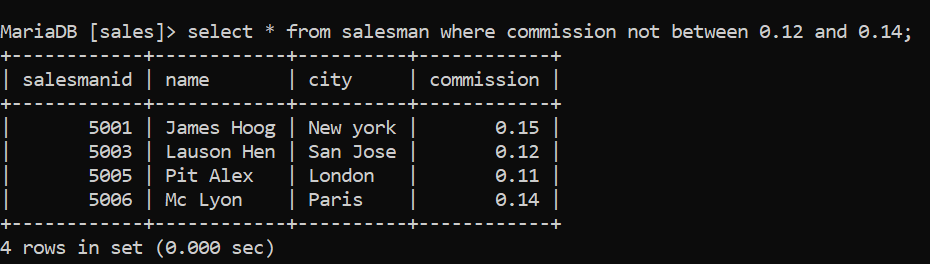
1. Write SQL query to selects all orders with Amount between 1983 and 3000

Ans- Select \* from Orders WHERE purch\_amt between 1983 AND 3000;



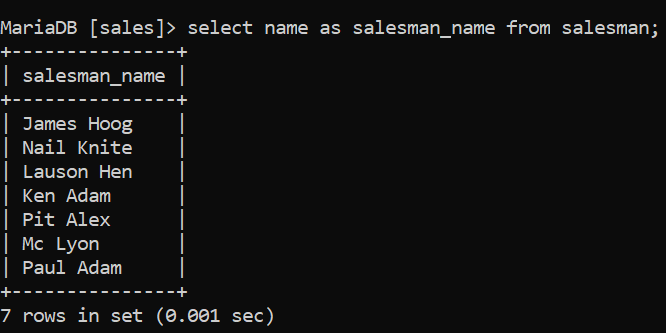
1. Write SQL query to selects all Salesman with Commission Not between 0.12 and 0.14

Ans- Select \* from Salesman WHERE commission NOT BETWEEN 0.12 and 0.14



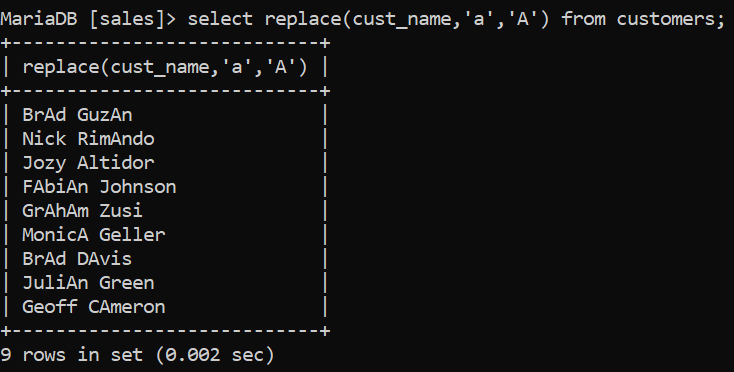
1. Write an SQL query to fetch “NAME” from Salesman table using the alias name as “SALESMAN\_NAME”

Ans- Select name as salesman\_name from salesman;



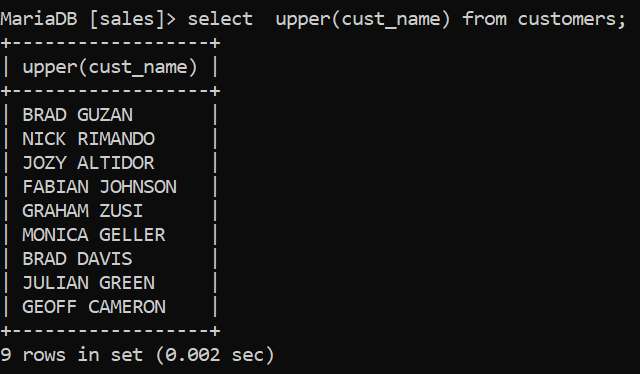
1. Write an SQL query to print the CUST\_NAME from Customers table after replacing ‘a’ with ‘A’.

Ans- Select REPLACE(cust\_name, ’a’, ’A’) from customers;



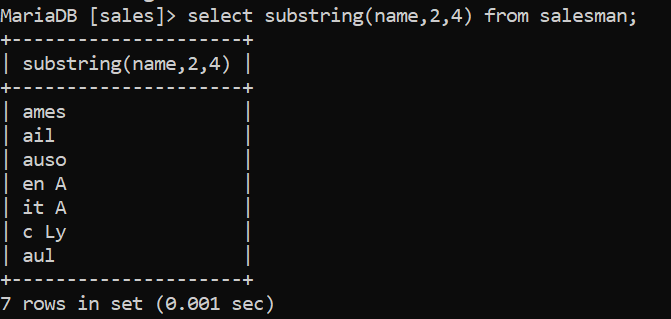
1. Write an SQL query to fetch “CUST\_NAME” from Customers table in upper case.

Ans- Select upper(cust\_name) from customers;



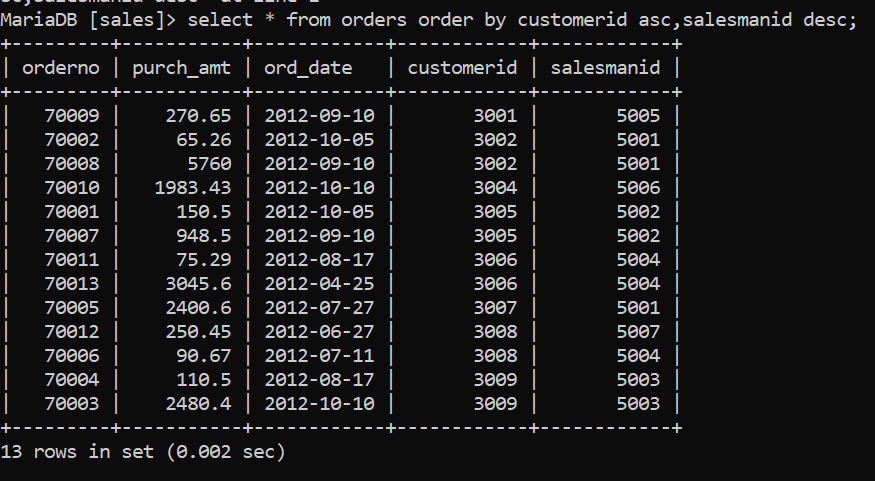
1. Write an SQL query to print the second four characters of name from salesman table.

Ans-Select substring(name,2,4) from salesman



1. Write an SQL query to print all orders details from the Orders table order by customerid Ascending and Salesmanid Descending.

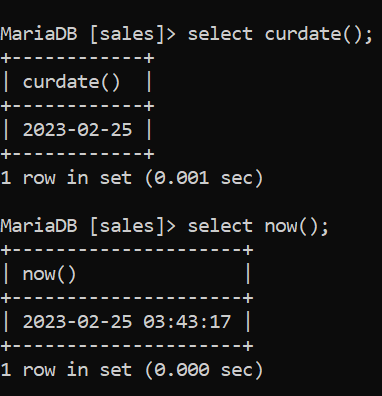
Ans- Select \* from orders ORDER BY customerid asc, salesmanid desc;



1. Write an SQL query to show the current date and time.

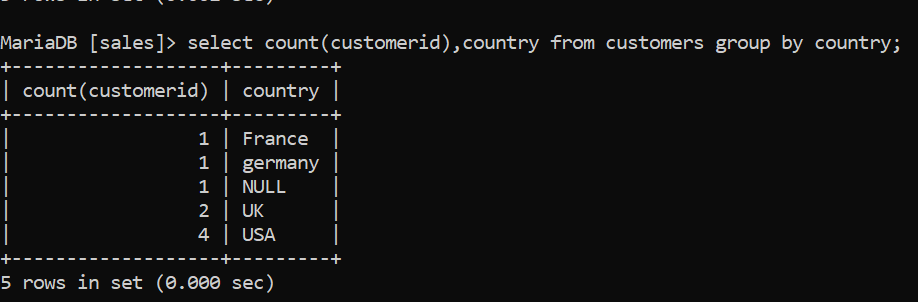
Ans- Select curdate();

Select now();



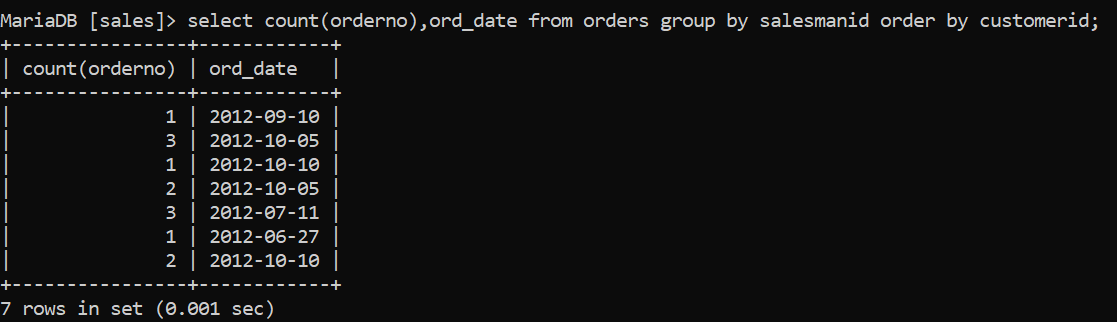
1. Write SQL query to lists the number of customers in each country.

Ans- Select count(customerid), country from customers group by country;



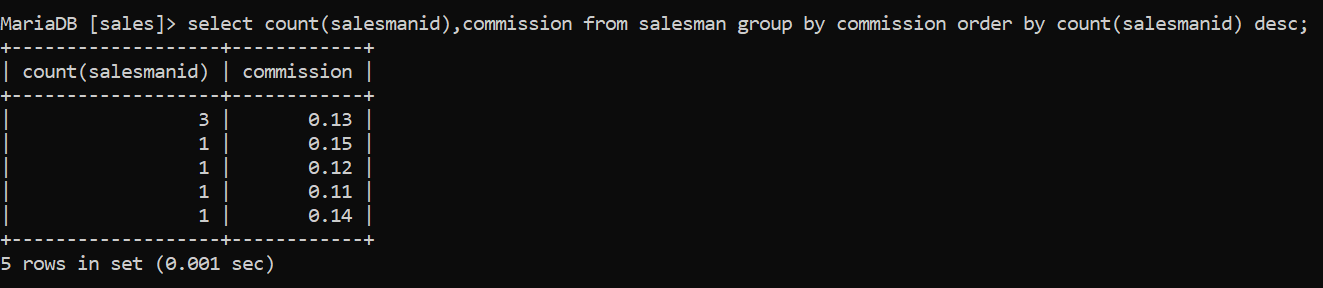
1. Write SQL query to list no. of orders order date in each salesmanid no. with customer list.

Ans- Select count(orderno), ord\_date from orders group by salesmanid order by customerid;



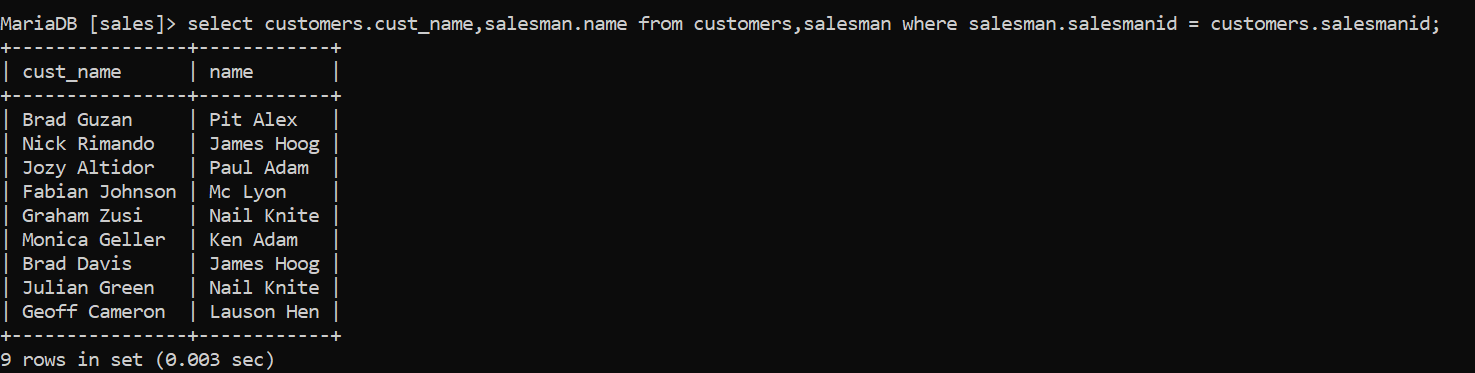
1. Write SQL query lists the number of salesman in each commission, sorted high to low

Ans- Select count(salesmanid), commission from salesman group by commission order by count(salesmanid) desc;



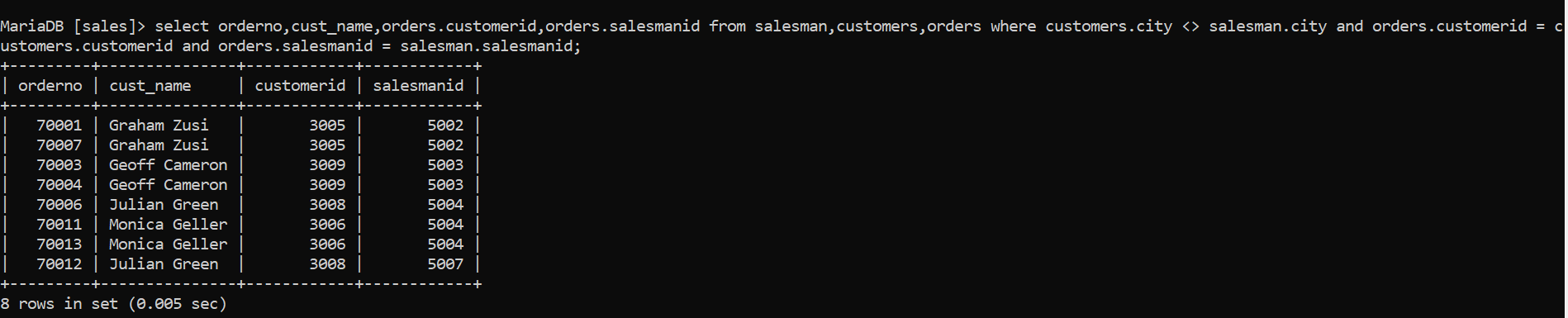
1. write a SQL query to locate all the customers and the salesperson who works for them. Return customer name, and salesperson name.

Ans- Select customers.cust\_name, salesman.name from customers, salesman.salesmanid = customers.salesmanid;



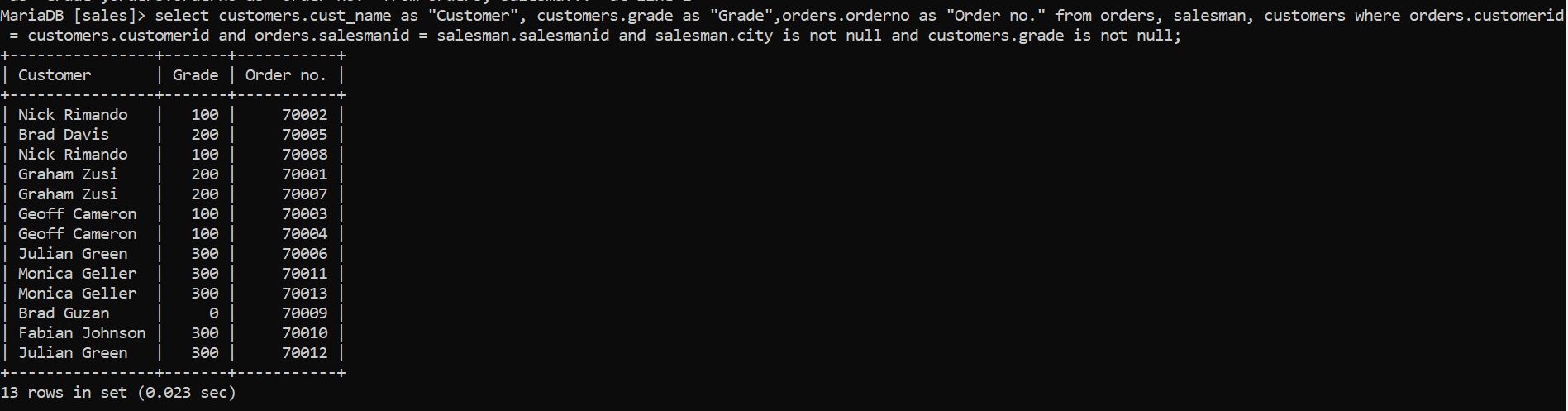
1. write a SQL query to find those salespeople who generated orders for their customers but are not located in the same city. Return orderno, cust\_name, customerid (orders table), salesmanid (orders table).

Ans- Select orderno, cust\_name, orders.customerid, orders.salesmanid from salesman, customers, orders WHERE customers.city <> salesman.city AND orders.customerid = customers.customerid AND orders.salesman = saleman.salemanid;



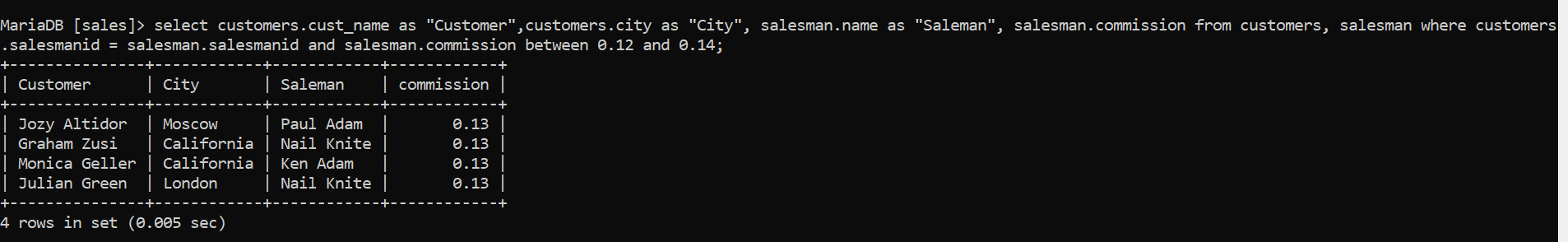
1. Write a SQL query to find those customers where each customer has a grade and is served by a salesperson who belongs to a city. Return cust\_name as "Customer", grade as "Grade".

Ans- Select customers.cust\_name AS "Customer", customers.grade AS "Grade",orders.orderno AS "Order no." from orders, salesman, customers WHERE orders.customerid = customers.customerid and orders.salesmanid = salesman.salesmanid and salesman.city is NOT NULL and customers.grade is NOT NULL;



1. write a SQL query to find those customers who are served by a salesperson and the salesperson earns commission in the range of 12% to 14% (Begin and end values are included.). Return cust\_name AS "Customer", city AS "City"

Ans- Select customers.cust\_name AS "Customer",customers.city AS "City", salesman.name AS "Saleman", salesman.commission from customers, salesman WHERE customers.salesmanid = salesman.salesmanid AND salesman.commission between 0.12 and 0.14;



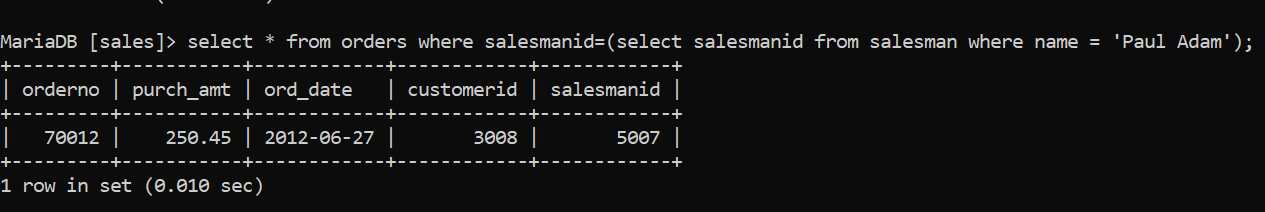
1. Write a SQL query to find all orders executed by the salesperson and ordered by the customer whose grade is greater than or equal to 200. Compute purch\_amt\*commission as “Commission”. Return customer name, commission as “Commission%” and Commission.

Ans- Select orderno, cust\_name, commission AS "Commission%",purch\_amt\*commission AS "commission" from salesman,orders,customers WHERE orders.customerid = customers.customerid AND orders.salesmanid = salesman.salesmanid AND customers.grade>=200;



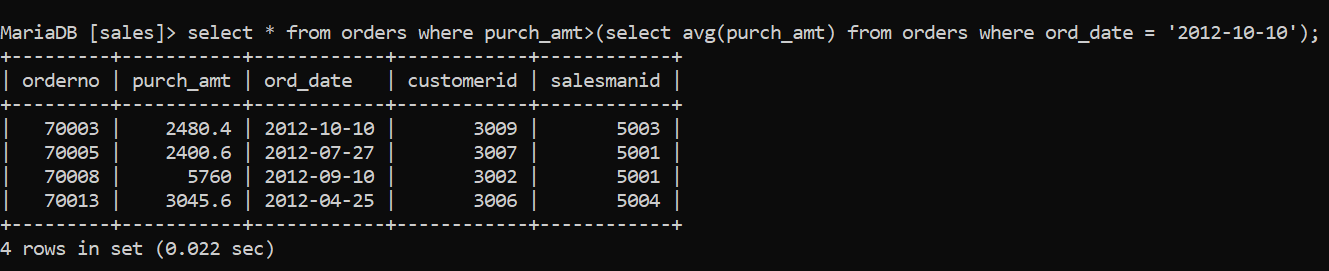
1. Write SQL query to display all the orders issued by the salesman 'Paul Adam' from the orders table.

Ans- Select \* from orders WHERE salesmanid=(Select salesmanid from salesman WHERE name = 'Paul Adam');



1. Write SQL query to display all the orders which values are greater than the average order value for 10th October 2012.

Ans- Select \* from orders WHERE purch\_amt>(select AVG(purch\_amt) from orders WHERE ord\_date = '2012-10-10');



1. Write SQL query to Find all orders attributed to salesmen in Paris.

Ans- Select \* from orders WHERE salesmanid IN(select salesmanid from salesman WHERE City = "Paris");

