**DATABASE ASSIGNMENT**

**SQL QUERIES**

1. ) Student table

CREATE TABLE student(

Rollno int PRIMARY KEY,

Name varchar(50),

Branch varchar(100)

);

CREATE TABLE exam(

S\_code varchar(50),

Marks int(11),

P\_code varchar(10),

FOREIGN KEY (Rollno) REFERENCES student(Rollno)

);

Student table insertion

INSERT INTO student(Rollno, Name, Branch)

VALUES (1,"Jay", "Computer Science");

INSERT INTO student(Rollno, Name, Branch)

VALUES (2,"Suhani", "Electronic and Communications");

INSERT INTO student(Rollno, Name, Branch)

VALUES (3,"Kriti", "Electronic and Communications");

Exam table insertion

INSERT into exam(Rollno, S\_code, Marks, P\_code)

VALUES(1, "CS11", 50, "CS");

INSERT into exam(Rollno, S\_code, Marks, P\_code)

VALUES(1, "CS12", 60, "CS");

INSERT into exam(Rollno, S\_code, Marks, P\_code)

VALUES(2, "EC101", 66, "EC");

INSERT into exam(Rollno, S\_code, Marks, P\_code)

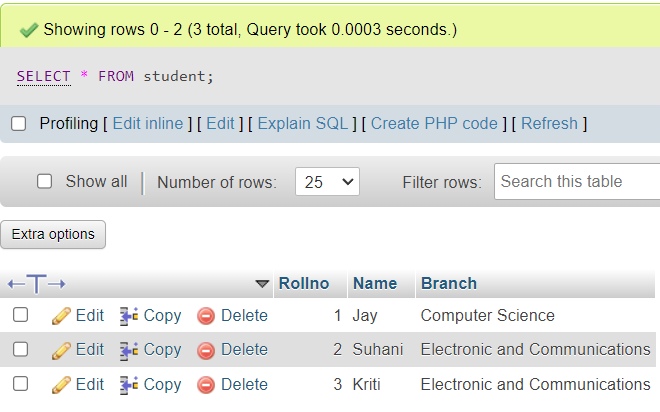
VALUES(2, "EC102", 70, "EC");

INSERT into exam(Rollno, S\_code, Marks, P\_code)

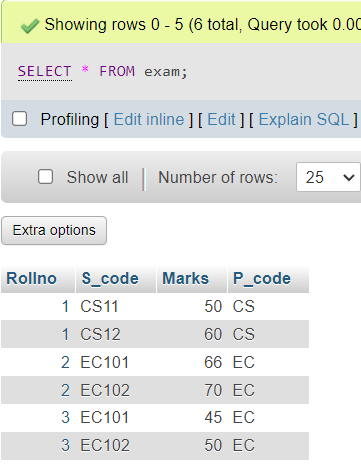
VALUES(3, "EC101", 45, "EC");

INSERT into exam(Rollno, S\_code, Marks, P\_code)

VALUES(3, "EC102", 50, "EC");

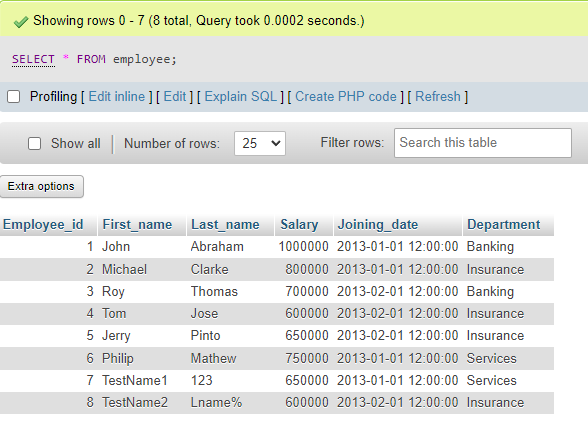


Exam table

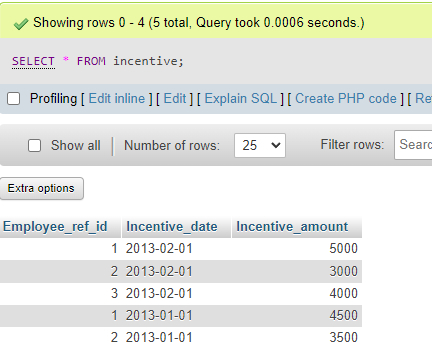


2.)

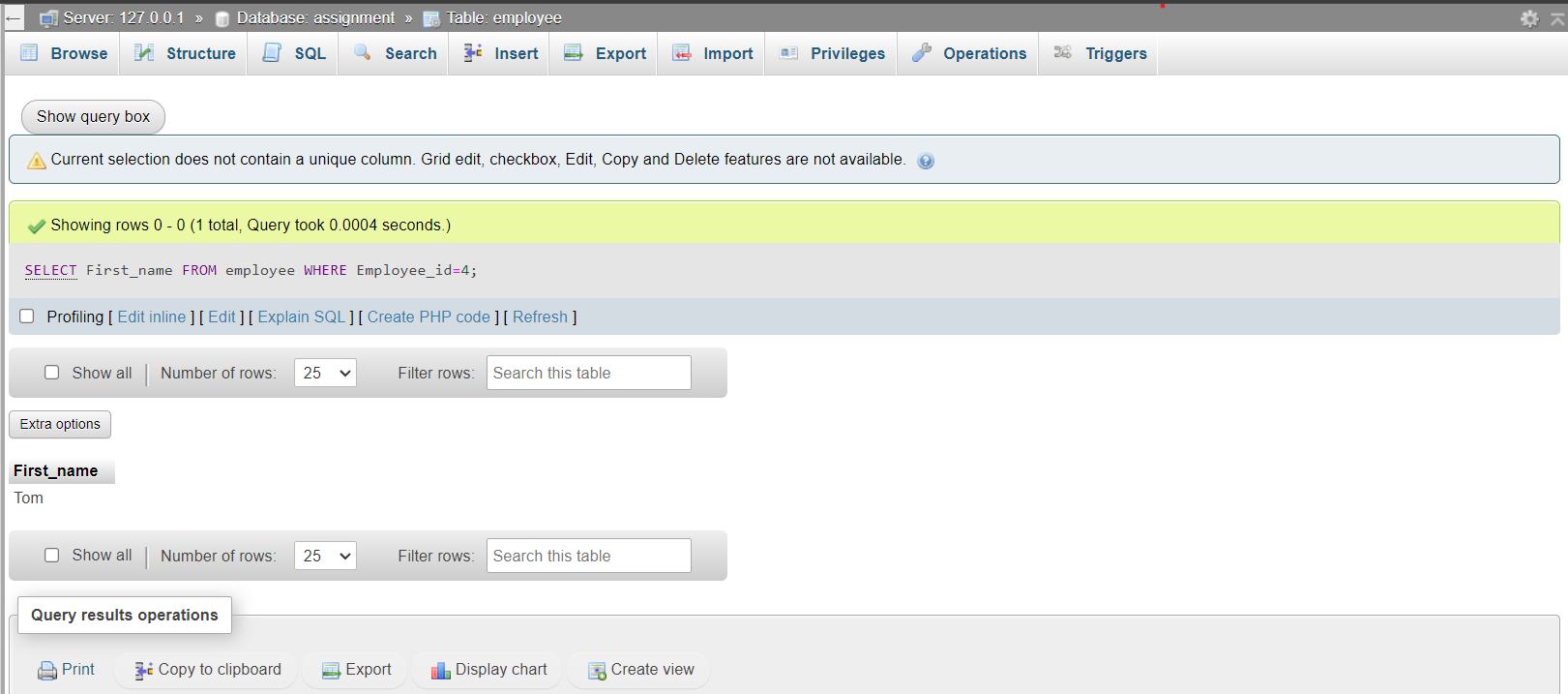
Employee Table



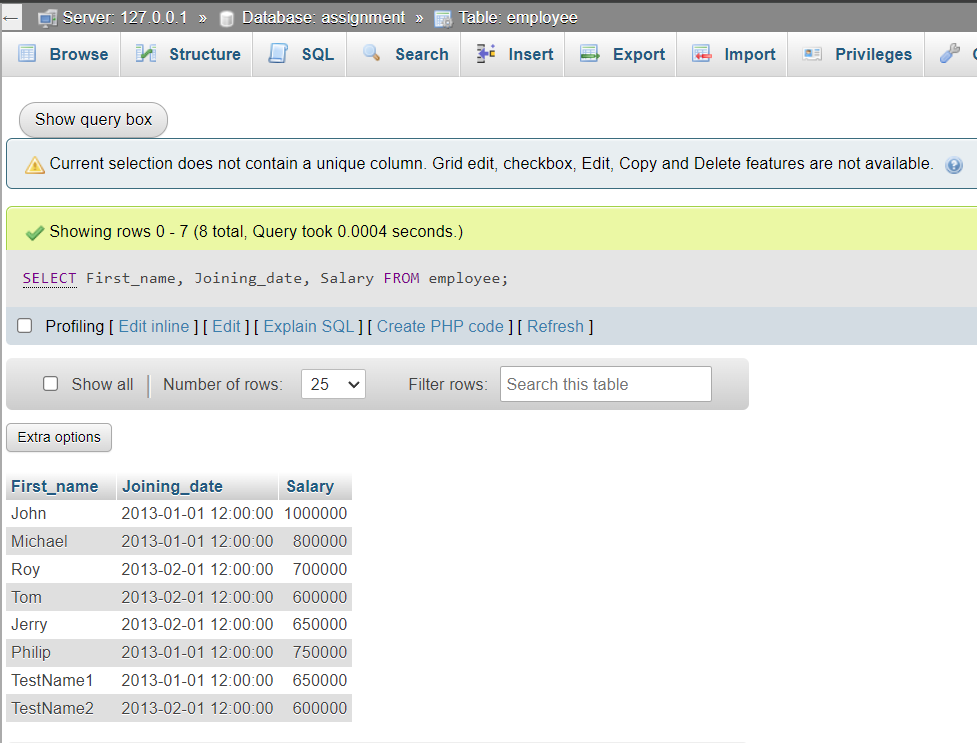
Incentive Table



1. Get First\_Name from employee table using Tom name “Employee Name”.



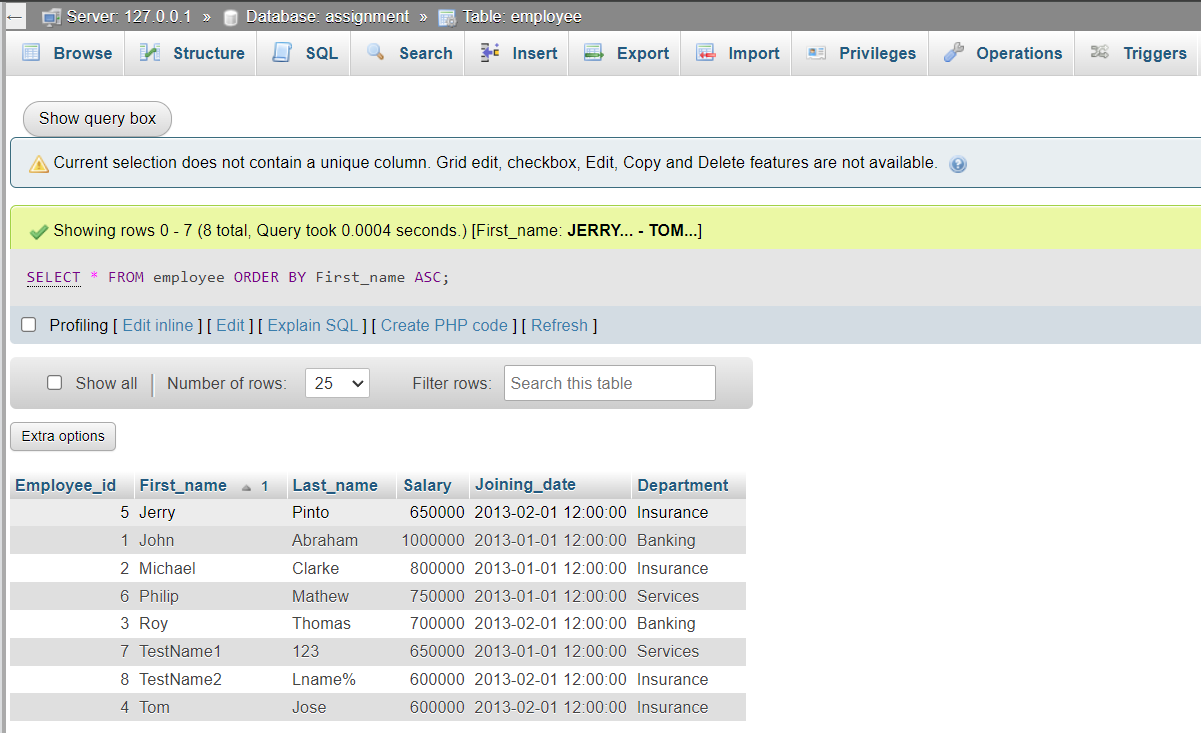
4.Get FIRST\_NAME, Joining Date, and Salary from employee table.



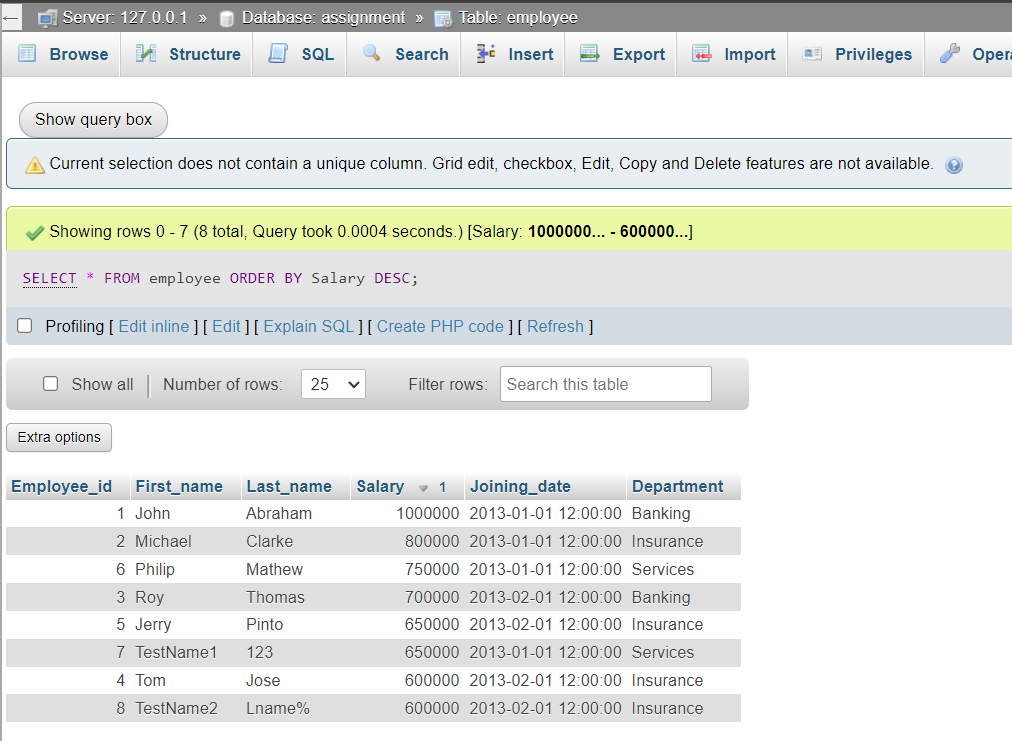
5.Get all employee details from the employee table order by First\_Name

Ascending and Salary descending?

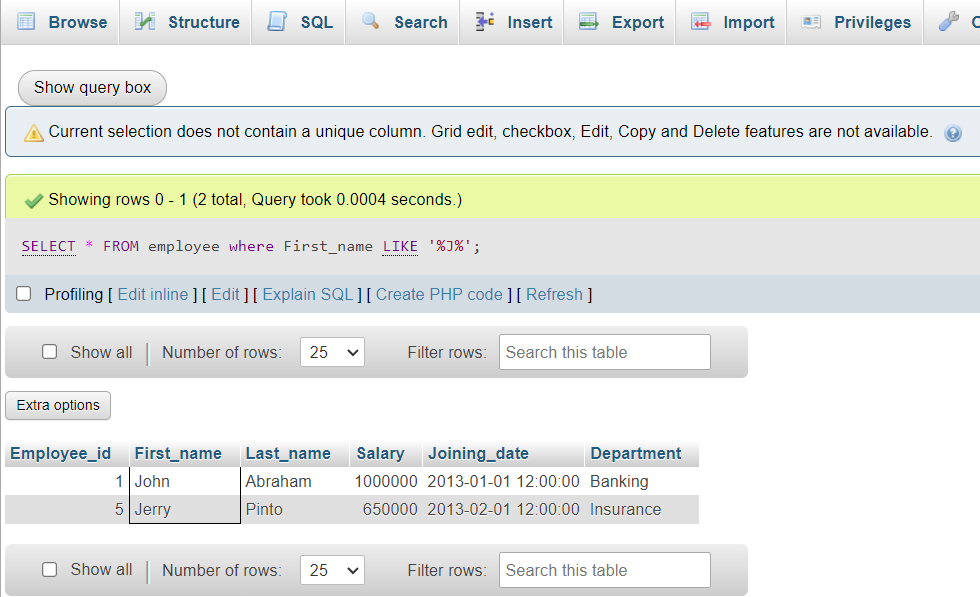
First Name Asc



Salary Desc

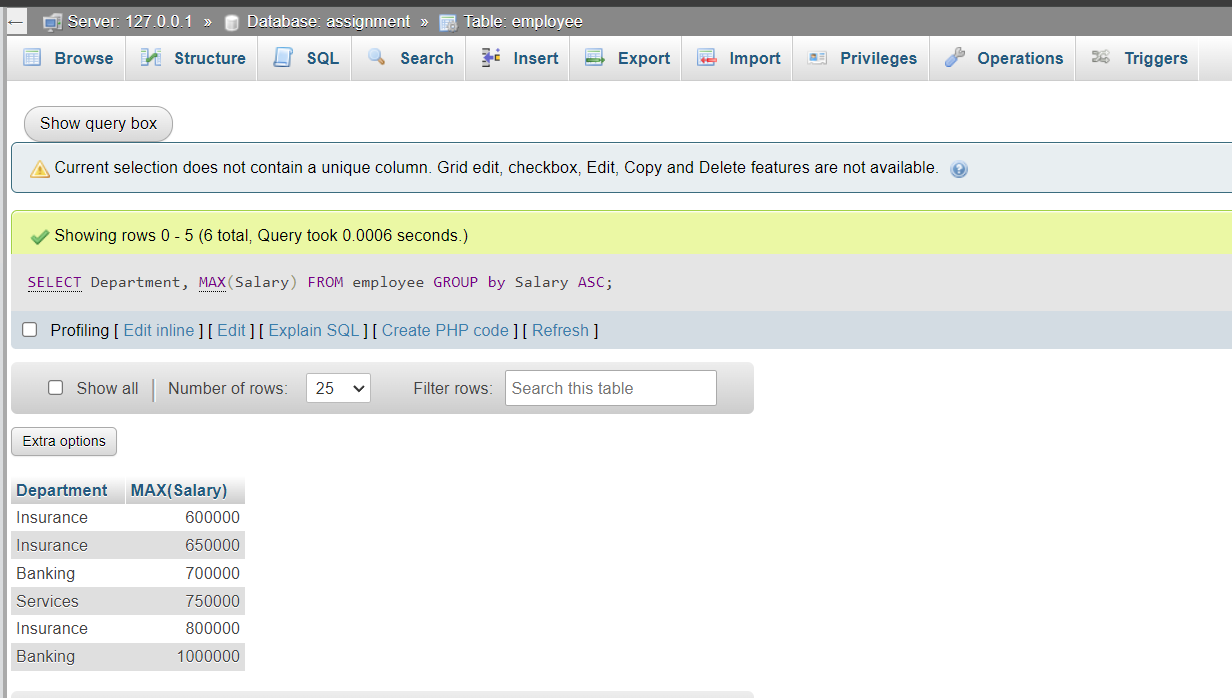


1. Get employee details from employee table whose first name contains ‘J’.

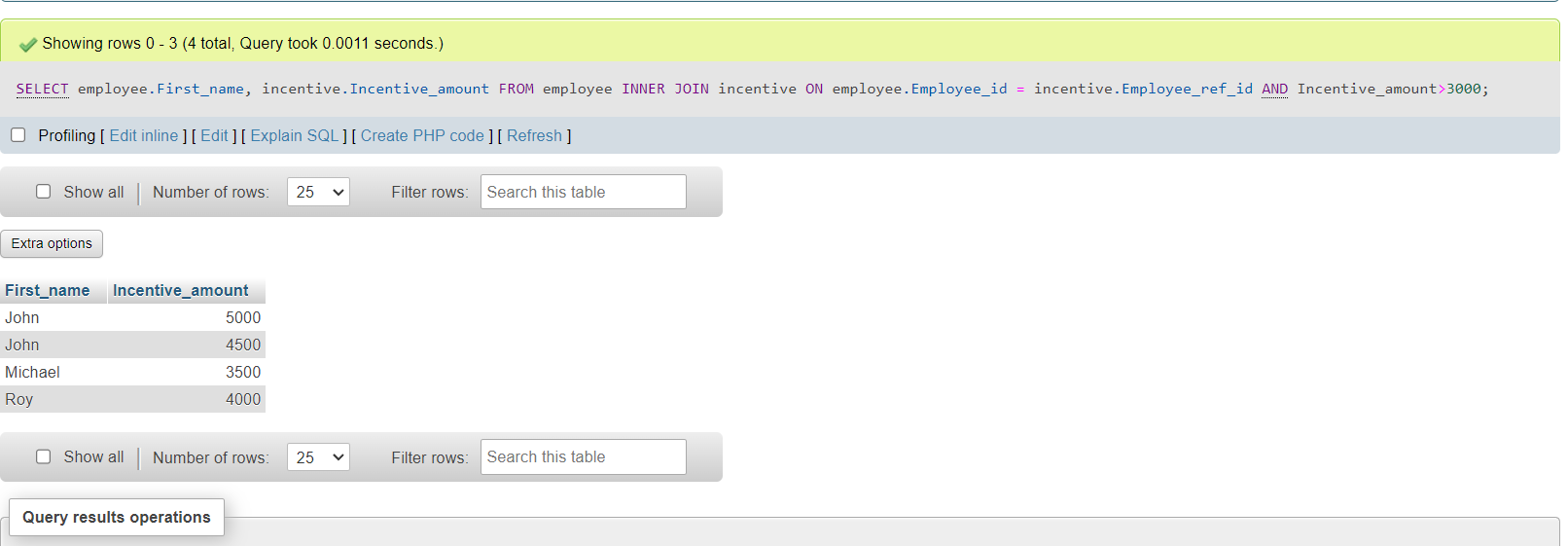


7.Get department wise maximum salary from employee table order by

Salary ascending?



1. Select first\_name, incentive amount from employee and incentives table forthose employees who have incentives and incentive amount greater than 3000



1. Create After Insert trigger on Employee table which insert records in viewtable

DELIMITER

CREATE TRIGGER trg\_InsertEmployee

AFTER INSERT

ON employee

FOR EACH ROW

BEGIN

INSERT INTO reminders

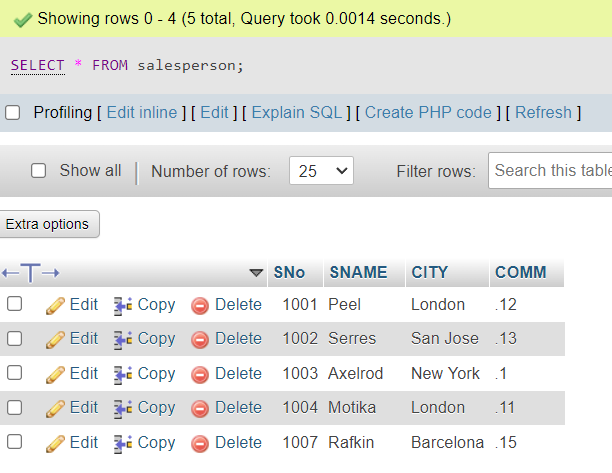
VALUES (NEW.employee\_id, NEW.First\_Name, NEW.Department);

END;

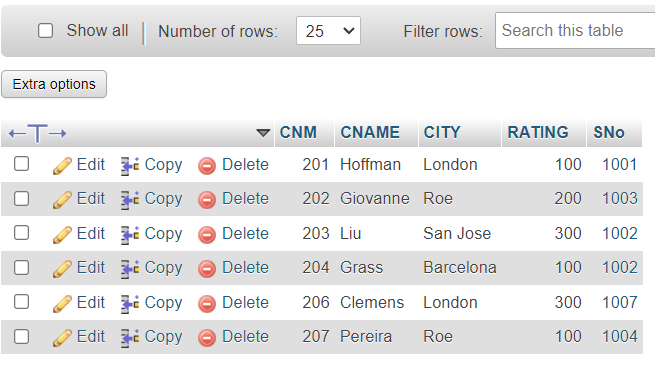
DELIMITER ;

11.)

**Salesperson Table**

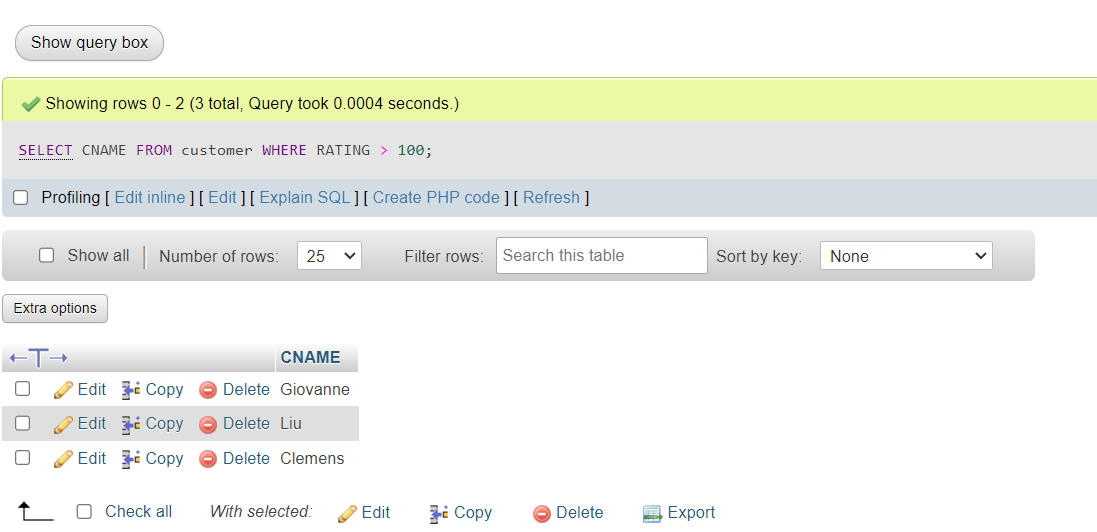


**Customer Table**



12.Retrieve the below data from above data:-

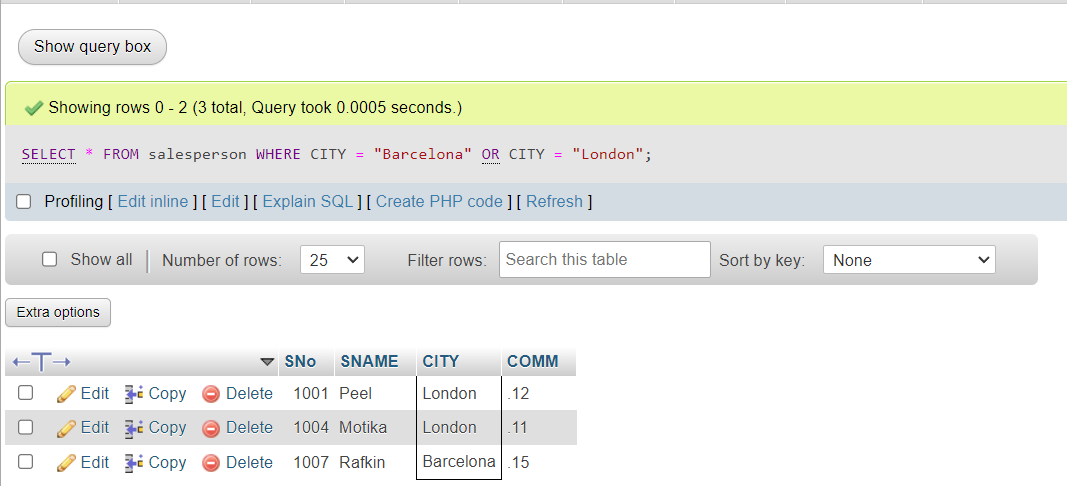
1. All Customer name whose rating is more than 100.



1. Names and cities of all salespeople in London with commission above 0.12

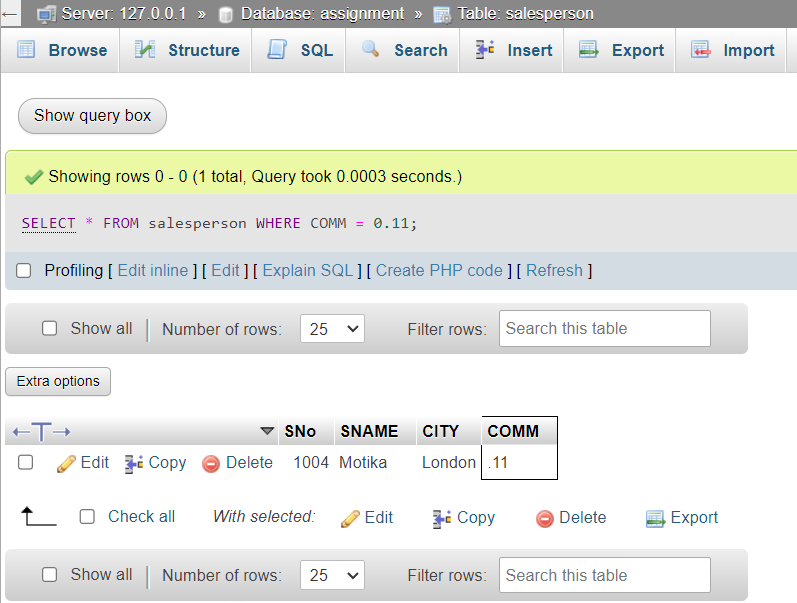


1. All salespeople either in Barcelona or in London

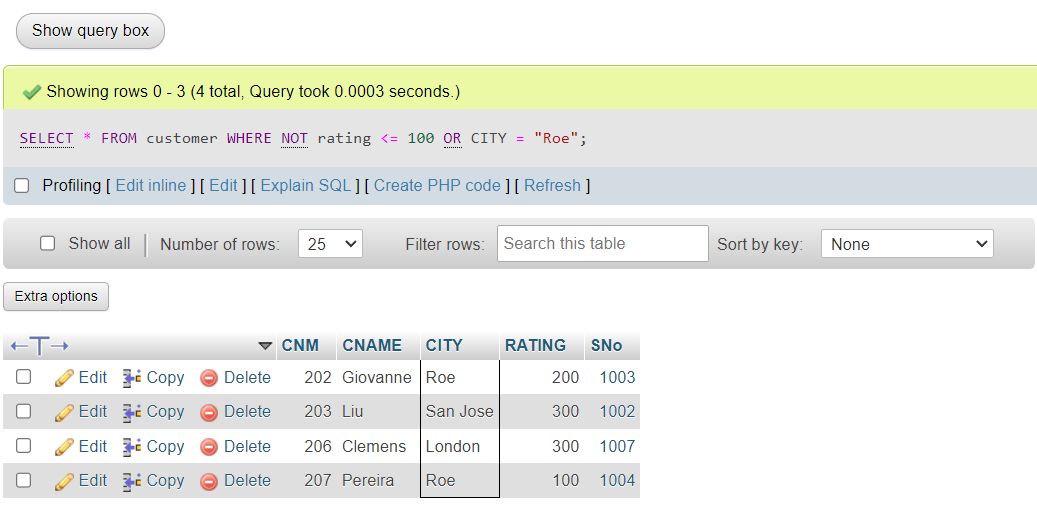


1. All salespeople with commission between 0.10 and 0.12. (Boundary

Values should be excluded).



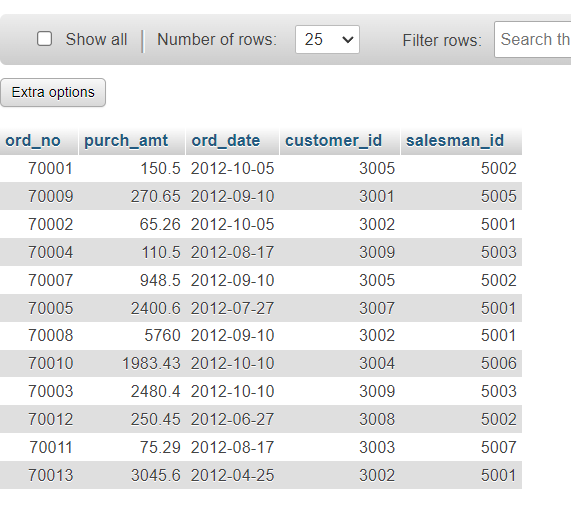
17.All customers excluding those with rating <= 100 unless they are located in Rome



18 )Write a SQL statement that displays all the information about all salespeople



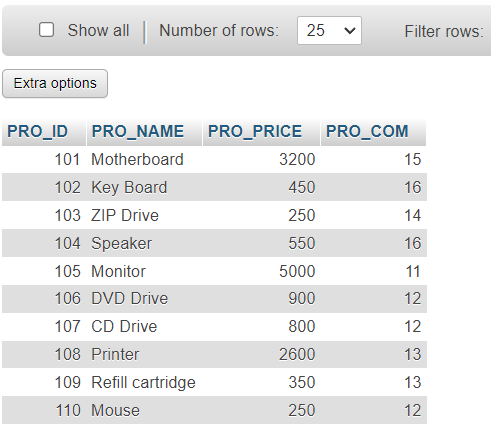
**Orders Table**



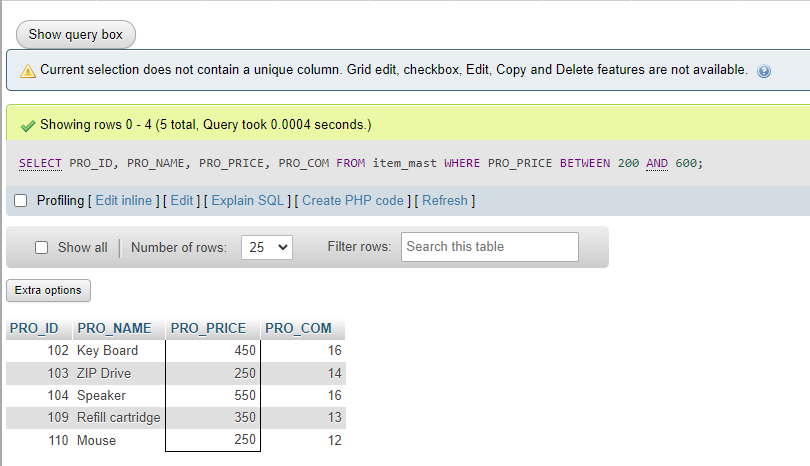
1. From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord\_no, ord\_date, purch\_amt.



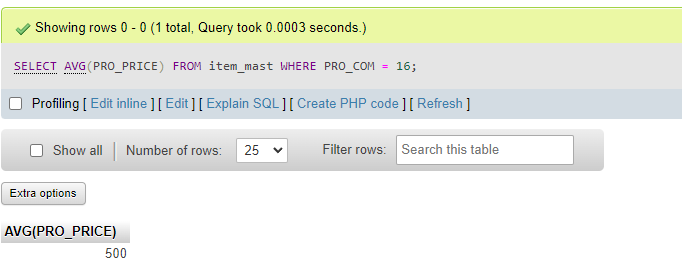
**Item\_Mast Table**



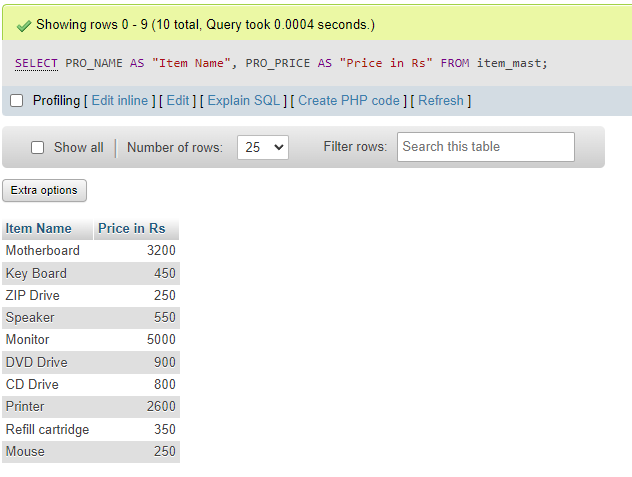
1. From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro\_id, pro\_name, pro\_price, and pro\_com



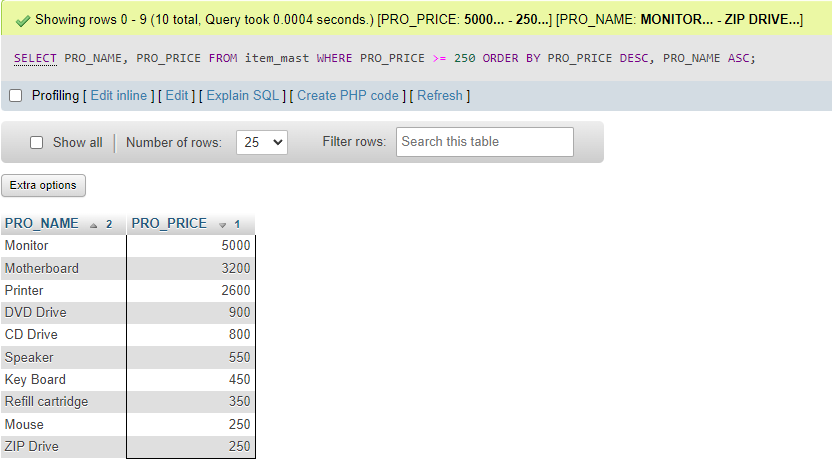
1. From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg.



1. From the following table, write a SQL query to display the pro\_name as 'Item Name' and pro\_price as 'Price in Rs.'



23.From the following table, write a SQL query to find the items whose prices are higher than or equal to $250. Order the result by product price in descending, then product name in ascending. Return pro\_name and pro\_price.



24.From the following table, write a SQL query to calculate average price of the items for each company. Return average price and company code.

