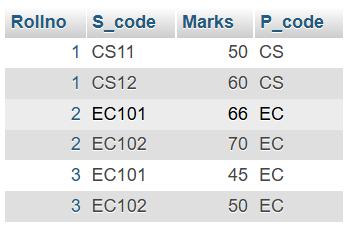
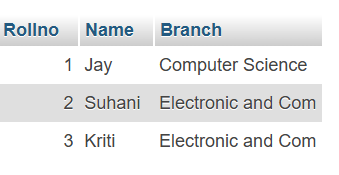
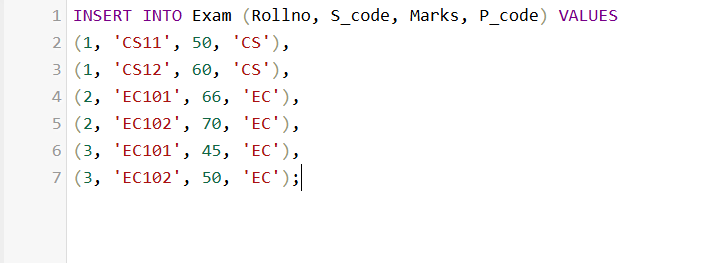
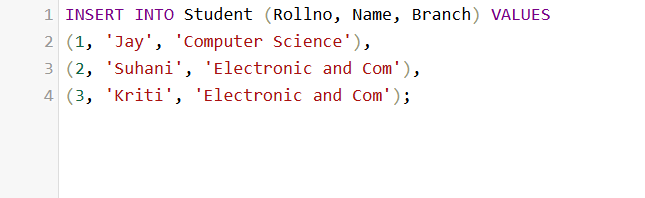
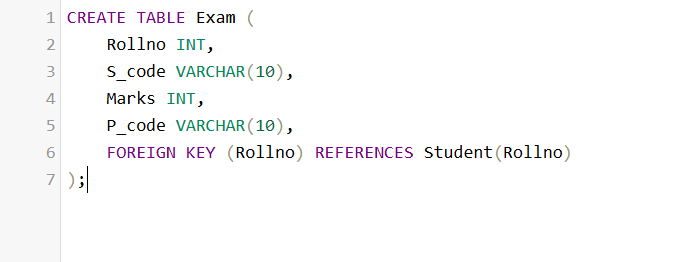
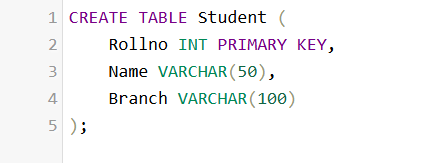
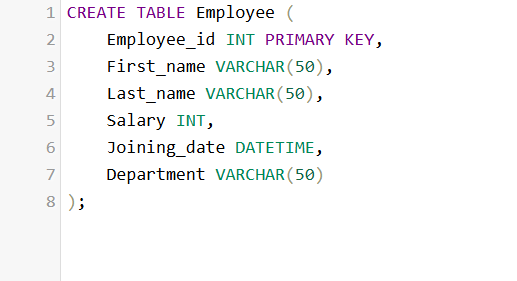
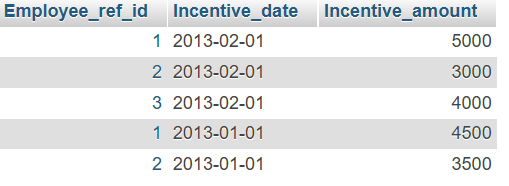
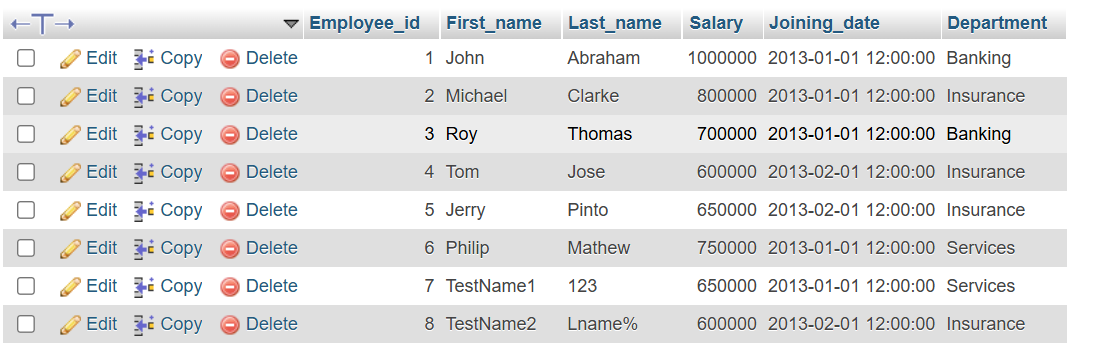
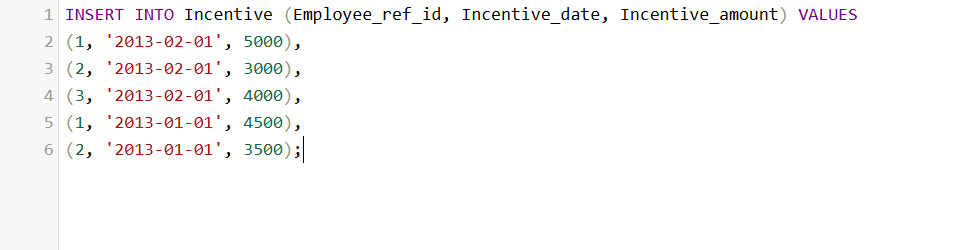
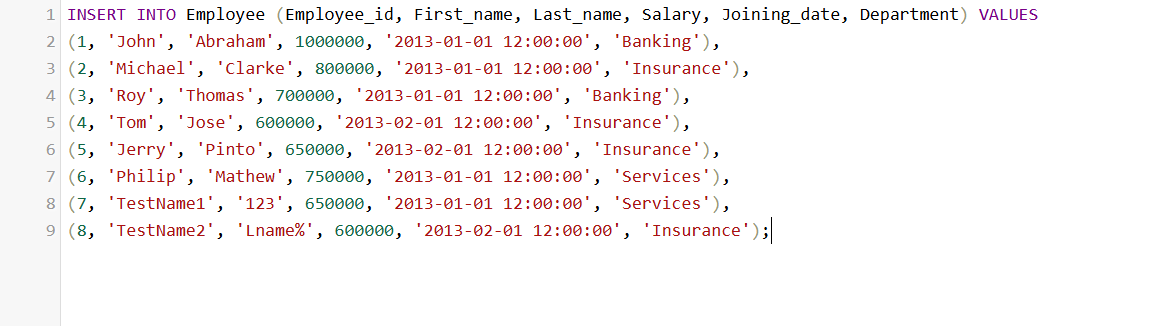
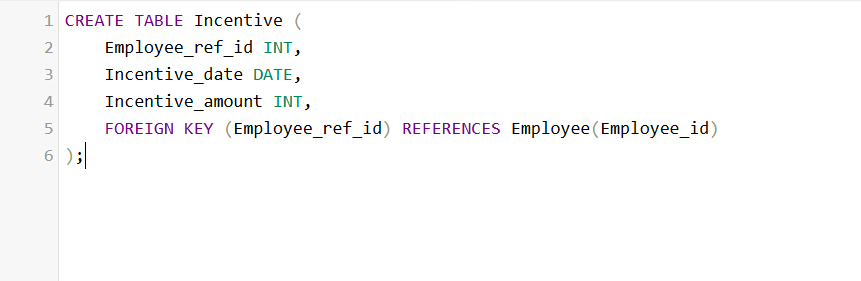
**SQL Queries**

1. Create Table Name : Student and Exam

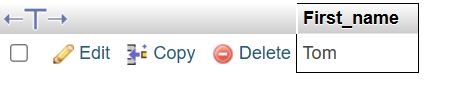
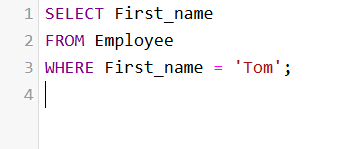


1. Create table given below: Employee and Incentive Table

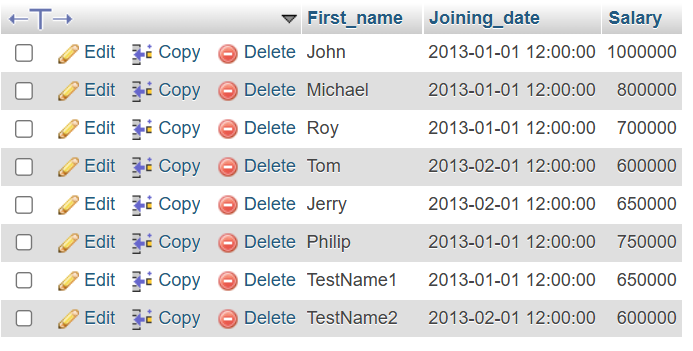
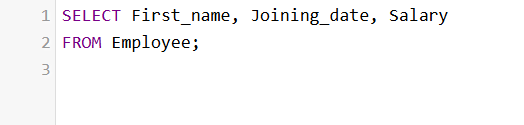




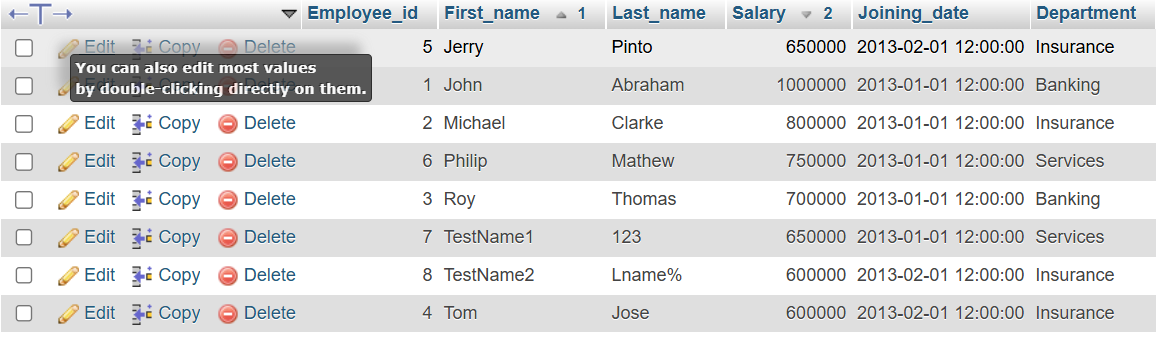
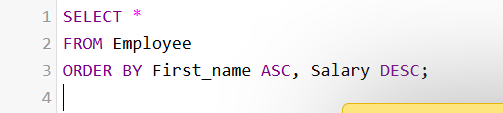
1. Get First\_Name from employee table using Tom name “Employee Name”.



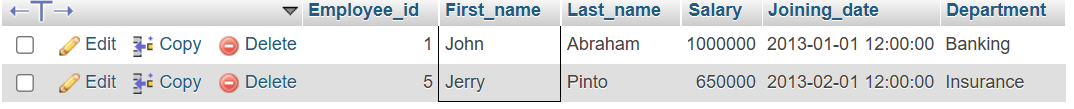
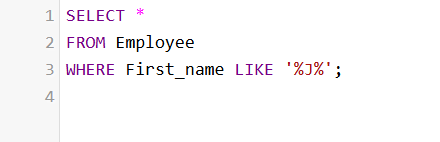
1. Get FIRST\_NAME, Joining Date, and Salary from employee table.



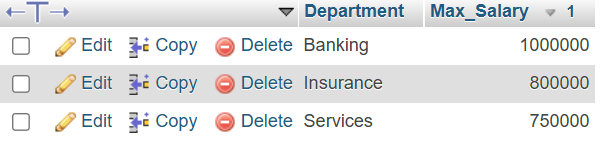
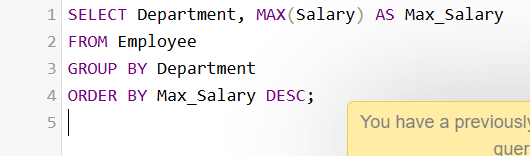
1. Get all employee details from the employee table order by First\_Name ascending and Salary descending?



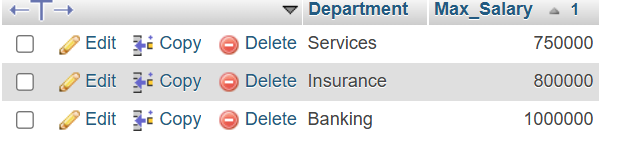
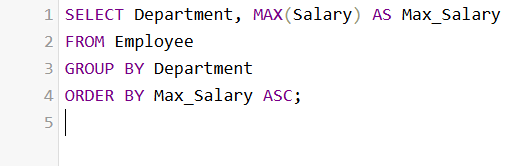
1. Get employee details from employee table whose first name contains ‘J’.



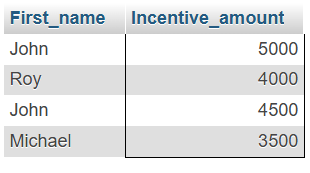
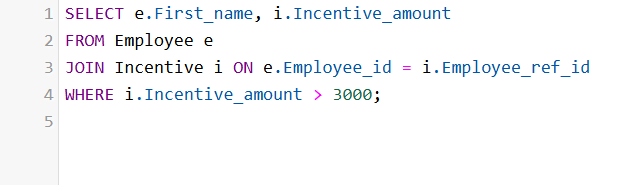
1. Get department wise maximum salary from employee table order by.



1. salary ascending?



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



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

CREATE TABLE ViewTable (

Employee\_id INT,

First\_name VARCHAR(50),

Last\_name VARCHAR(50),

Salary INT,

Joining\_date DATETIME,

Department VARCHAR(50)

);

CREATE TRIGGER after\_employee\_insert

AFTER INSERT ON Employee

FOR EACH ROW

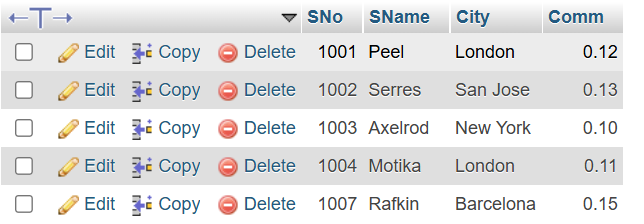
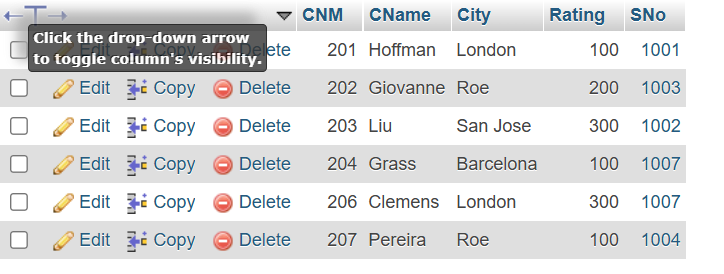
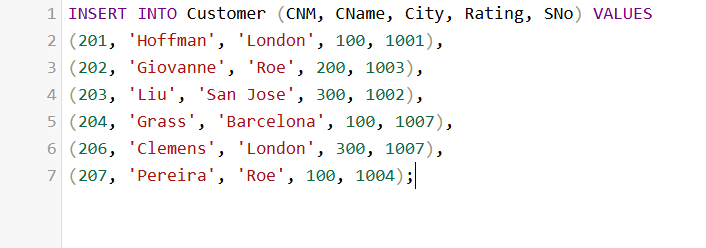
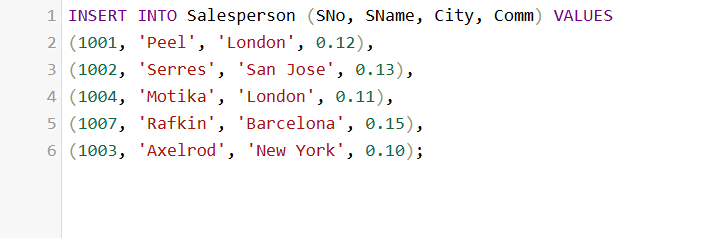
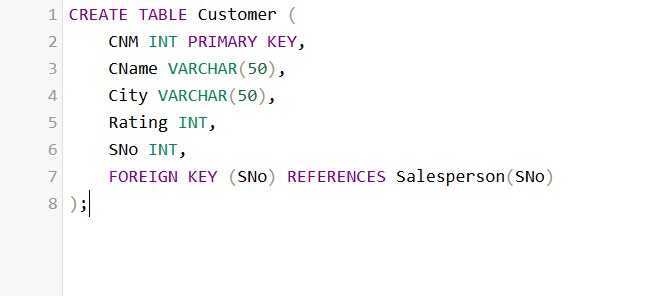
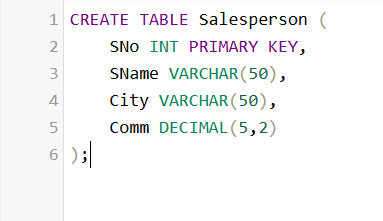
BEGIN

INSERT INTO ViewTable (Employee\_id, First\_name, Last\_name, Salary, Joining\_date, Department)

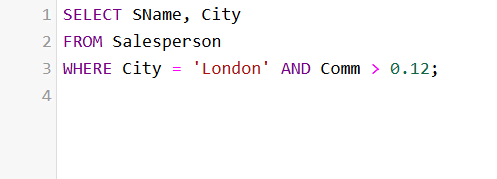
VALUES (NEW.Employee\_id, NEW.First\_name, NEW.Last\_name, NEW.Salary, NEW.Joining\_date, NEW.Department);

END;

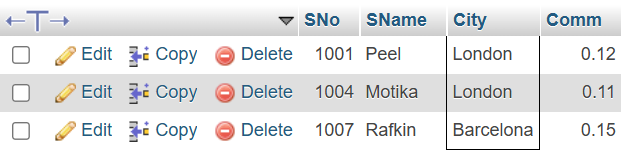
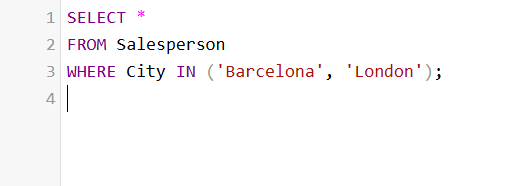
11.Create table given below: Salesperson and Customer



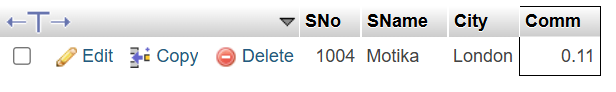
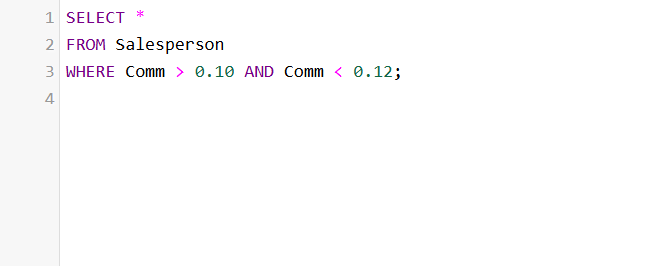
14.Names and cities of all salespeople in London with commission above 0.12



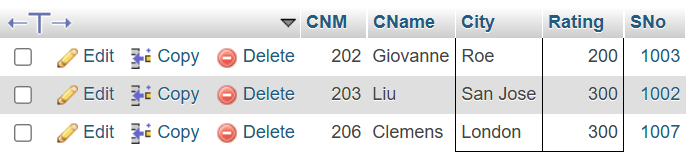
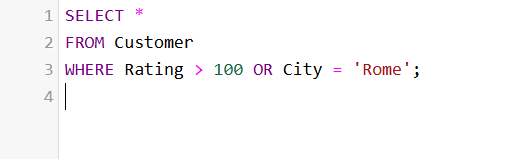
15.All salespeople either in Barcelona or in London



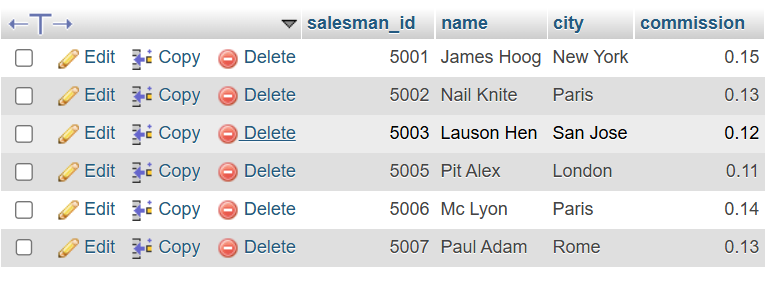
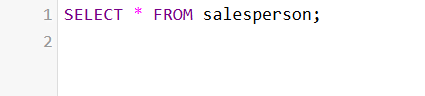
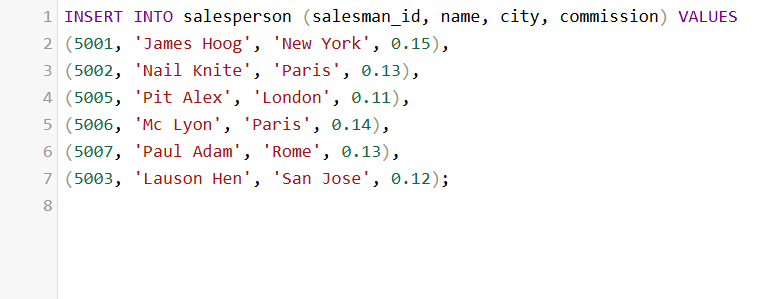
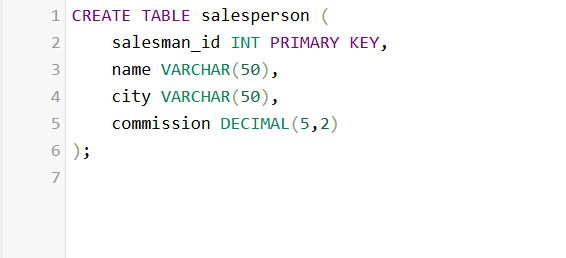
16. All salespeople with commission between 0.10 and 0.12. (Boundary valuesshould be excluded).



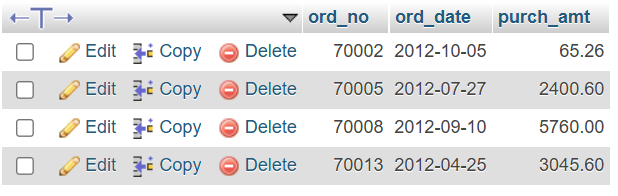
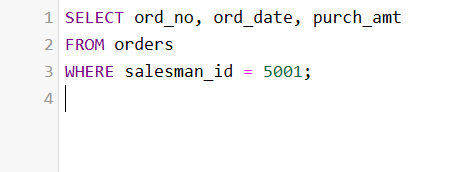
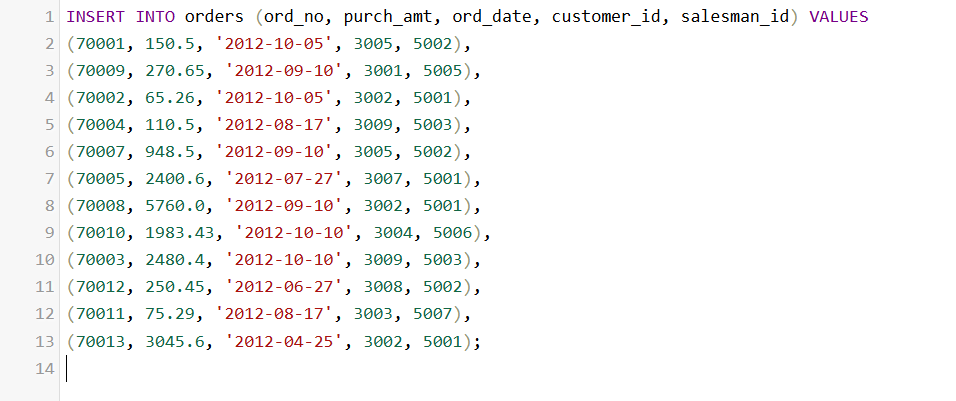
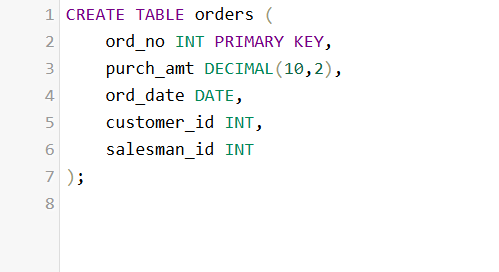
17. All customers excluding those with rating <= 100 unless they are located in Rome



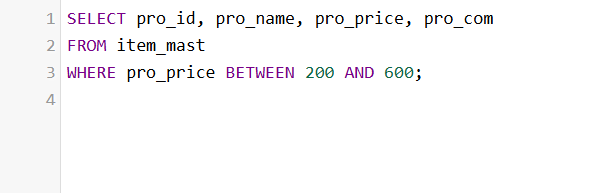
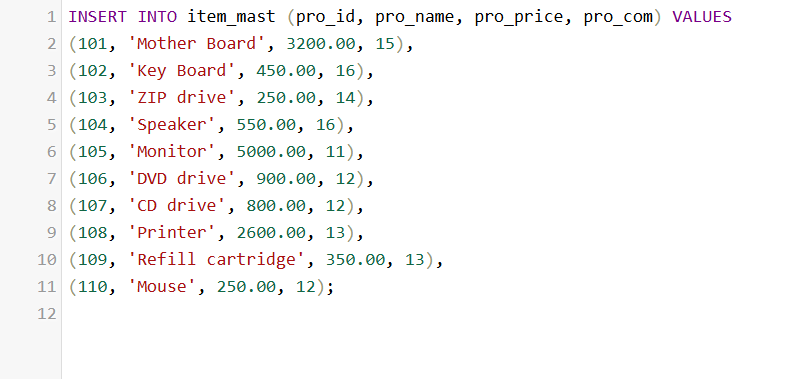
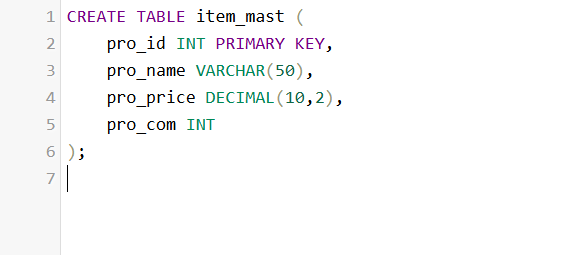
18. Write a SQL statement that displays all the information about all salespeople.

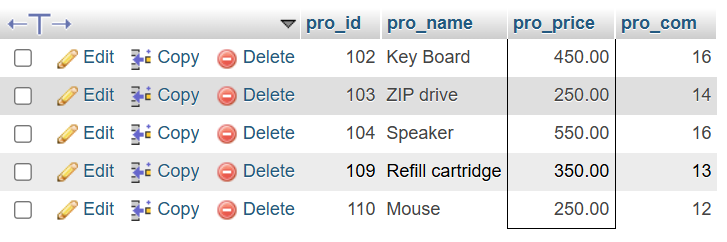


19. From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord\_no, ord\_date, purch\_amt.

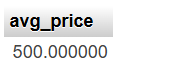
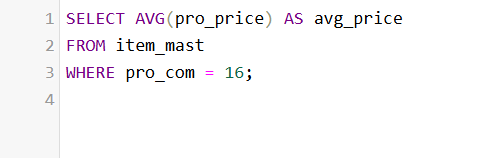


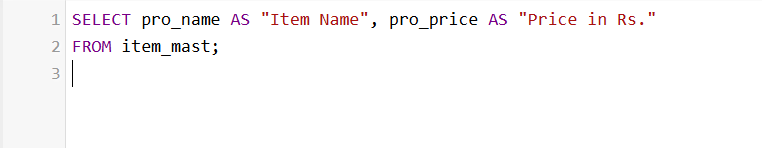
20. From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro\_id, pro\_name, pro\_price, and pro\_com.

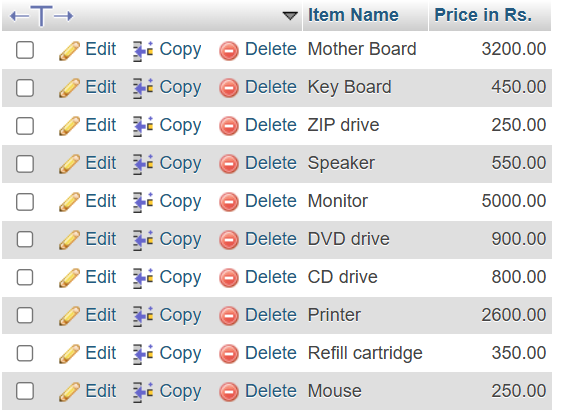




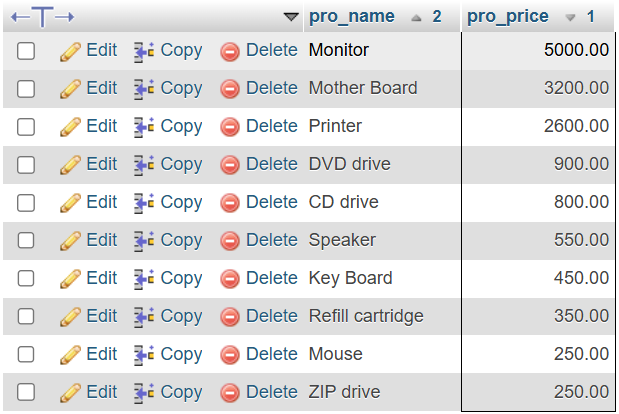
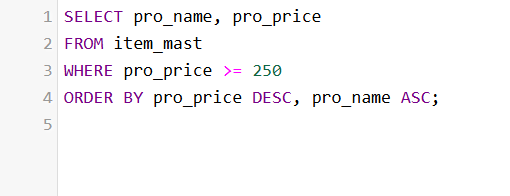
21. From the following table, write a SQL query to calculate the average price for a manufacturer code of 16. Return avg.



22. From the following table, write a SQL query to display the pro\_name as 'Item Name' and pro\_priceas '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.

