**MODULE:5 (Database)**

**Topics Covered – Basics of Database**

**1. What do you understand By Database**

Ans. Database is like a storage area where we can store data or it is a place where data get stored

**2. What is Normalization?**

Ans. Normalization is the process of organizing data in a database to reduce redundancy (duplicate data) and ensure data is logically stored.

**3. What is Difference between DBMS and RDBMS?**

Ans. A **DBMS** (Database Management System) is software for storing and managing data, typically without structured relationships, whereas an **RDBMS** (Relational Database Management System) is a type of DBMS that stores data in structured tables with defined relationships.

**4. What is MF Cod Rule of RDBMS Systems?**

Ans. The **Codd's Rules**, developed by Dr. Edgar F. Codd in 1985, define a set of 13 principles or rules (numbered 0 to 12) to evaluate whether a database management system is truly **relational** (RDBMS).

**5. What do you understand By Data Redundancy?**

Ans. Data Redundancy means when a same data is stored at different location in database which causes wastage of memory.

**6. What is DDL Interpreter?**

Ans. A **DDL Interpreter** is a database system component that processes **Data Definition Language (DDL)** commands (such as CREATE, ALTER, and DROP) to manage database

**7. What is DML Compiler in SQL?**

Ans. A **DML Compiler** in SQL is a component of a database management system that translates **Data Manipulation Language (DML)** commands—such as INSERT, UPDATE, DELETE, and SELECT—into low-level instructions the database can execute.

**8. What is SQL Key Constraints writing an Example of SQL Key Constraints**

Ans. SQL Key constraints are the rules that are given to the table

1. Primary Key
2. Not null
3. Unique
4. Foreign key
5. Check

Example 1

CREATE TABLE Customers (

CustomerID INT PRIMARY KEY,

FirstName VARCHAR(50),

LastName VARCHAR(50),

Email VARCHAR(100) UNIQUE

);

Example 2

CREATE TABLE Orders (

OrderID INT not null,

OrderDate DATE,

CustomerID INT,

FOREIGN KEY (CustomerID) REFERENCES Customers(CustomerID)

);

Example 3

CREATE TABLE Products (

ProductID INT PRIMARY KEY,

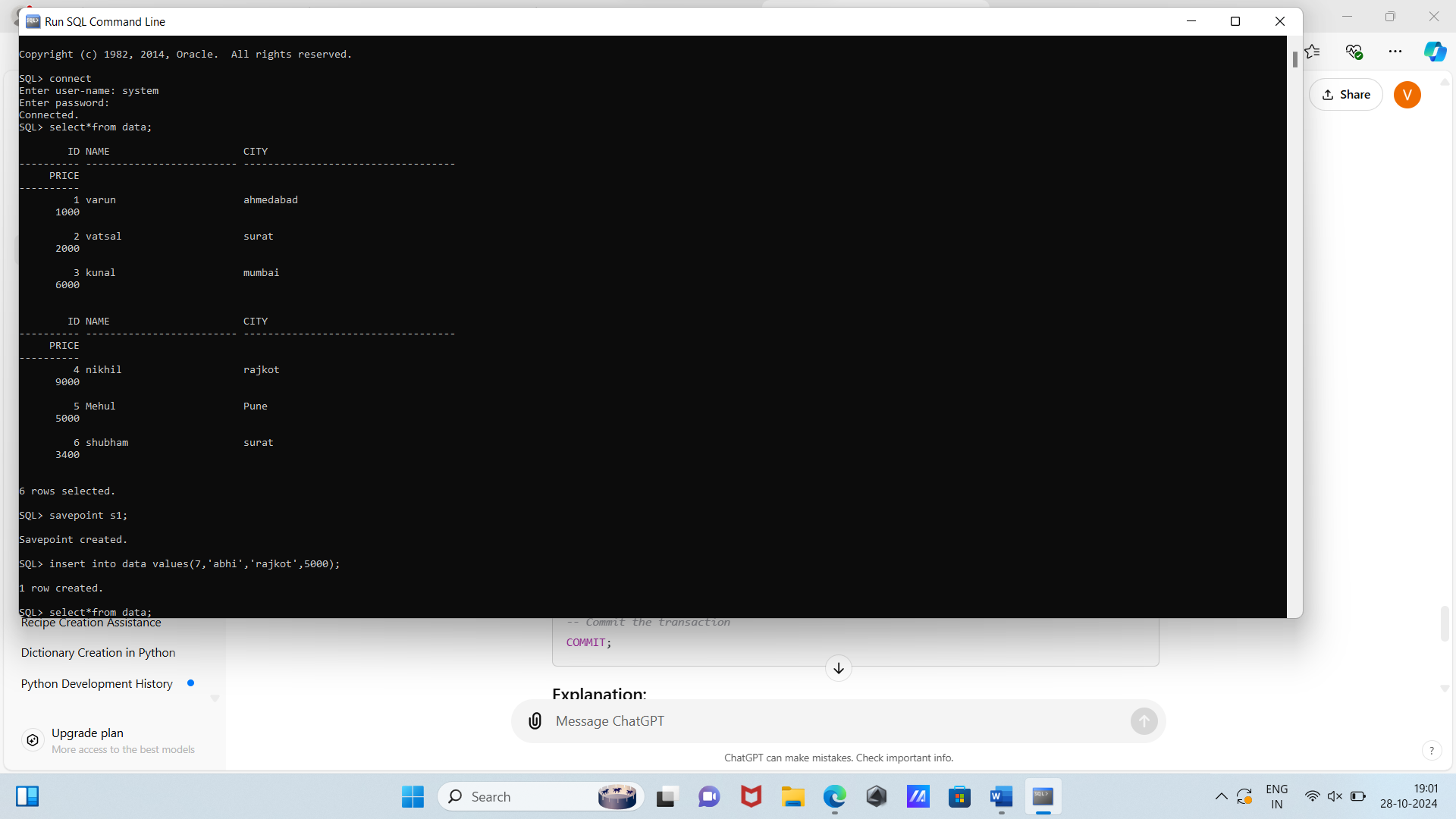
ProductName VARCHAR(100) NOT NULL,

Price DECIMAL(10, 2) CHECK (Price > 0),

Quantity INT CHECK (Quantity >= 0)

);

**9. What is save Point? How to create a save Point write a Query?**

Ans. A save point is a marker in a database which we can rollback means it saves that data temporarily and if we made any changes in the data we can do unto by roll back

Example

**10.What is trigger and how to create a Trigger in SQL?**

Ans. A trigger is a special type of stored procedure in a database that automatically executes (or "fires") in response to specific events on a table, such as INSERT, UPDATE, or DELETE operations.

Example:

CREATE TRIGGER trigger\_name

AFTER | BEFORE | INSTEAD OF trigger\_event

ON table\_name

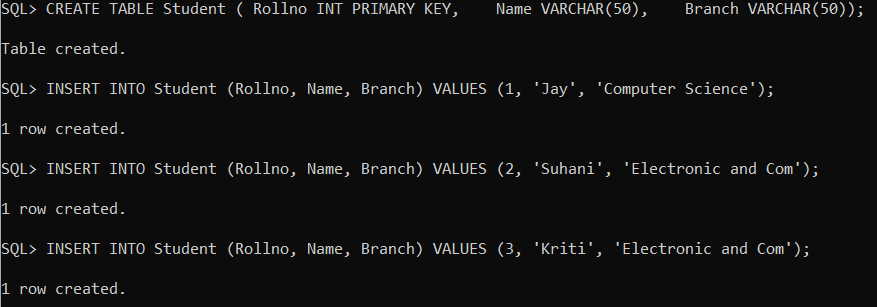
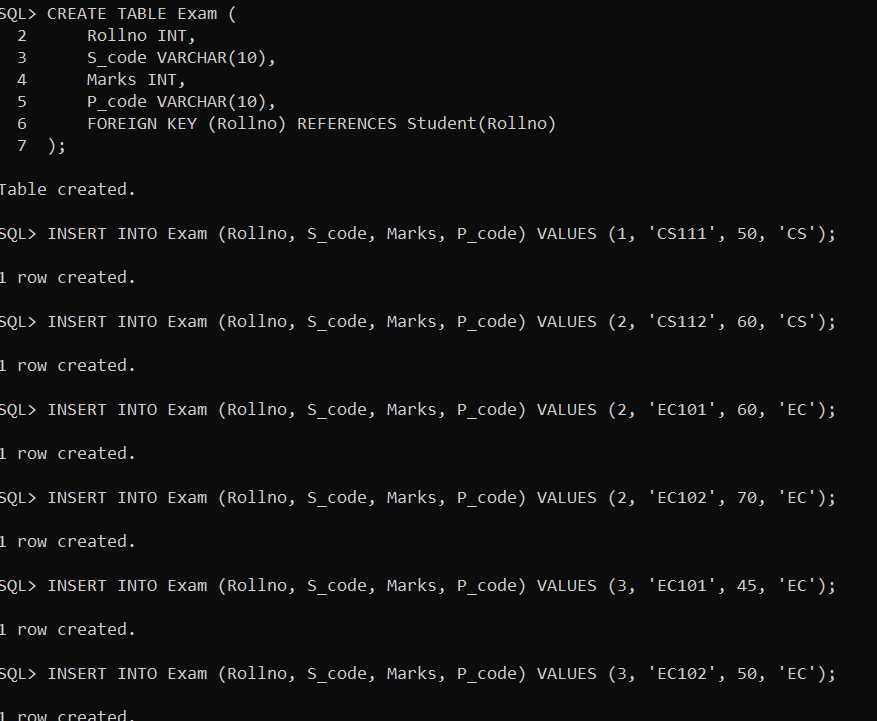
FOR EACH ROW

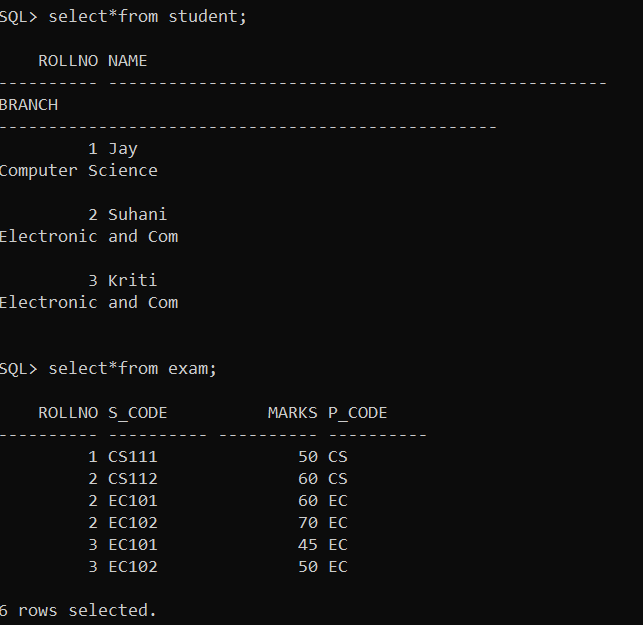
BEGIN

-- Trigger actions (SQL statements)

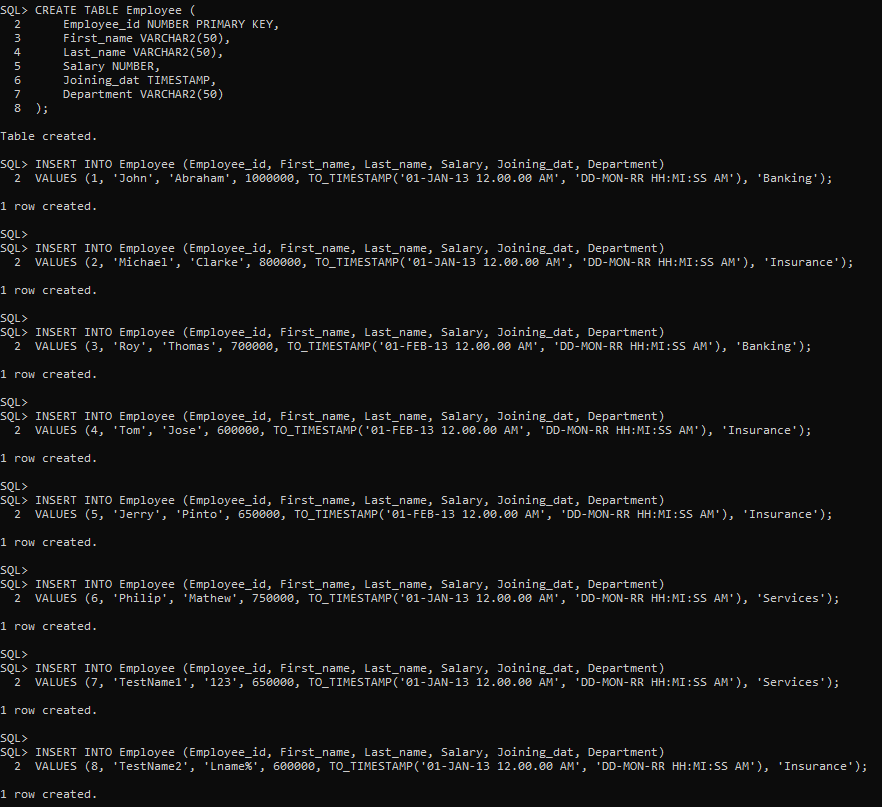
END;

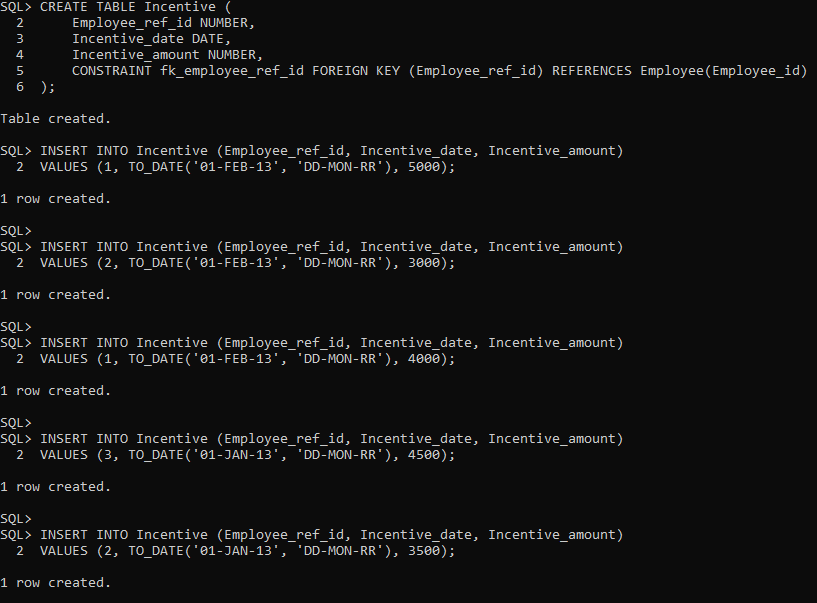
* **Topics Covered – SQL Queries**

**1. Create Table Name : Student and Exam**

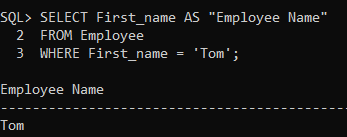


**2. Create table given below: Employee and Incentive Table**

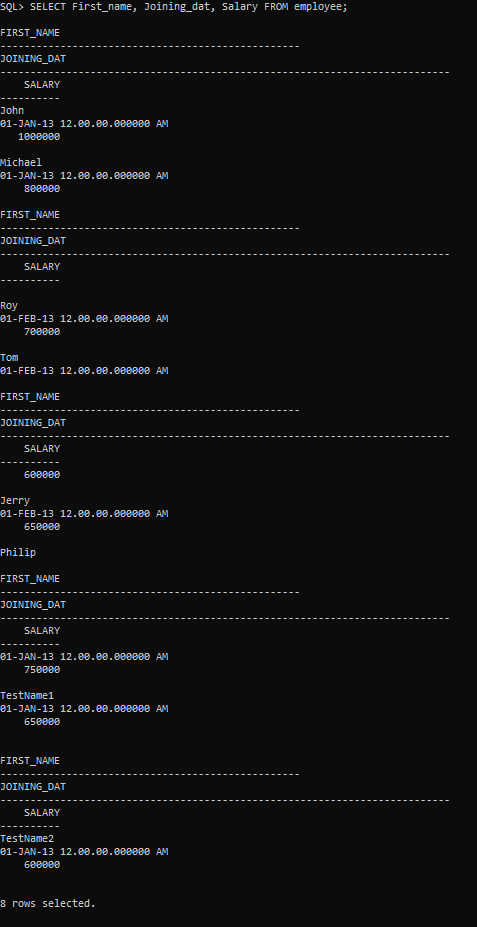
Employee table

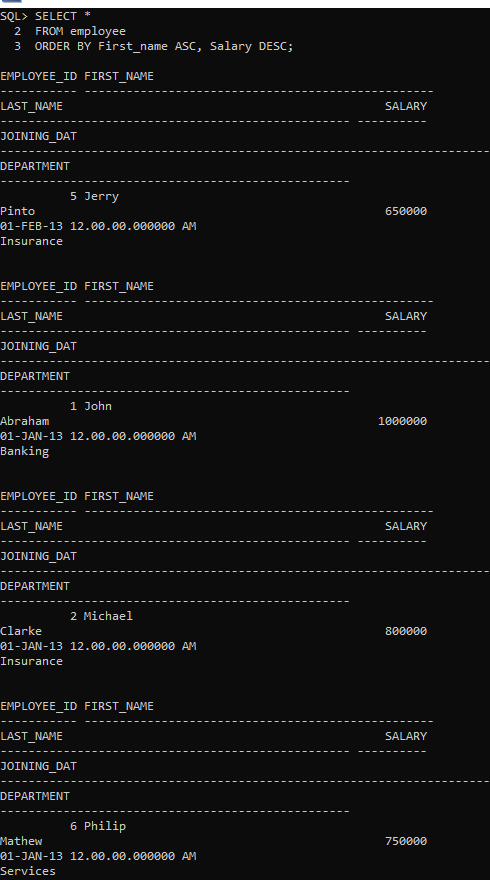
Incentive table

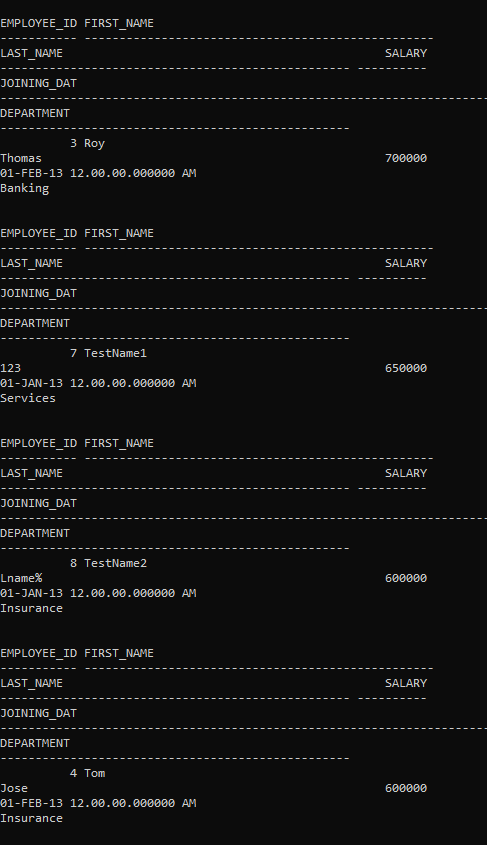
**3. Get First\_Name from employee table using Tom name “Employee Name”.**



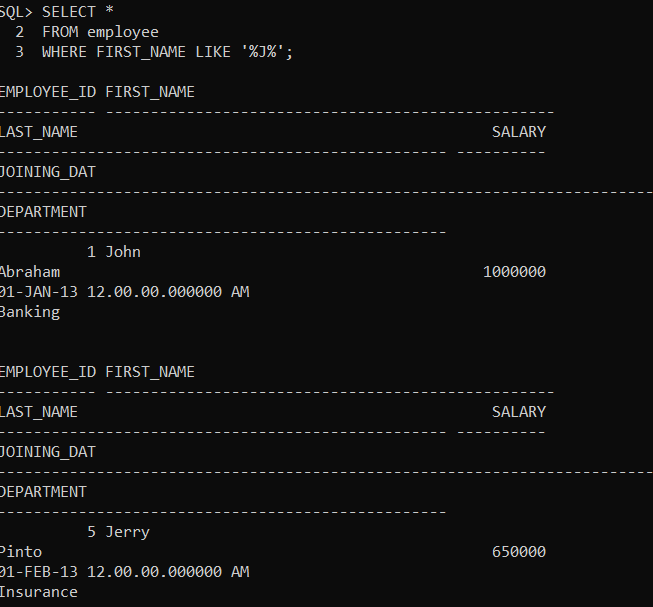
**4. Get FIRST\_NAME, Joining Date, and Salary from employee table.**



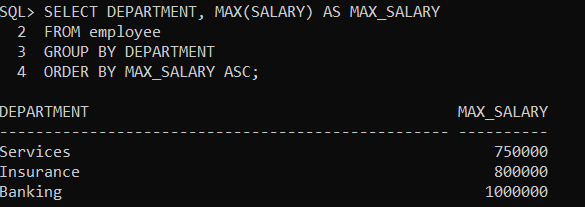
**5. Get all employee details from the employee table order by First\_Name ascending and Salary descending?**



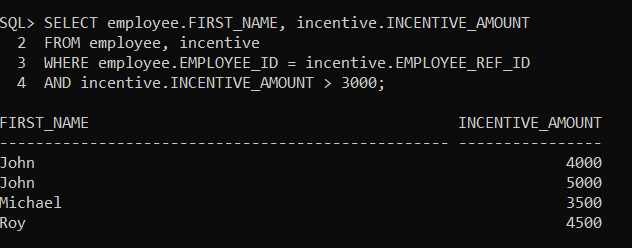
**6. Get employee details from employee table whose first name contains ‘J’.**



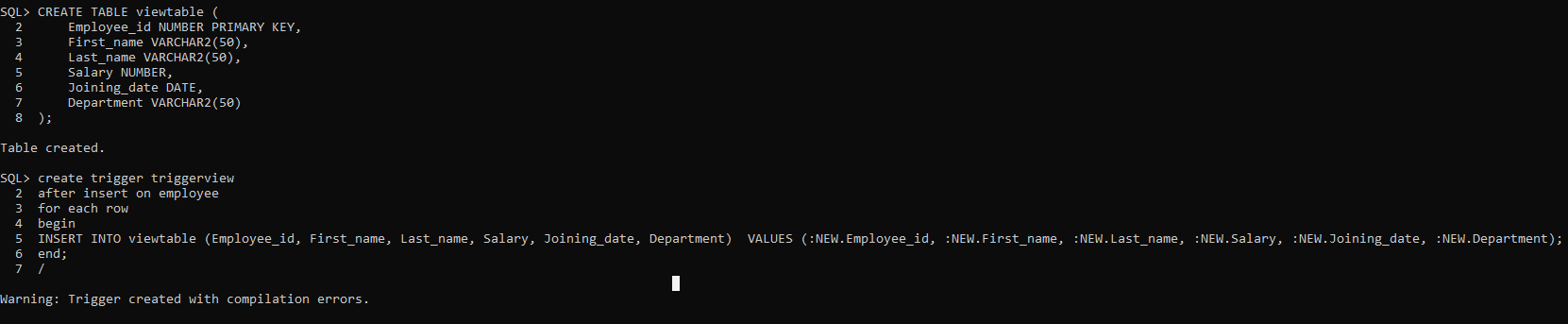
**7. Get department wise maximum salary from employee table order by salary ascending?**



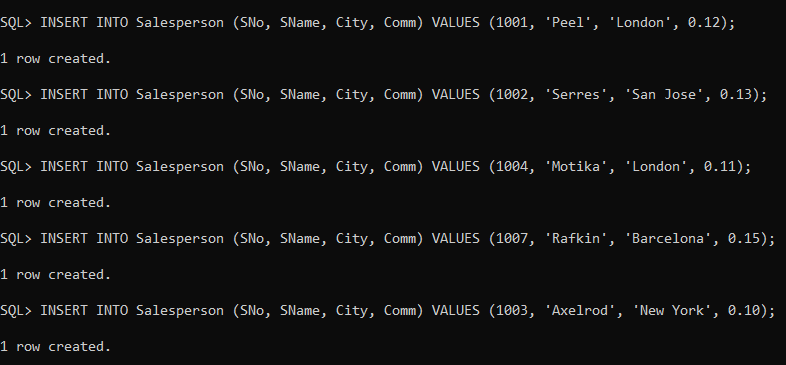
**9. Select first\_name, incentive amount from employee and incentives table for those employees who have incentives and incentive amount greater than 3000**

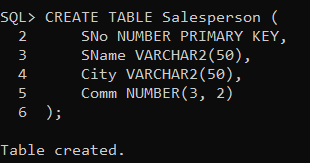


**10.Create After Insert trigger on Employee table which insert records in view table**

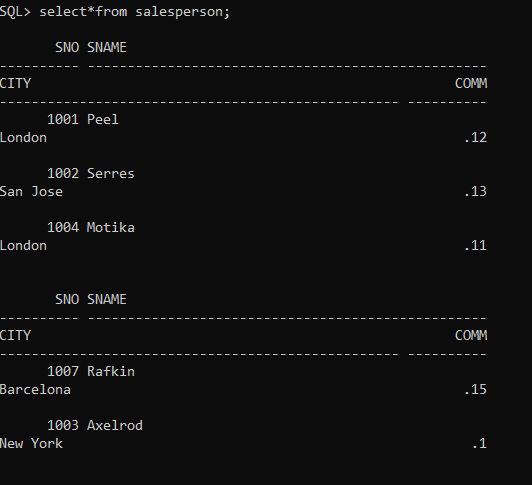


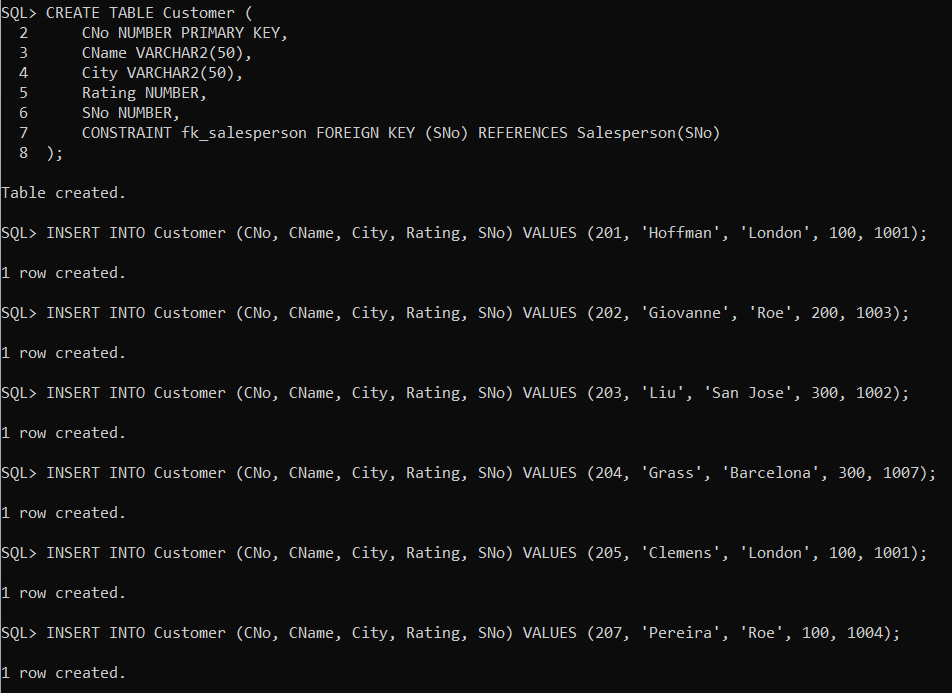
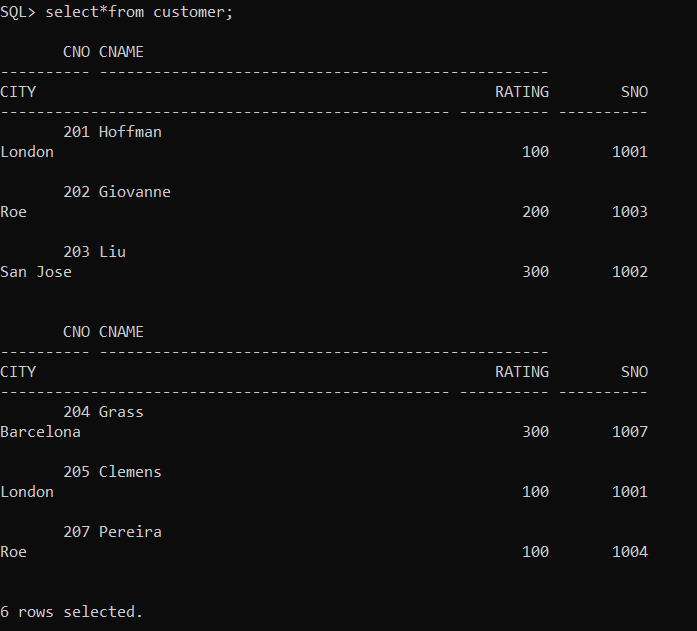
**11. Create table given below: Salesperson and Customer**

 Salesperson Table

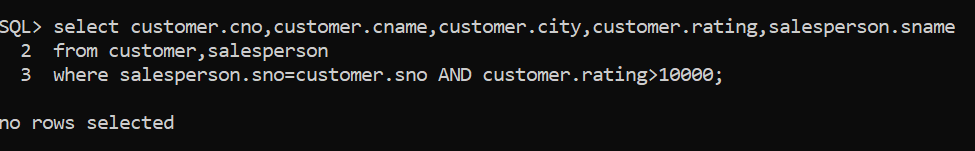


**output**

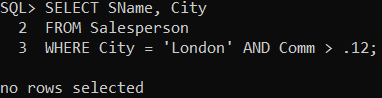


Customer Table  **Output**

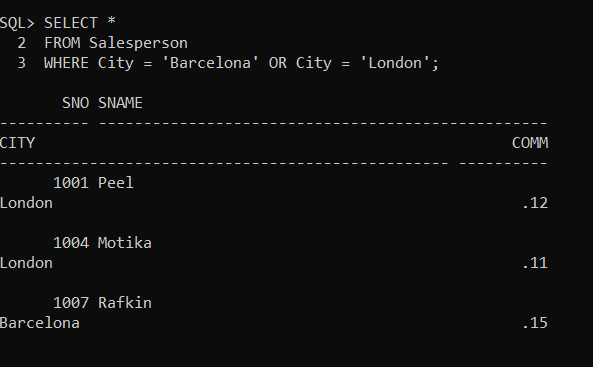
**12. Retrieve the below data from above table All orders for more than $1000.**



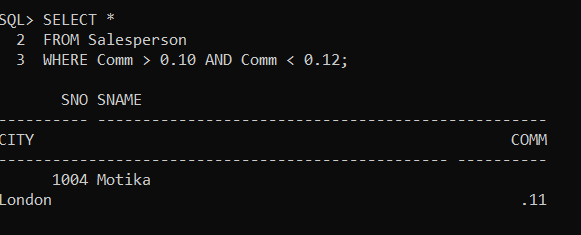
**14.Names and cities of all salespeople in London with commission above 0.12**



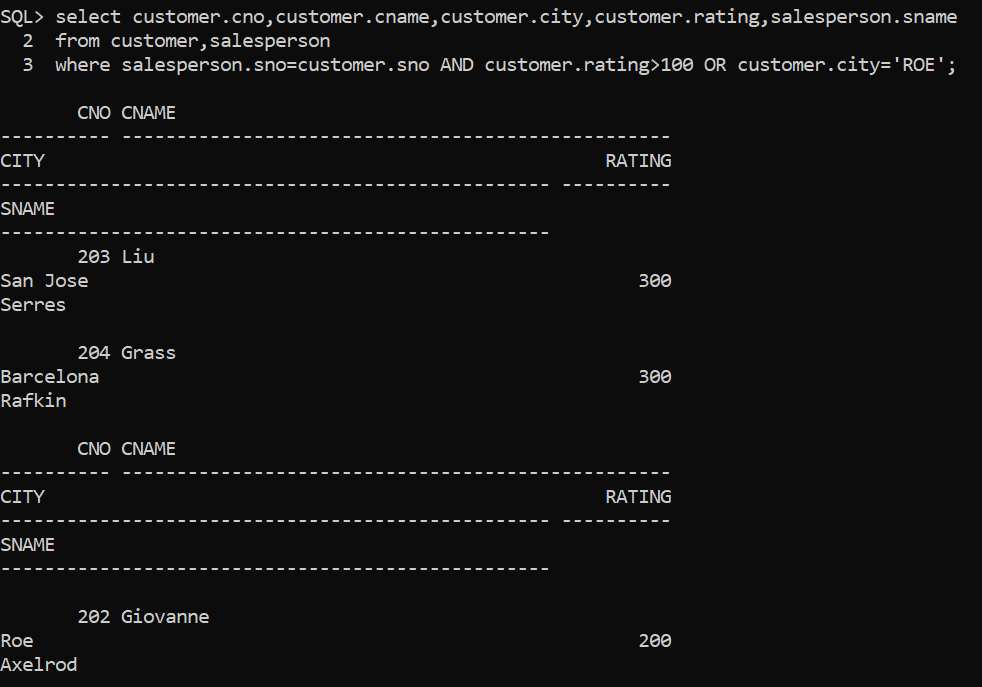
**15.All salespeople either in Barcelona or in London**

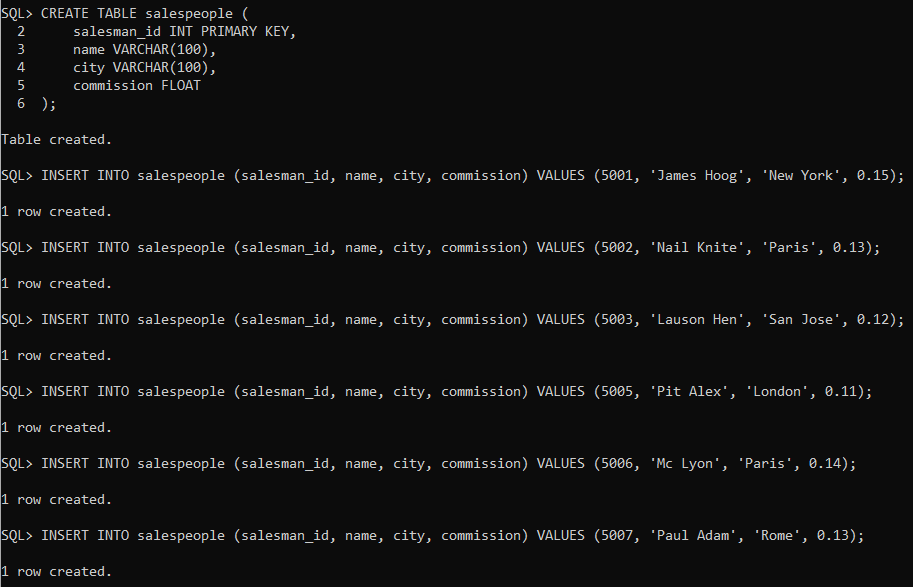


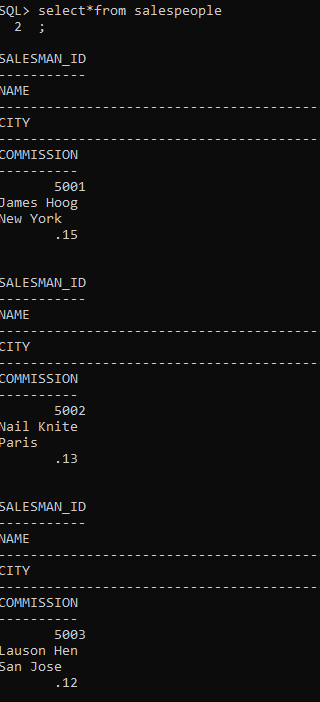
**16. All salespeople with commission between 0.10 and 0.12. (Boundary values should be excluded).**



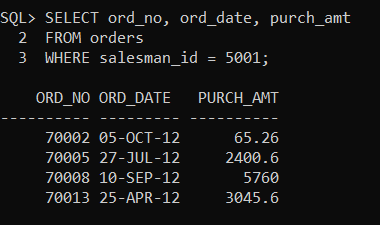
**17. All customers excluding those with rating <= 100 unless they are located in Roe**



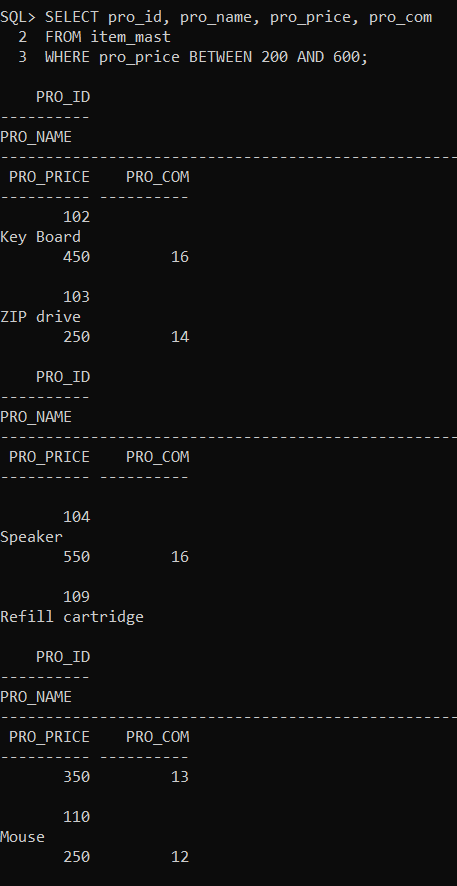
**18. Write a SQL statement that displays all the information about all sales people**

** Output**

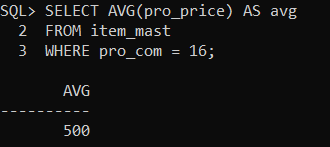
**19. From the following table, write a SQL query to find orders that are delivered by a salesperson with ID. 5001. Return ord\_no, ord\_date, purch\_amt.**

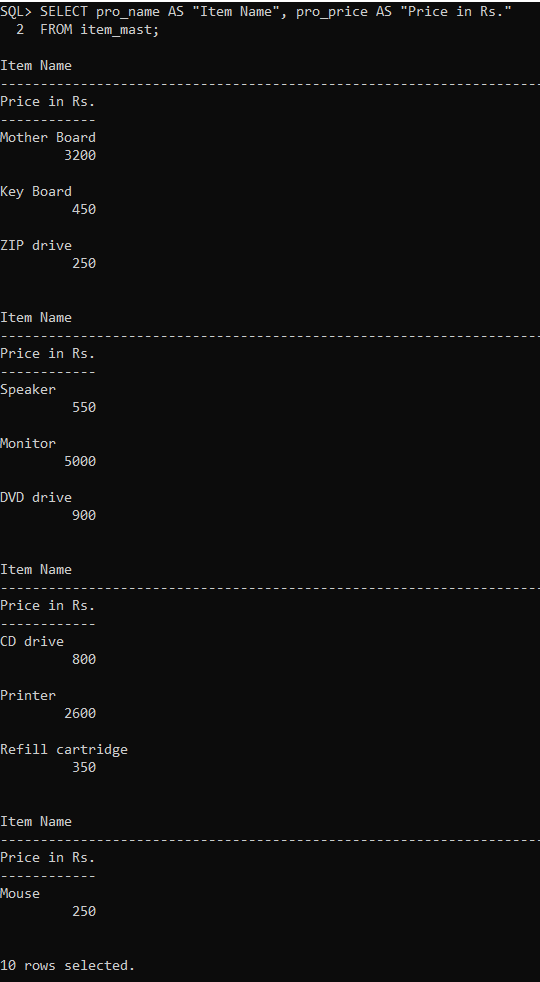


**20. From the following table, write a SQL query to select a range of products whose price is in the range Rs.200 to Rs.600. Begin and end values are included. Return pro\_id, pro\_name, pro\_price, and pro\_com.**

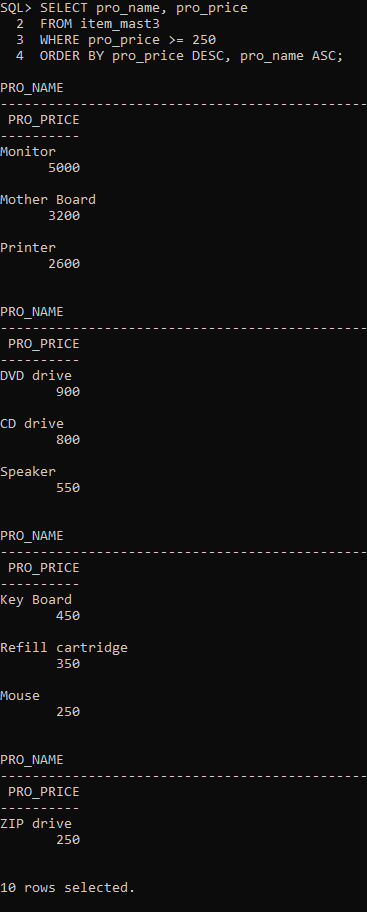
****

**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\_prices '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.**

