|  |  |
| --- | --- |
| 1. SQL Exercise - Advanced concepts | Exercise 1: Ranking and Window Functions |

CREATE TABLE Products (

ProductID INT PRIMARY KEY,

ProductName VARCHAR(50),

Category VARCHAR(50),

Price DECIMAL(10, 2)

);

INSERT INTO Products (ProductID, ProductName, Category, Price) VALUES

(1, 'Laptop', 'Electronics', 80000),

(2, 'TV', 'Electronics', 60000),

(3, 'Camera', 'Electronics', 60000),

(4, 'Jeans', 'Fashion', 2000),

(5, 'Shirt', 'Fashion', 1500),

(6, 'Jacket', 'Fashion', 2000),

(7, 'Watch', 'Fashion', 3000);

SELECT \*

FROM (

SELECT \*,

ROW\_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS RowNum,

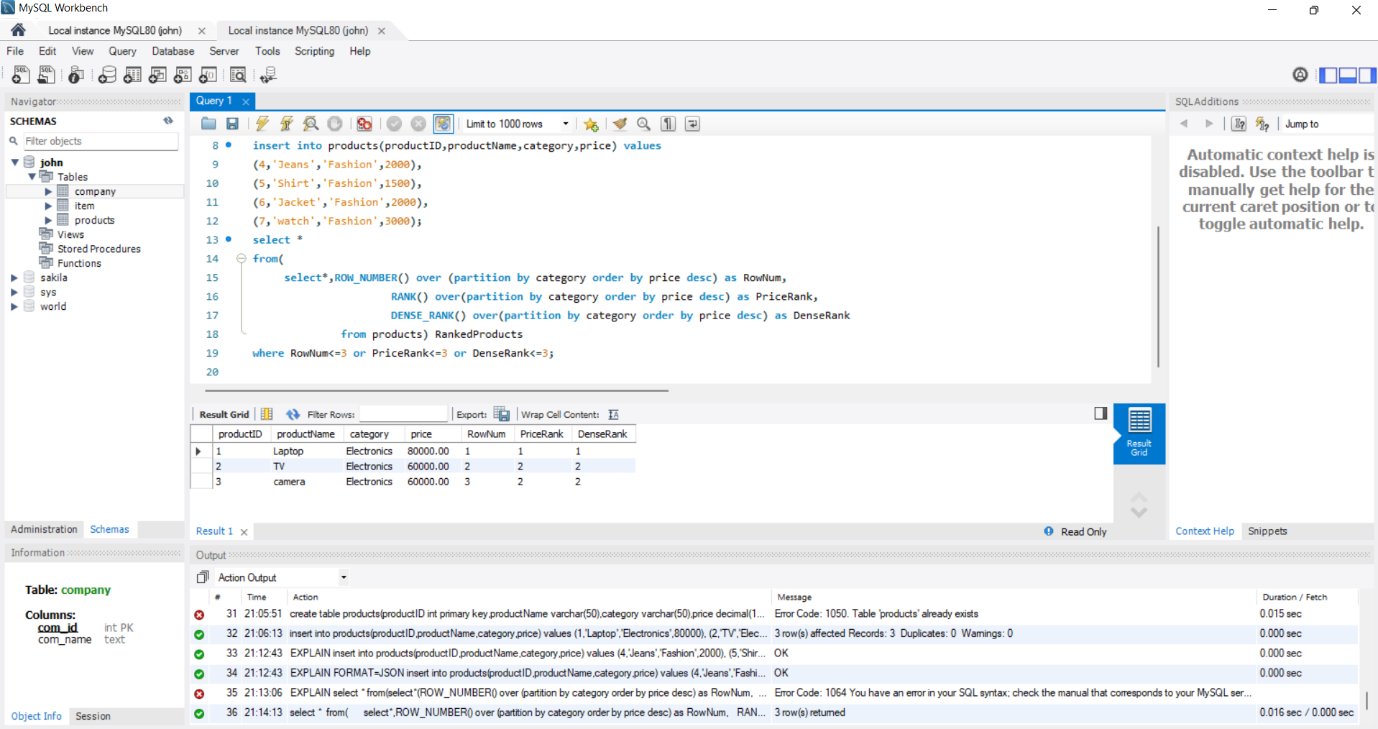
RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS PriceRank,

DENSE\_RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS DenseRank

FROM Products

) RankedProducts

WHERE RowNum <= 3 OR PriceRank <= 3 OR DenseRank <= 3;



|  |  |
| --- | --- |
| 4. SQL Exercise - Stored procedure | Exercise 1: Create a Stored Procedure |

use dbms;

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY,

DepartmentName VARCHAR(100)

);

CREATE TABLE Employees (

EmployeeID INT PRIMARY KEY AUTO\_INCREMENT,

FirstName VARCHAR(50),

LastName VARCHAR(50),

DepartmentID INT,

Salary DECIMAL(10,2),

JoinDate DATE,

FOREIGN KEY (DepartmentID) REFERENCES Departments(DepartmentID)

);

INSERT INTO Departments (DepartmentID, DepartmentName) VALUES

(1, 'HR'),

(2, 'Finance'),

(3, 'IT'),

(4, 'Marketing');

INSERT INTO Employees (FirstName, LastName, DepartmentID, Salary, JoinDate) VALUES

('John', 'Doe', 1, 5000.00, '2020-01-15'),

('Jane', 'Smith', 2, 6000.00, '2019-03-22'),

('Michael', 'Johnson', 3, 7000.00, '2018-07-30'),

('Emily', 'Davis', 4, 5500.00, '2021-11-05');

DELIMITER $$

CREATE PROCEDURE sp\_GetEmployeesByDepartment(IN DeptID INT)

BEGIN

SELECT

E.EmployeeID,

E.FirstName,

E.LastName,

D.DepartmentName,

E.Salary,

E.JoinDate

FROM

Employees E

INNER JOIN

Departments D ON E.DepartmentID = D.DepartmentID

WHERE

E.DepartmentID = DeptID;

END$$

DELIMITER ;

CALL sp\_GetEmployeesByDepartment(3);

DELIMITER $$

CREATE PROCEDURE sp\_InsertEmployee (

IN FirstName VARCHAR(50),

IN LastName VARCHAR(50),

IN DepartmentID INT,

IN Salary DECIMAL(10,2),

IN JoinDate DATE

)

BEGIN

INSERT INTO Employees (FirstName, LastName, DepartmentID, Salary, JoinDate)

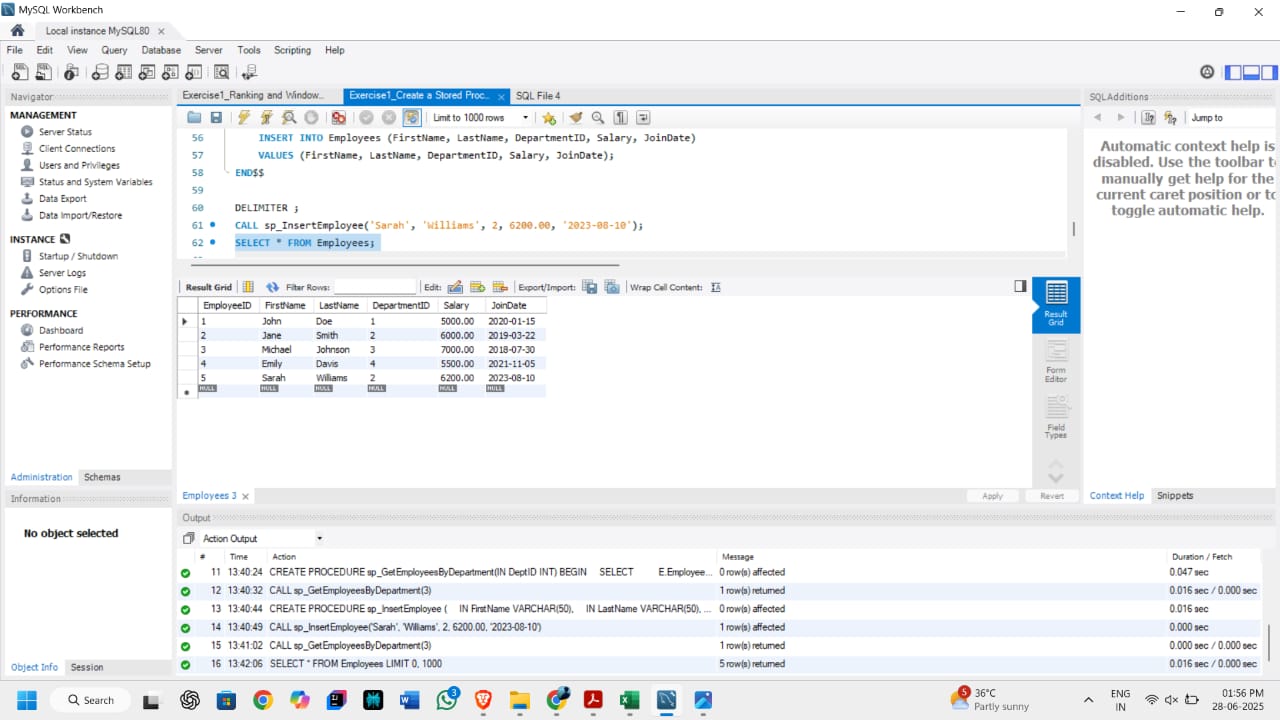
VALUES (FirstName, LastName, DepartmentID, Salary, JoinDate);

END$$

DELIMITER ;

CALL sp\_InsertEmployee('Sarah', 'Williams', 2, 6200.00, '2023-08-10');

SELECT \* FROM Employees;



|  |  |
| --- | --- |
| 4. SQL Exercise - Stored procedure | Exercise 5: Return Data from a Stored Procedure |

use dbms;

DELIMITER $$

CREATE PROCEDURE sp\_CountEmployeesByDepartment (

IN dept\_id INT

)

BEGIN

SELECT COUNT(\*) AS TotalEmployees

FROM Employees

WHERE DepartmentID = dept\_id;

END $$

DELIMITER ;

CALL sp\_CountEmployeesByDepartment(2);