SQL Q1)

-- Create and use the database

CREATE DATABASE OnlineRetailDb;

GO

USE OnlineRetailDB;

GO

-- Reset Products table if it exists

DROP TABLE IF EXISTS Products;

-- Create table

CREATE TABLE Products (

    ProductID INT PRIMARY KEY,

    ProductName VARCHAR(100),

    Category VARCHAR(50),

    Price DECIMAL(10, 2)

);

-- Insert sample products

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

(1, 'Laptop', 'Electronics', 1200.00),

(2, 'Smartphone', 'Electronics', 800.00),

(3, 'Tablet', 'Electronics', 800.00),

(4, 'Headphones', 'Accessories', 150.00),

(5, 'Mouse', 'Accessories', 150.00),

(6, 'Keyboard', 'Accessories', 100.00);

-- Exercise 1: ROW\_NUMBER

SELECT

    ProductID, ProductName, Category, Price,

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

FROM Products;

-- Exercise 1: RANK

SELECT

    ProductID, ProductName, Category, Price,

    RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS RankNum

FROM Products;

-- Exercise 1: DENSE\_RANK

SELECT

    ProductID, ProductName, Category, Price,

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

FROM Products;

-- Show only Top 3 products per category using ROW\_NUMBER

WITH RankedProducts AS (

    SELECT \*, ROW\_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS rn

    FROM Products

)

SELECT \* FROM RankedProducts WHERE rn <= 3;

SQLQ2)  
-- Drop existing procedures if they exist

IF OBJECT\_ID('sp\_InsertEmployee', 'P') IS NOT NULL

    DROP PROCEDURE sp\_InsertEmployee;

GO

IF OBJECT\_ID('sp\_GetEmployeesByDepartment', 'P') IS NOT NULL

    DROP PROCEDURE sp\_GetEmployeesByDepartment;

GO

-- Create Departments table

CREATE TABLE Departments (

    DepartmentID INT PRIMARY KEY,

    DepartmentName VARCHAR(100)

);

GO

-- Create Employees table

CREATE TABLE Employees (

    EmployeeID INT PRIMARY KEY IDENTITY(1,1),

    FirstName VARCHAR(50),

    LastName VARCHAR(50),

    DepartmentID INT FOREIGN KEY REFERENCES Departments(DepartmentID),

    Salary DECIMAL(10,2),

    JoinDate DATE

);

GO

-- Insert sample data into Departments

INSERT INTO Departments (DepartmentID, DepartmentName) VALUES

(1, 'HR'),

(2, 'Finance'),

(3, 'IT'),

(4, 'Marketing');

GO

-- Insert sample data into Employees

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');

GO

-- Create stored procedure to insert a new employee

CREATE PROCEDURE sp\_InsertEmployee

    @FirstName VARCHAR(50),

    @LastName VARCHAR(50),

    @DepartmentID INT,

    @Salary DECIMAL(10,2),

    @JoinDate DATE

AS

BEGIN

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

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

END;

GO

-- Create stored procedure to get employees by department

CREATE PROCEDURE sp\_GetEmployeesByDepartment

    @DepartmentID INT

AS

BEGIN

    SELECT \* FROM Employees

    WHERE DepartmentID = @DepartmentID;

END;

GO

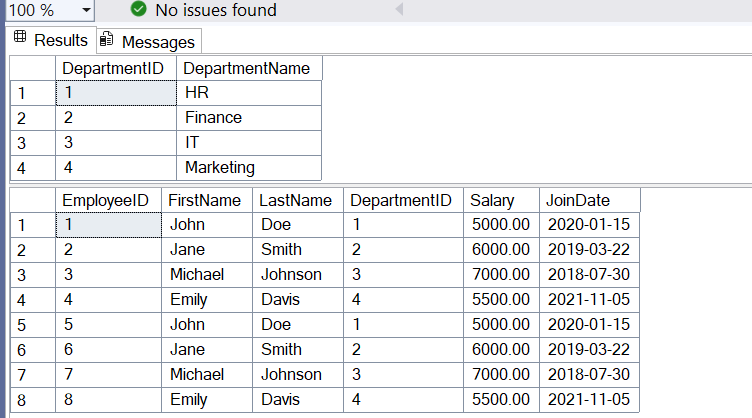
-- Example: Insert new employee

EXEC sp\_InsertEmployee 'Alice', 'Williams', 3, 7200.00, '2023-10-01';

-- Example: Get all employees from department 3

EXEC sp\_GetEmployeesByDepartment 3;

OUTPUT)



SQL Q3)

USE EmployeeDB;

GO

-- Drop the procedure if it already exists

IF OBJECT\_ID('sp\_GetEmployeeCountByDepartment', 'P') IS NOT NULL

    DROP PROCEDURE sp\_GetEmployeeCountByDepartment;

GO

-- Create the procedure

CREATE PROCEDURE sp\_GetEmployeeCountByDepartment

    @DepartmentID INT

AS

BEGIN

    SELECT COUNT(\*) AS EmployeeCount

    FROM Employees

    WHERE DepartmentID = @DepartmentID;

END;

GO

EXEC sp\_GetEmployeeCountByDepartment @DepartmentID = 3;

OUTPUT)

