**ARCHIT GAUTAM   
SUPERSET ID: 6361966**

**WEEK-2 HANDS ON:**

1. **SQL Exercise - Advanced Concpts**

Exercise 1: Ranking and window function

#code

-- 1. Drop the table if it already exists

DROP TABLE IF EXISTS Products\_new;

-- 2. Create the Products\_new table

CREATE TABLE Products\_new (

ProductID INT PRIMARY KEY,

ProductName VARCHAR(50),

Category VARCHAR(50),

Price DECIMAL(10, 2)

);

-- 3. Insert sample data

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

(1, 'Laptop', 'Electronics', 1200),

(2, 'Smartphone', 'Electronics', 1000),

(3, 'Headphones', 'Electronics', 300),

(4, 'Blender', 'Kitchen', 150),

(5, 'Microwave', 'Kitchen', 200),

(6, 'Toaster', 'Kitchen', 150),

(7, 'Monitor', 'Electronics', 300);

-- 4. View the entire table

SELECT \* FROM Products\_new;

-- 5. Use ROW\_NUMBER() to get top 3 per category (unique ranks)

PRINT '--- Top 3 using ROW\_NUMBER() ---';

SELECT \*

FROM (

SELECT

ProductID,

ProductName,

Category,

Price,

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

FROM Products\_new

) AS Ranked

WHERE RowNum <= 3;

-- 6. Use RANK() to get top 3 per category (including ties)

PRINT '--- Top 3 using RANK() ---';

SELECT \*

FROM (

SELECT

ProductID,

ProductName,

Category,

Price,

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

FROM Products\_new

) AS Ranked

WHERE PriceRank <= 3;

-- 7. Use DENSE\_RANK() to get top 3 per category (no gaps)

PRINT '--- Top 3 using DENSE\_RANK() ---';

SELECT \*

FROM (

SELECT

ProductID,

ProductName,

Category,

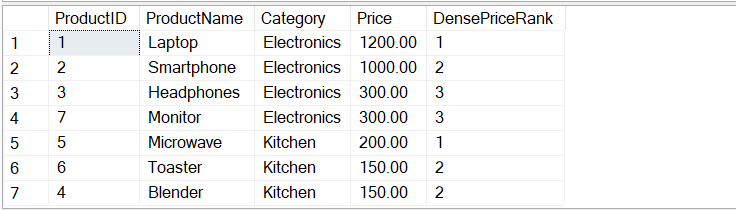
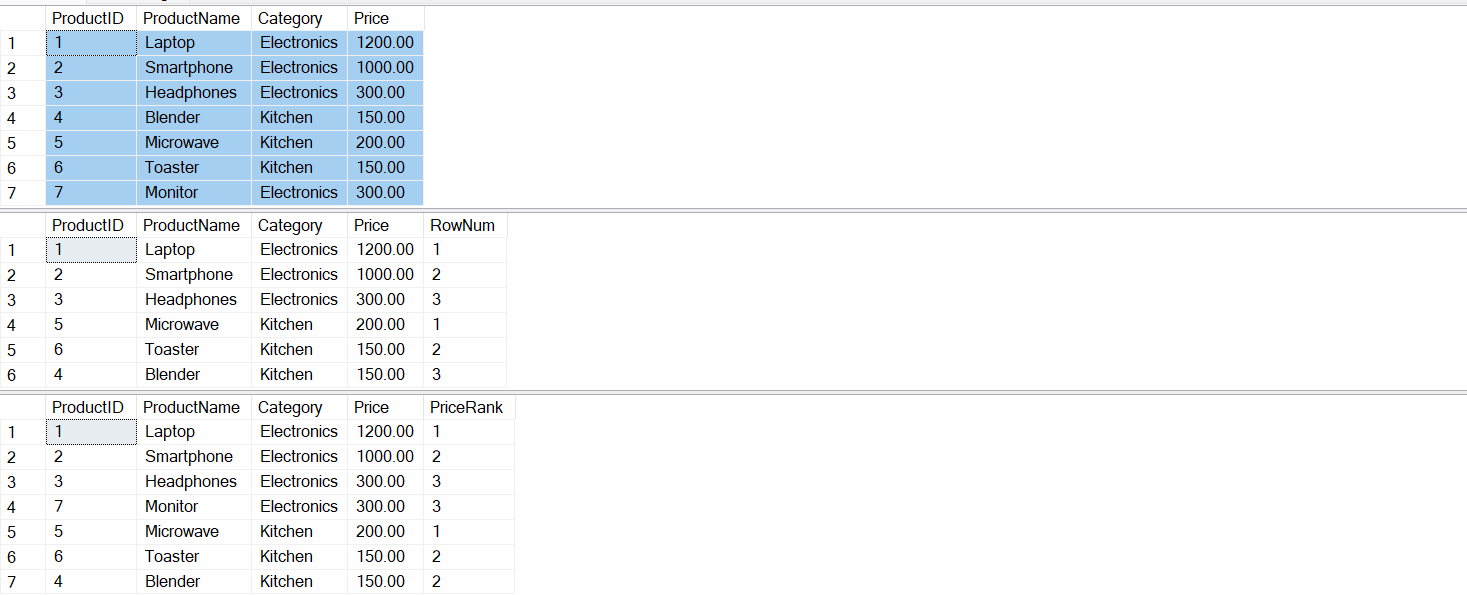
Price,

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

FROM Products\_new

) AS Ranked

WHERE DensePriceRank <= 3;



1. **Stored Procedure - exercise 1:**

-- 1. Drop existing tables if they exist

DROP TABLE IF EXISTS Employees;

DROP TABLE IF EXISTS Departments;

GO

-- 2. Create Departments table

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY,

DepartmentName VARCHAR(100)

);

GO

-- 3. 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

-- 4. Insert sample data into Departments

INSERT INTO Departments (DepartmentID, DepartmentName) VALUES

(1, 'HR'),

(2, 'Finance'),

(3, 'IT'),

(4, 'Marketing');

GO

-- 5. 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

-- 6. Stored Procedure to get employees by DepartmentID

CREATE OR ALTER PROCEDURE sp\_GetEmployeesByDepartment

@DeptID INT

AS

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;

GO

-- 7. Stored Procedure to insert a new employee

CREATE OR ALTER 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

EXEC sp\_GetEmployeesByDepartment @DeptID = 3;

EXEC sp\_InsertEmployee

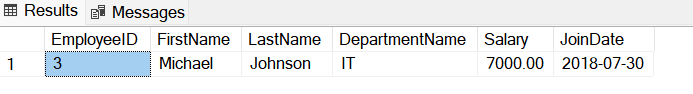
@FirstName = 'Alice',

@LastName = 'Walker',

@DepartmentID = 2,

@Salary = 5800.00,

@JoinDate = '2022-09-10';



**Strored procedure - Exercise 5:**

-- 8. Stored Procedure to count number of employees in a department

CREATE OR ALTER PROCEDURE sp\_GetEmployeeCountByDepartment

@DeptID INT

AS

BEGIN

SELECT

D.DepartmentName,

COUNT(E.EmployeeID) AS EmployeeCount

FROM Departments D

LEFT JOIN Employees E ON D.DepartmentID = E.DepartmentID

WHERE D.DepartmentID = @DeptID

GROUP BY D.DepartmentName;

END;

GO

EXEC sp\_GetEmployeesByDepartment @DeptID = 1;

EXEC sp\_GetEmployeeCountByDepartment @DeptID = 2;

