**1. SQL Exercise - Advanced concepts**

**Exercise 1: Ranking and Window Functions**

CREATE TABLE Products (

ProductID SERIAL PRIMARY KEY,

ProductName VARCHAR(100),

Category VARCHAR(50),

Price DECIMAL(10, 2)

);

INSERT INTO Products (ProductName, Category, Price) VALUES

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

('Monitor', 'Electronics', 1200),

('Camera', 'Electronics', 900),

('Tablet', 'Electronics', 800),

('Phone', 'Electronics', 800),

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

('Mixer', 'Kitchen', 150),

('Oven', 'Kitchen', 300),

('Kettle', 'Kitchen', 200),

('Toaster', 'Kitchen', 100);

SELECT

ProductID,

ProductName,

Category,

Price,

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

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

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

FROM Products;

WITH RankedProducts AS (

SELECT \*,

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

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

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

FROM Products

)

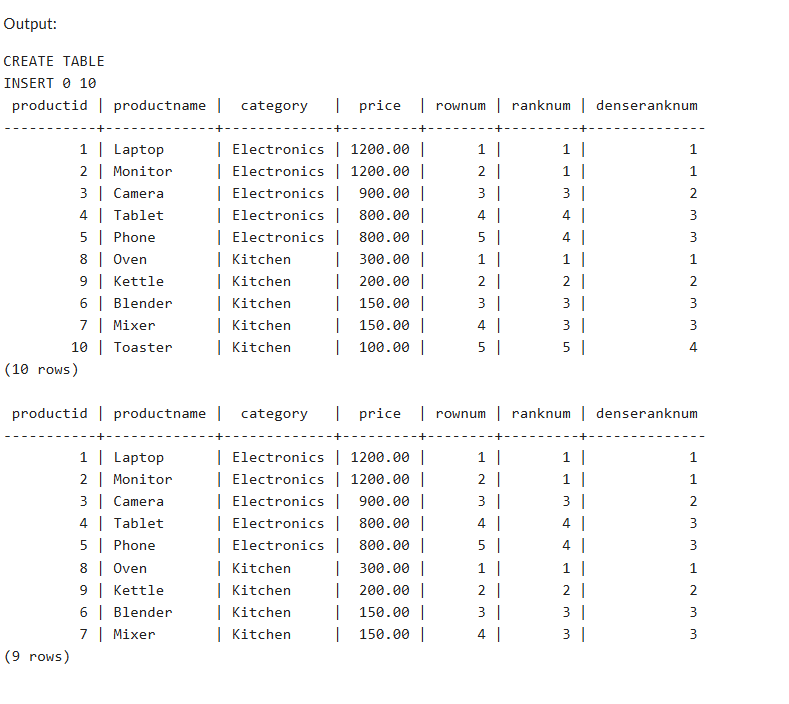
SELECT \*

FROM RankedProducts

WHERE RowNum <= 3 OR RankNum <= 3 OR DenseRankNum <= 3

ORDER BY Category, Price DESC;

**OUTPUT**



**4. SQL Exercise - Stored procedure**

**Exercise 1: Create a Stored Procedure**

SELECT DB\_NAME() AS CurrentDatabase;

GO

IF OBJECT\_ID('Employees', 'U') IS NOT NULL

DROP TABLE Employees;

IF OBJECT\_ID('Departments', 'U') IS NOT NULL

DROP TABLE Departments;

GO

CREATE TABLE Departments (

DepartmentID INT PRIMARY KEY,

DepartmentName VARCHAR(100)

);

GO

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 INTO Departments (DepartmentID, DepartmentName) VALUES

(1, 'HR'),

(2, 'Finance'),

(3, 'IT'),

(4, 'Marketing');

GO

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

VALUES

('Gowri', 'Nandhini', 1, 60000, '2023-06-01'),

('Kiran', 'Kumar', 2, 55000, '2022-04-15'),

('Sita', 'Rani', 1, 58000, '2021-12-20');

GO

-- Drop procedures if they exist

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

DROP PROCEDURE sp\_InsertEmployee;

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

DROP PROCEDURE sp\_GetEmployeesByDepartment;

GO

-- Create sp\_InsertEmployee

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 sp\_GetEmployeesByDepartment

CREATE PROCEDURE sp\_GetEmployeesByDepartment

@DepartmentID INT

AS

BEGIN

SELECT

EmployeeID,

FirstName,

LastName,

DepartmentID,

Salary,

JoinDate

FROM Employees

WHERE DepartmentID = @DepartmentID;

END;

GO

-- Call the procedure to get employees by department

EXEC sp\_GetEmployeesByDepartment @DepartmentID = 1;

GO

-- Call the procedure to insert a new employee

EXEC sp\_InsertEmployee

@FirstName = 'Arjun',

@LastName = 'Reddy',

@DepartmentID = 3,

@Salary = 67000,

@JoinDate = '2024-01-10';

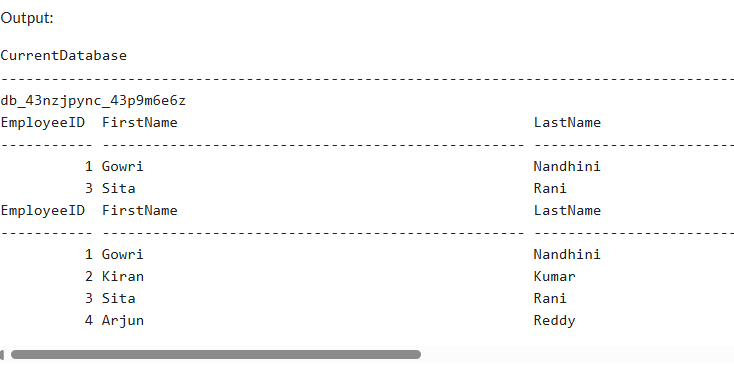
GO

-- Final check: display all employees

SELECT \* FROM Employees;

GO

**OUTPUT**



**Exercise 5: Return Data from a Stored Procedure**

IF OBJECT\_ID('Employees', 'U') IS NOT NULL

DROP TABLE Employees;

IF OBJECT\_ID('Departments', 'U') IS NOT NULL

DROP TABLE Departments;

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', 1, 5500.00, '2021-11-05'),

('Arjun', 'Reddy', 3, 6700.00, '2024-01-10');

GO

-- Drop procedure if it exists

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

DROP PROCEDURE sp\_GetEmployeeCountByDepartment;

GO

-- Create stored procedure to return employee count by department

CREATE PROCEDURE sp\_GetEmployeeCountByDepartment

@DepartmentID INT

AS

BEGIN

SELECT

COUNT(\*) AS EmployeeCount

FROM

Employees

WHERE

DepartmentID = @DepartmentID;

END;

GO

-- Execute the procedure for DepartmentID = 1

EXEC sp\_GetEmployeeCountByDepartment @DepartmentID = 1;

GO

-- Execute the procedure for DepartmentID = 3

EXEC sp\_GetEmployeeCountByDepartment @DepartmentID = 3;

GO

**OUTPUT**

****