**Week 2 : SQL Advanced SQL HandsOn Exercises**

**Exercise 1: Ranking and Window Functions:**

*1. Use ROW\_NUMBER() to assign a unique rank within each category.*

*Query:-*

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

(5, 'Smartwatch', 'Electronics', 800.00),

(6, 'Charger', 'Accessories', 50.00),

(7, 'Mouse', 'Accessories', 150.00),

(8, 'Backpack', 'Accessories', 80.00),

(9, 'Notebook', 'Stationery', 10.00),

(10, 'Pen', 'Stationery', 2.00),

(11, 'Marker', 'Stationery', 5.00),

(12, 'Printer', 'Electronics', 1000.00),

(13, 'Desk Lamp', 'Home Decor', 300.00),

(14, 'Cushion', 'Home Decor', 200.00),

(15, 'Rug', 'Home Decor', 500.00);

SELECT

Category,

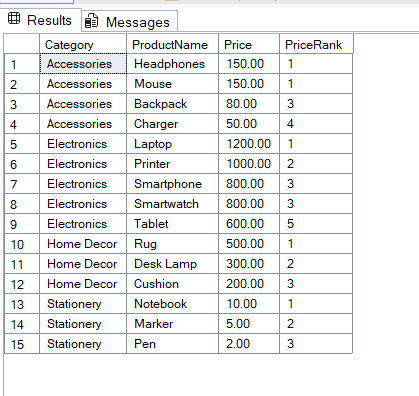
ProductName,

Price,

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

FROM Products;

*Output:-*



*2. Use RANK() and DENSE\_RANK() to compare how ties are handled.*

*Query:-*

SELECT

Category,

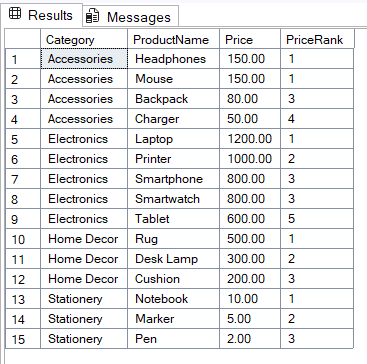
ProductName,

Price,

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

FROM Products;

***Output:-***



***Query:-***

SELECT

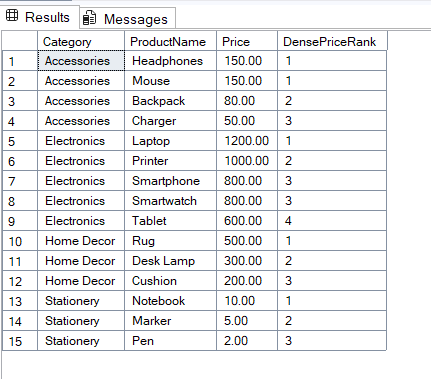
Category,

ProductName,

Price,

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

FROM Products;

******

*3. Use PARTITION BY Category and ORDER BY Price DESC.*

*Query:-*

SELECT \* FROM (

SELECT

Category,

ProductName,

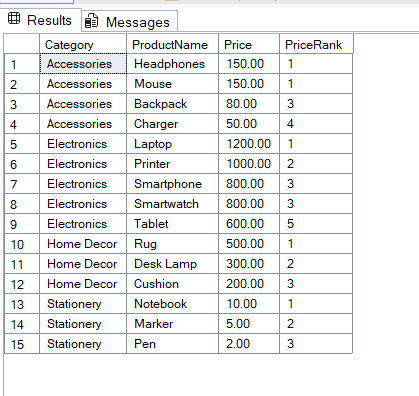
Price,

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

FROM Products

) AS Ranked

WHERE DensePriceRank <= 3;



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

**1.Stored Procedure to Return Total Employees by Department**

CREATE PROCEDURE sp\_CountEmployeesByDepartment

@DepartmentID INT

AS

BEGIN

SELECT

d.DepartmentName,

COUNT(e.EmployeeID) AS TotalEmployees

FROM

Departments d

LEFT JOIN

Employees e ON d.DepartmentID = e.DepartmentID

WHERE

d.DepartmentID = @DepartmentID

GROUP BY

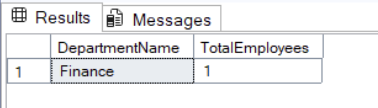
d.DepartmentName;

END;

**2.** **Execute the Stored Procedure**

EXEC sp\_CountEmployeesByDepartment @DepartmentID = 1;

**OUTPUT :-**

****