**WEEK-2**

Task-1

**Exercise 1: Ranking and Window Functions**

Goal: Use ROW\_NUMBER(), RANK(), DENSE\_RANK(), OVER(), and PARTITION BY.

Scenario:

Find the top 3 most expensive products in each category using different ranking functions.

Steps:

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

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

3. Use PARTITION BY Category and ORDER BY Price DESC

**CODE:**

CREATE TABLE Products (

ProductID INT PRIMARY KEY,

ProductName VARCHAR(100),

Category VARCHAR(50),

Price DECIMAL(10, 2)

);

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

(1, 'Laptop', 'Electronics', 1000.00),

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

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

(4, 'Headphones', 'Electronics', 2000.00),

(5, 'T-shirt', 'Clothing', 2500.00),

(6, 'Jeans', 'Clothing', 500.00),

(7, 'Jacket', 'Clothing', 100.00),

(8, 'Shoes', 'Clothing', 1200.00),

(9, 'Microwave', 'Home', 1500.00),

(10, 'Refrigerator', 'Home', 500.00),

(11, 'Blender', 'Home', 1300.00),

(12, 'Oven', 'Home', 1500.00);

**Table Created:**



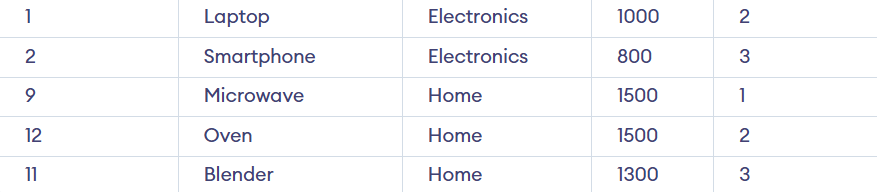
**Now 1st step Question**

**Query:**

SELECT \* FROM (SELECT \*, ROW\_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS RowNum FROM Products) AS Ranked WHERE RowNum <= 3;

**Output:**





**Now 2nd Step Question**

**Query:**

SELECT \* FROM (SELECT \*, RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS RankNum FROM Products) AS Ranked WHERE RankNum <= 3;

**OUTPUT:**





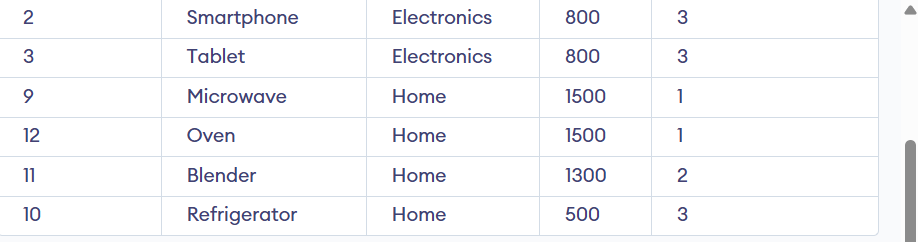
**Now 3rd Step Question**

**Query:**

SELECT \* FROM (SELECT \*, DENSE\_RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS DenseRankNum FROM Products) AS Ranked WHERE DenseRankNum <= 3;

**OUTPUT:**





NAME: Ayushman Upadhyay

College Email: [2230164@kiit.ac.in](mailto:2230164@kiit.ac.in)

SuperSet ID: 6358125