**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

**SQL Queries/Outputs**

1> CREATE TABLE Products (

2> ProductID INT PRIMARY KEY,

3> ProductName VARCHAR(100),

4> Category VARCHAR(50),

5> Price DECIMAL(10,2)

6> );

7> go

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

2> (1, 'Laptop A', 'Laptop', 1500.00),

3> (2, 'Laptop B', 'Laptop', 1200.00),

4> (3, 'Laptop C', 'Laptop', 1500.00),

5> (4, 'Laptop D', 'Laptop', 1000.00),

6> (5, 'Laptop E', 'Laptop', 900.00),

7> (6, 'Desktop A', 'Desktop', 1300.00),

8> (7, 'Desktop B', 'Desktop', 1100.00),

9> (8, 'Desktop C', 'Desktop', 1100.00),

10> (9, 'Desktop D', 'Desktop', 1000.00),

11> (10, 'Tablet A', 'Tablet', 800.00),

12> (11, 'Tablet B', 'Tablet', 750.00),

13> (12, 'Tablet C', 'Tablet', 700.00);

14> go

(12 rows affected)

1> select \* from Products;

2> go

ProductID ProductName Category Price

----------- ---------------------------------------------------------------------------------------------------- -------------------------------------------------- ------------

1 Laptop A Laptop 1500.00

2 Laptop B Laptop 1200.00

3 Laptop C Laptop 1500.00

4 Laptop D Laptop 1000.00

5 Laptop E Laptop 900.00

6 Desktop A Desktop 1300.00

7 Desktop B Desktop 1100.00

8 Desktop C Desktop 1100.00

9 Desktop D Desktop 1000.00

10 Tablet A Tablet 800.00

11 Tablet B Tablet 750.00

12 Tablet C Tablet 700.00

(12 rows affected)

1> SELECT

2> ProductID,

3> ProductName,

4> Category,

5> Price,

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

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

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

9> FROM Products;

10> go

ProductID ProductName Category Price RowNum RankNum DenseRankNum

----------- ---------------------------------------------------------------------------------------------------- -------------------------------------------------- ------------ -------------------- -------------------- --------------------

6 Desktop A Desktop 1300.00 1 1 1

7 Desktop B Desktop 1100.00 2 2 2

8 Desktop C Desktop 1100.00 3 2 2

9 Desktop D Desktop 1000.00 4 4 3

1 Laptop A Laptop 1500.00 1 1 1

3 Laptop C Laptop 1500.00 2 1 1

2 Laptop B Laptop 1200.00 3 3 2

4 Laptop D Laptop 1000.00 4 4 3

5 Laptop E Laptop 900.00 5 5 4

10 Tablet A Tablet 800.00 1 1 1

11 Tablet B Tablet 750.00 2 2 2

12 Tablet C Tablet 700.00 3 3 3

(12 rows affected)

1> WITH RankedProducts AS (

2> SELECT

3> ProductID,

4> ProductName,

5> Category,

6> Price,

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

8> FROM Products

9> )

10> SELECT \*

11> FROM RankedProducts

12> WHERE RowNum <= 3;

13> go

ProductID ProductName Category Price RowNum

----------- ---------------------------------------------------------------------------------------------------- -------------------------------------------------- ------------ --------------------

6 Desktop A Desktop 1300.00 1

7 Desktop B Desktop 1100.00 2

8 Desktop C Desktop 1100.00 3

1 Laptop A Laptop 1500.00 1

3 Laptop C Laptop 1500.00 2

2 Laptop B Laptop 1200.00 3

10 Tablet A Tablet 800.00 1

11 Tablet B Tablet 750.00 2

12 Tablet C Tablet 700.00 3

(9 rows affected)

1>