**Superset id:- 6363303**

**Exercise 1: Ranking and Window Functions**

**Q.1 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.**

**Query for ROW\_NUMBER() :-**

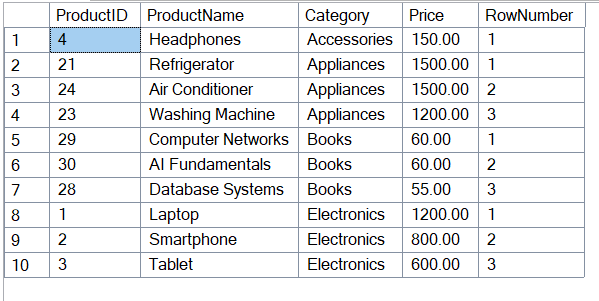
SELECT \* FROM (

SELECT ProductID,ProductName,Category,Price,ROW\_NUMBER() OVER (PARTITION BY Category ORDER BY Price DESC) AS RowNumber

FROM Products

) AS ExpensiveProducts

WHERE RowNumber <= 3;

****

**Query for RANK() :-**

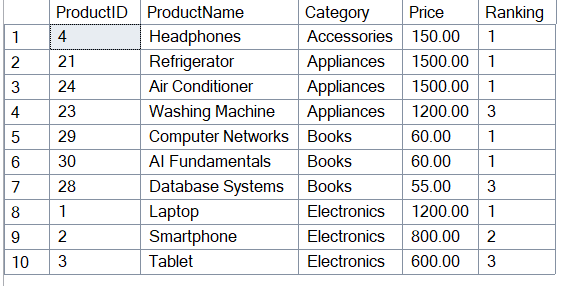
SELECT \* FROM (

SELECT ProductID,ProductName,Category,Price,RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS Ranking

FROM Products

) AS ExpensiveProducts

WHERE Ranking <= 3;



**Query for DENSE\_RANK() :-**

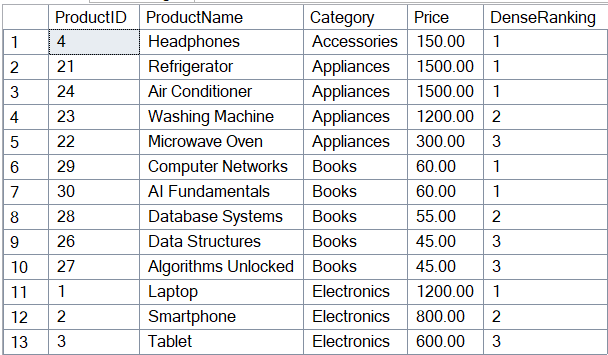
SELECT \* FROM (

SELECT ProductID,ProductName,Category,Price,DENSE\_RANK() OVER (PARTITION BY Category ORDER BY Price DESC) AS DenseRanking

FROM Products

) AS ExpensiveProducts

WHERE DenseRanking <= 3;



**Exercise 2: Index**

**Q.2 Creating a Non-Clustered Index.**

**Before:-**

SELECT \* FROM Products WHERE ProductName = 'Laptop';

**Query for Non-Clustered Index :-**

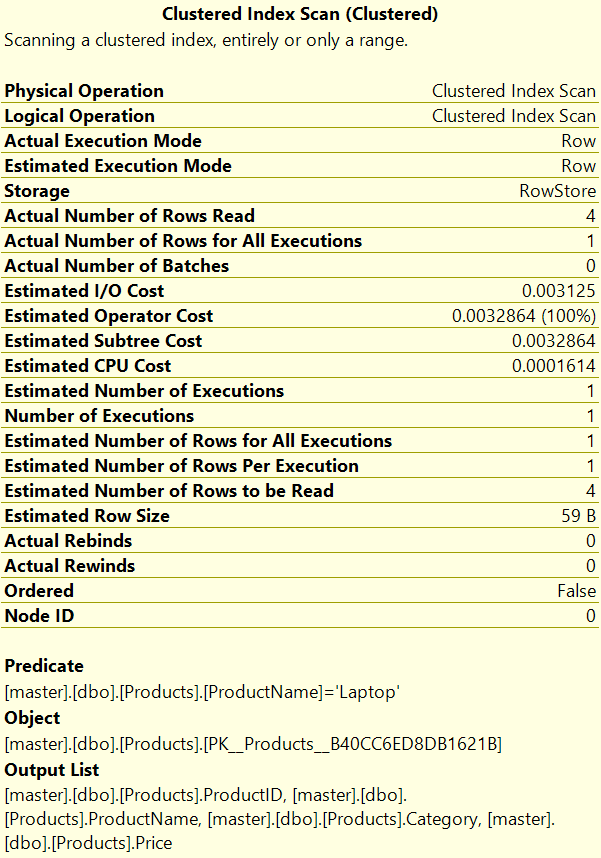
CREATE NONCLUSTERED INDEX IX\_Products\_ProductName

ON Products (ProductName);

**After:-**

SELECT \* FROM Products WHERE ProductName = 'Laptop';





**Q.2 Creating a Clustered Index.**

-- Dropped foreign key and clustered primary key to allow creating a clustered index on OrderDate and then recreated primary key as NONCLUSTERED.

SELECT name

FROM sys.foreign\_keys

WHERE parent\_object\_id = OBJECT\_ID('OrderDetails');

ALTER TABLE OrderDetails

DROP CONSTRAINT FK\_\_OrderDeta\_\_Order\_\_3AA1AEB8;

SELECT name

FROM sys.key\_constraints

WHERE type = 'PK' AND OBJECT\_NAME(parent\_object\_id) = 'Orders';

ALTER TABLE Orders

DROP CONSTRAINT PK\_\_Orders\_\_C3905BAFD4A6A3E5;

**Before:-**

SELECT \* FROM Orders WHERE OrderDate = '2023-01-15';

**Query for Clustered Index :-**

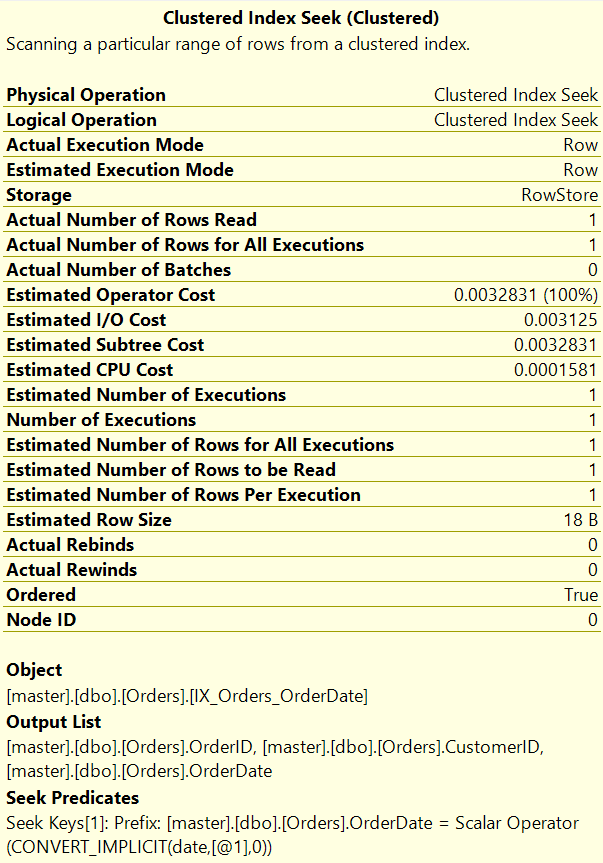
CREATE CLUSTERED INDEX IX\_Orders\_OrderDate

ON Orders (OrderDate);

**After:-**

SELECT \* FROM Orders WHERE OrderDate = '2023-01-15';



****

**Q.3 Creating a Composite Index.**

**Before:-**

SELECT \* FROM Orders WHERE CustomerID = 1 AND OrderDate = '2023-01-15';

**Query for Non-Clustered Index :-**

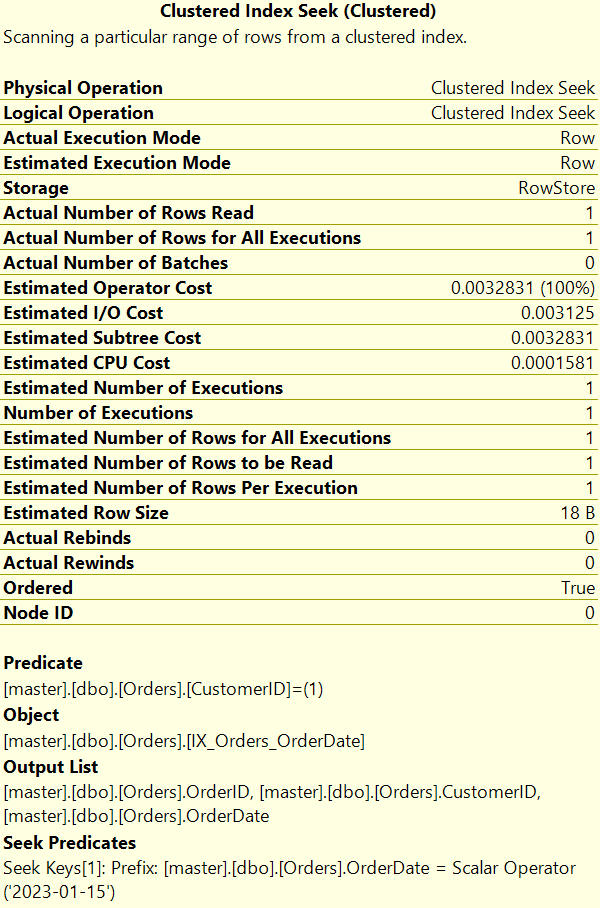
CREATE NONCLUSTERED INDEX IX\_Orders\_CustomerID\_OrderDate

ON Orders (CustomerID, OrderDate);

**After:-**

SELECT \* FROM Orders WHERE CustomerID = 1 AND OrderDate = '2023-01-15';

****

****

**Ayush Kumar (6363303)**