**Task**

1. **Write SQL Queries** to:
   * a. Find the **top 5 most expensive products** in stock.
   * b. Get the **total number of products** added in the last 30 days.
   * c. Show **average price per category**, sorted by highest to lowest.

* **Optimization Check**:
  + Suggest **indexes** you’d add to improve performance.
  + Write a query to **check execution plan** for the first query using SET SHOWPLAN\_ALL.

1. **Bonus (Optional)**:
   * Create a **basic script** to export this data to a .csv file.

**Solutions**:

**1.Find the top 5 most expensive products in stock.**

SELECT TOP 5 ProductID, ProductName, Price FROM Products WHERE InStock > 0 ORDER BY Price DESC;

**2.Get the total number of products added in the last 30 days.**

SELECT COUNT(\*) AS TotalRecentProducts FROM Product WHERE CreatedAt >= DATEADD(DAY, -30, GETDATE());

**3.Show average price per category, sorted by highest to lowest.**

SELECT Category, AVG(Price) AS AvgPrice FROM Products GROUP BY Category ORDER BY AvgPrice DESC;

**4.Suggest indexes you’d add to improve performance.**

For Query 1:

CREATE NONCLUSTERED INDEX idx\_instock\_price ON Products(InStock, Price DESC);

For Query 2:

CREATE NONCLUSTERED INDEX idx\_createdat ON Products(CreatedAt);

For Query 3:

CREATE NONCLUSTERED INDEX idx\_category\_price ON Products(Category, Price);

**5.Write a query to check execution plan for the first query using SET SHOWPLAN\_ALL.**

SET SHOWPLAN\_ALL ON;

-- Query to analyze

SELECT TOP 5 ProductID, ProductName, Price FROM Products WHERE InStock > 0 ORDER BY Price DESC;

SET SHOWPLAN\_ALL OFF;

**6.Create a basic script to export this data to a .csv file.**

bcp "SELECT ProductID, ProductName, Category, Price, InStock, CreatedAt FROM MyDatabase.dbo.Products"

queryout "C:\Exports\products.csv" -c -t, -T -S localhost