INVENTORY MANAGEMENT ANALYSIS

INTRODUCTION

Inventory management is critically important for balancing supply and demand while minimising costs. This project explores key aspects such as stock levels, supplier performance, warehouse utilisation, and reorder strategies. I conducted this project to identify trends, optimise restocking, and assess cost efficiency using SQL queries. Through this project, I have applied various SQL techniques and commands including Joins, Common Table Expressions(CTEs), Aggregations etc to extract meaningful insights from the data.

DATASET OVERVIEW

PRODUCT TABLE

Column Name	Description
Product_ID	Unique identifier for the product
Product_Name	Name of the product
Category	Category to which the product belongs
Stock_Quantity	Quantity of the product available in stock
Stock_Level	Stock level category (e.g., Low, Medium, High)
Status	Current status of the product (e.g., Active, Discontinued)

ORDER TABLE

Column Name	Description
Product_ID	Unique identifier for the product
Order_ID	Unique identifier for the order
Supplier_ID	Unique identifier for the product supplier
Unit_Price	Price per unit of the product
Reorder_Point	Minimum quantity at which reordering is triggered
Lead_Time_Days	Number of days it takes to receive the product after ordering
Last_Restock_Date	Date when the product was last restocked
Min_Order_Quantity	Minimum quantity that must be ordered when placing a restock
Entry_Date	Date the product entry was added to the system

SUPPLIER TABLE

Column Name	Description
Supplier_ID	Unique identifier for the product supplier
Warehouse_Location	Location of the warehouse storing the product
Country	Country where the product is stored or distributed
Latitude	Latitude coordinate of the warehouse location
Longitude	Longitude coordinate of the warehouse location

OBJECTIVE

- To analyze stock and inventory levels, supplier and restocking performance, cost and pricing using the Inventory Management Dataset;
- Apply Entity-Relationship Modelling to visually represent how tables relate in the database;
- Apply Joins, CTEs, Aggregations etc for meaningful insights;
- Improve query readability, optimize performance and explore business strategies.

Order Table (Inventory)

- Product_ID
- 🔑 Order_ID
- Supplier_ID
- Unit_price
- Reorder_Point
- Lead_Time_Days
- Last_Restock_Date
- Min_Order_Quantity
- Entry_Date

Supplier Table (Inventory)

- 🔑 Supplier_ID
- Warehouse_Location
- Country
- Latitude

Product Table (Inventory)

- 🔑 Product_ID
- Product_Name
- Category
- Stock_Quantity
- Stock_Level
- Status

QUERIES AND VISUAL REPRESENTATIONS

```sql
--1. Which product category has the highhest total quantity across all products?

select category, sum(Stock_Quantity) As Total_Stock

from [Inventory].[Product_Table]

group by category

order by Total_Stock Desc
```

	category	Total_Stock
1	Home & Garden	380836
2	Books	373129
3	Office Supplies	367475
4	Sports	360953
5	Toys	352315
6	Clothing	340162
7	Electronics	328396

Product Category	Sum of Stock_Quantity
Home & Garden	380836
Books	373129
Office Supplies	367475
Sports	360953
Toys	352315
Clothing	340162
Electronics	328396

Product Categories and Total stock Quantities

Electronics 328,396
Clothing 340,162
Toys 352,315
Sports 360,953
Office Supplies 367,475
Books 373,129
Home & Garden 380,836

01
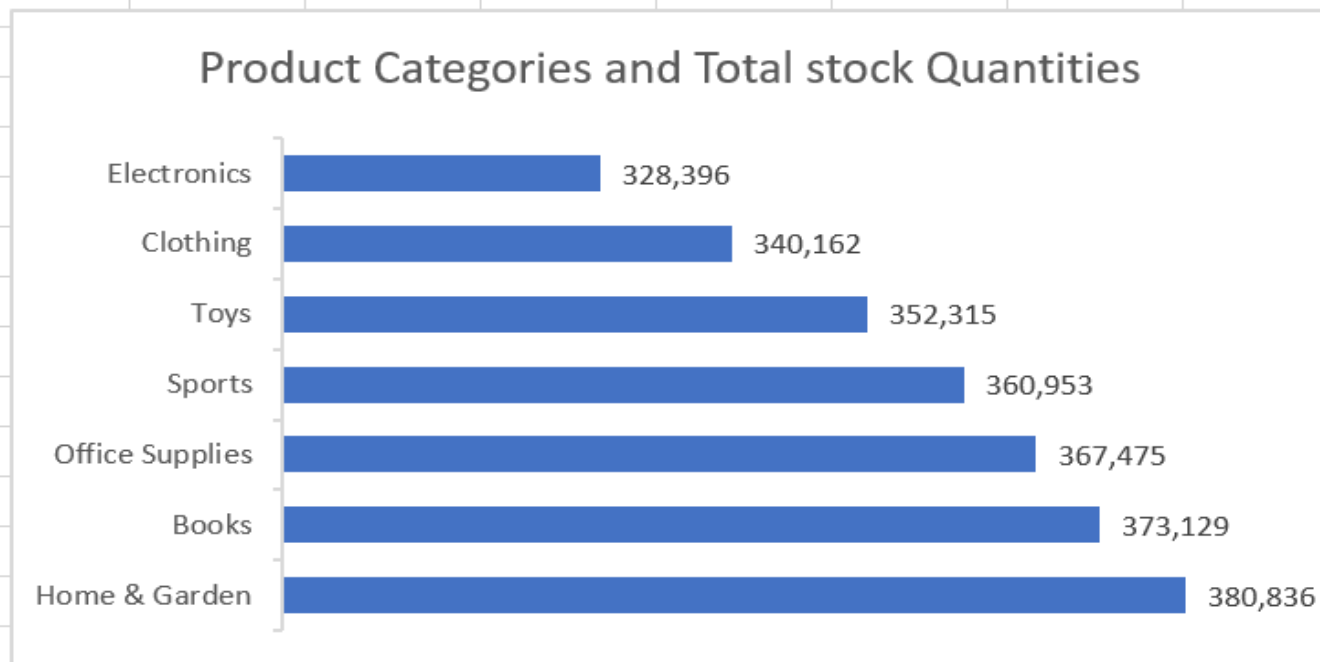
```sql
--2. How many products currently have stock quantities below their reorder point?
   select count(*) as Stock_Quantity_below_reorder_point
   from [Inventory].[Product_Table] p
   join [Inventory].[Order_Table] o
      on p.Product_ID = o.Product_ID
   where p.stock_quantity < o.Reorder_point;
```

Stock_Quantity_below_reorder_point

1	262

Below Re-order Point	Count of Below Re-order Point
NO	4738
YES	262
Grand Total	**5000**

PRODUCTS WITH STOCK QUANTITIES BELOW RE-ORDER POINT

262
4738

- NO
- YES

02
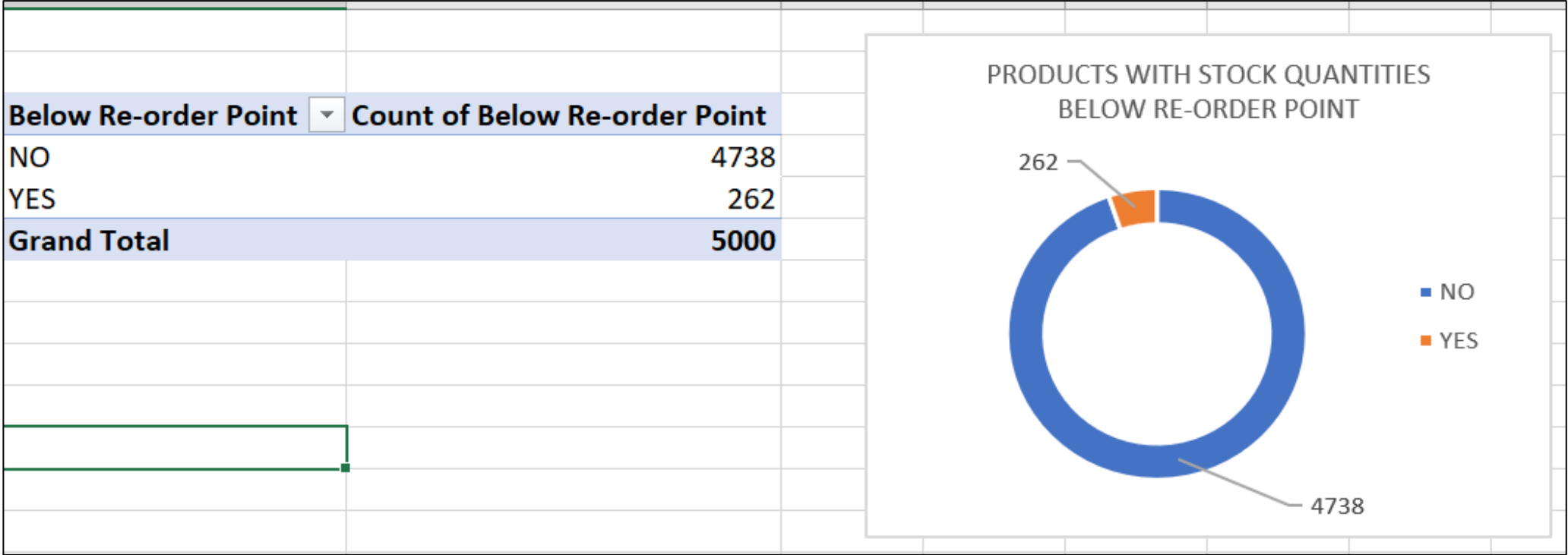
```
--3. What minimum order quantity is required for a company to restock all products below the reorder point?
  select SUM(Min_Order_Quantity) as Total_Min_Order_Quantity_of_products_below_reorder_point
  from [Inventory].[Product_Table] p
  join [Inventory].[Order_Table] o
    on p.Product_ID = o.Product_ID
  where p.stock_quantity < o.Reorder_point;
```
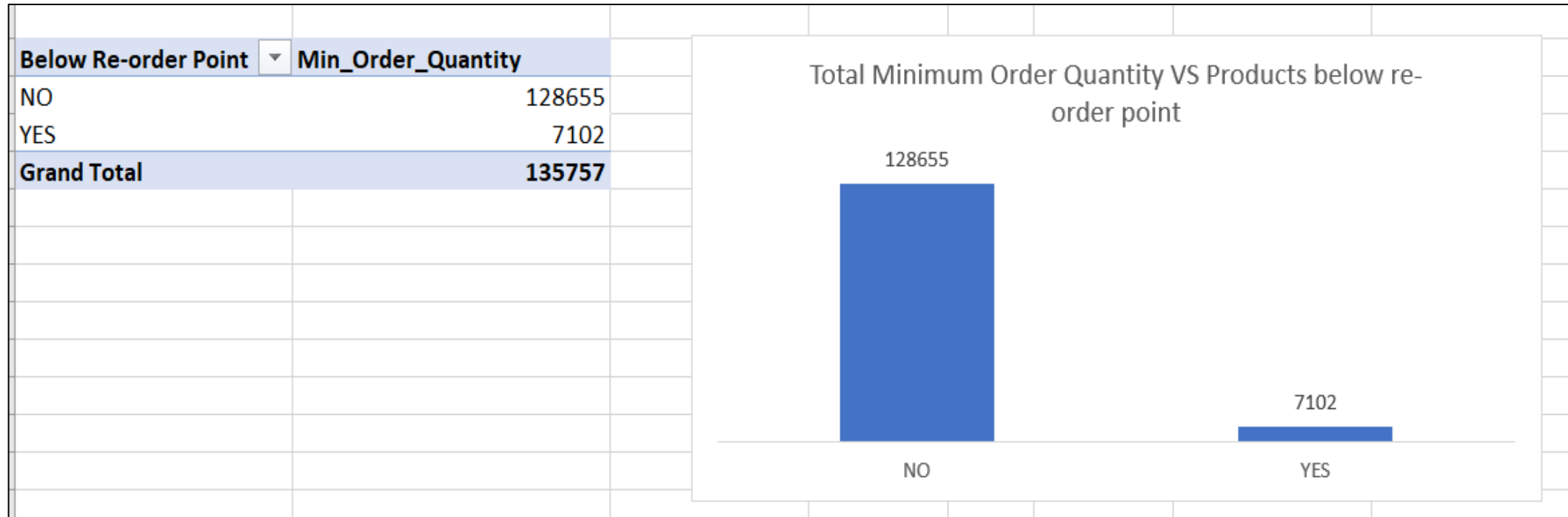
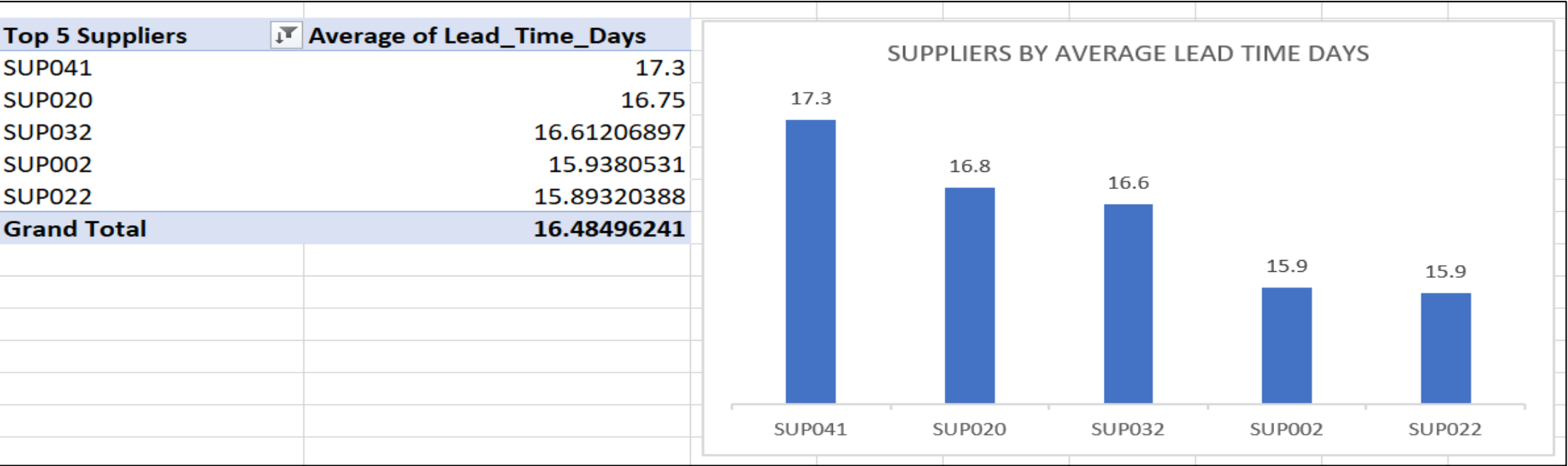|	Total_Min_Order_Quantity_of_products_below_reorder_point
1	7102

Below Re-order Point	Min_Order_Quantity
NO	128655
YES	7102
Grand Total	135757

Total Minimum Order Quantity VS Products below re-order point

03

```sql
--4. Which supplier has the highest average lead time for restocking?
    select Top 1
    Supplier_ID, AVG(Lead_Time_Days) as Supplier_with_highest_average_lead_time_days
    from [Inventory].[Order_Table]
    group by Supplier_ID
    order by  Supplier_with_highest_average_lead_time_days Desc
```

	Supplier_ID	Supplier_with_highest_average_lead_time_days
1	SUP041	17

Top 5 Suppliers	Average of Lead_Time_Days
SUP041	17.3
SUP020	16.75
SUP032	16.61206897
SUP002	15.9380531
SUP022	15.89320388
Grand Total	**16.48496241**

SUPPLIERS BY AVERAGE LEAD TIME DAYS

SUP041	SUP020	SUP032	SUP002	SUP022
17.3	16.8	16.6	15.9	15.9

04

```sql
--5. What is the average time since the last restock for products marked as "Out of Stock"?
    select
    avg(datediff(day, Last_Restock_Date,GETDATE())) as Avg_time_since_last_restock
    from [Inventory].[Product_Table] as p
    join [Inventory].[Order_Table] as o
on p.Product_ID = o.Product_ID
where status = 'Out of Stock'
```
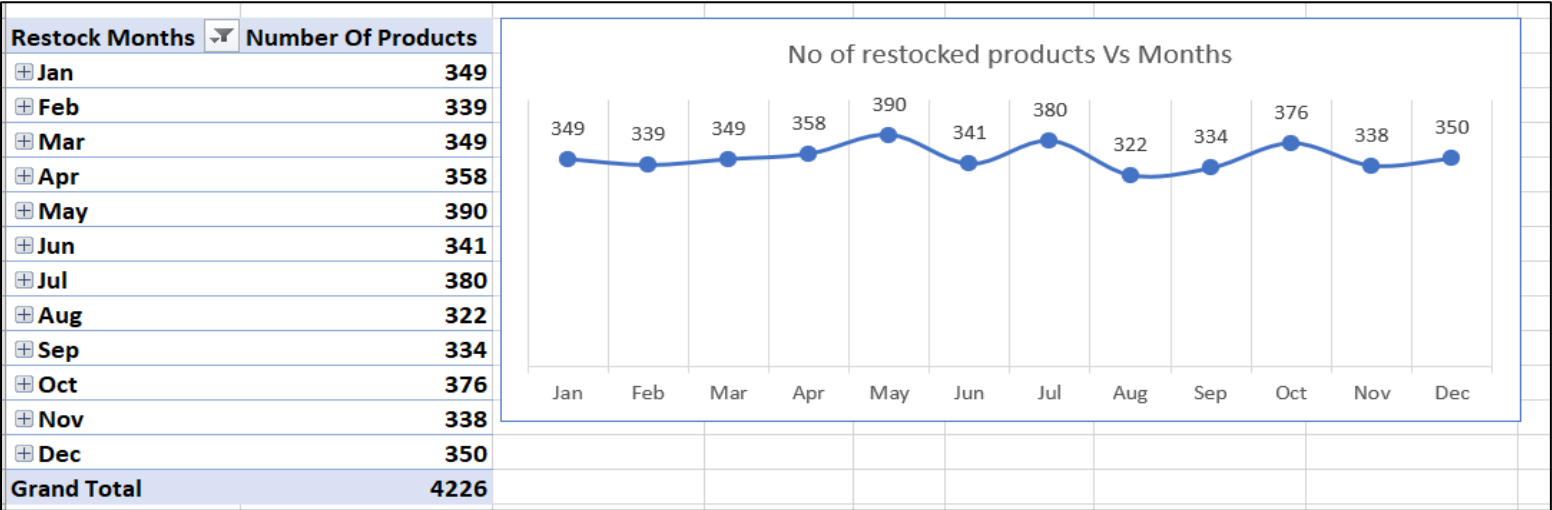
Results | **Messages**

	Avg_time_since_last_restock
1	234

```sql
--6. Identify the months when the highest number of products were restocked--are there any seasonal trends?
    select count(p.Product_ID) as Number_of_Products, DATENAME(month, Last_restock_date) as Highest_Restock_Months
    from [Inventory].[Order_Table] as o
    join [Inventory].[Product_Table] as p
on p.Product_ID = o.Product_ID
where status = 'In Stock'
    Group by DATENAME(month, Last_restock_date)
    order by Number_of_Products Desc
```

Results | **Messages**

	Number_of_Products	Highest_Restock_Months
1	390	May
2	380	July
3	376	October
4	358	April
5	350	December
6	349	January
7	349	March
8	341	June
9	339	February
10	338	November
11	334	September
12	322	August

Restock Months ▼	Number Of Products
⊞ Jan	349
⊞ Feb	339
⊞ Mar	349
⊞ Apr	358
⊞ May	390
⊞ Jun	341
⊞ Jul	380
⊞ Aug	322
⊞ Sep	334
⊞ Oct	376
⊞ Nov	338
⊞ Dec	350
Grand Total	4226

No of restocked products Vs Months

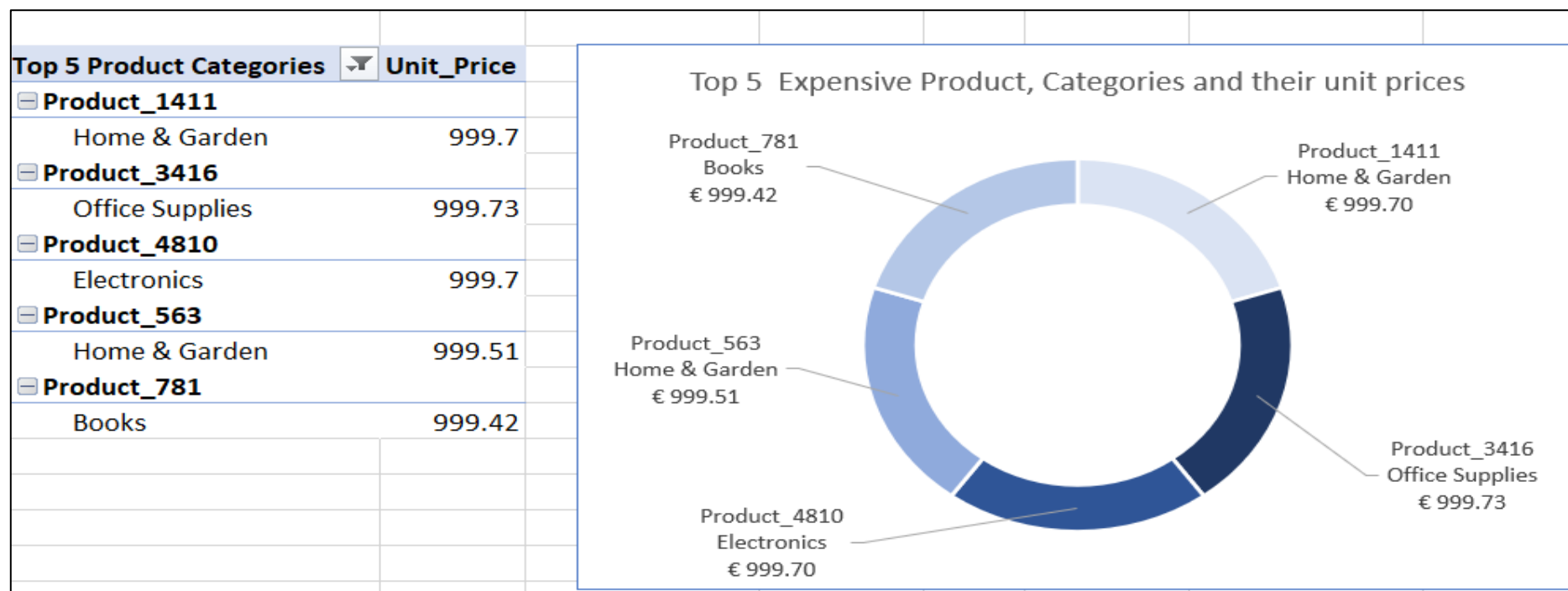| Jan	Feb	Mar	Apr	May	Jun	Jul	Aug	Sep	Oct	Nov	Dec
349	339	349	358	390	341	380	322	334	376	338	350

05 & 06

```
--7. Identify the top 5 most expensive products (by unit price) and their categories.
   select top 5
   Category as Most_Expensive_Products, Max(Unit_Price) as Prices
   from [Inventory].[Order_Table] as o
   join [Inventory].[Product_Table] as p
 on p.Product_ID = o.Product_ID
   group by Category
   order by Prices desc
```

	Most_Expensive_Products	Prices
1	Office Supplies	999.73
2	Electronics	999.7
3	Home & Garden	999.7
4	Books	999.42
5	Clothing	998.94

Top 5 Product Categories	Unit_Price
Product_1411	
Home & Garden	999.7
Product_3416	
Office Supplies	999.73
Product_4810	
Electronics	999.7
Product_563	
Home & Garden	999.51
Product_781	
Books	999.42

Top 5 Expensive Product, Categories and their unit prices

Product_781 Books € 999.42
Product_1411 Home & Garden € 999.70
Product_563 Home & Garden € 999.51
Product_3416 Office Supplies € 999.73
Product_4810 Electronics € 999.70

07

INSIGHTS GENERATED

- The Home & Garden category holds the highest total stock across all product categories.

- 262 products currently have stock quantities below their reorder point showing a potential restocking need.

- With SUP041 being the supplier with the highest average lead time of 17.3 days for restocking followed closely by SUP020 with 16.7 days, I went further to review several suppliers showing lead times ranging from 15 to 17 days.

- The average time since the last restock for products marked Out of Stock is about 234 days a key indicator of delay.

- When assessing restock frequency, May had the most restocked products, indicating seasonal trends.

- I also found the top five most expensive products; Products 1411 and 563 which falls under the Home & Garden category followed by Products 3416, 4810 and 781 under the Office Supplies, Electronics and Books categories respectively.
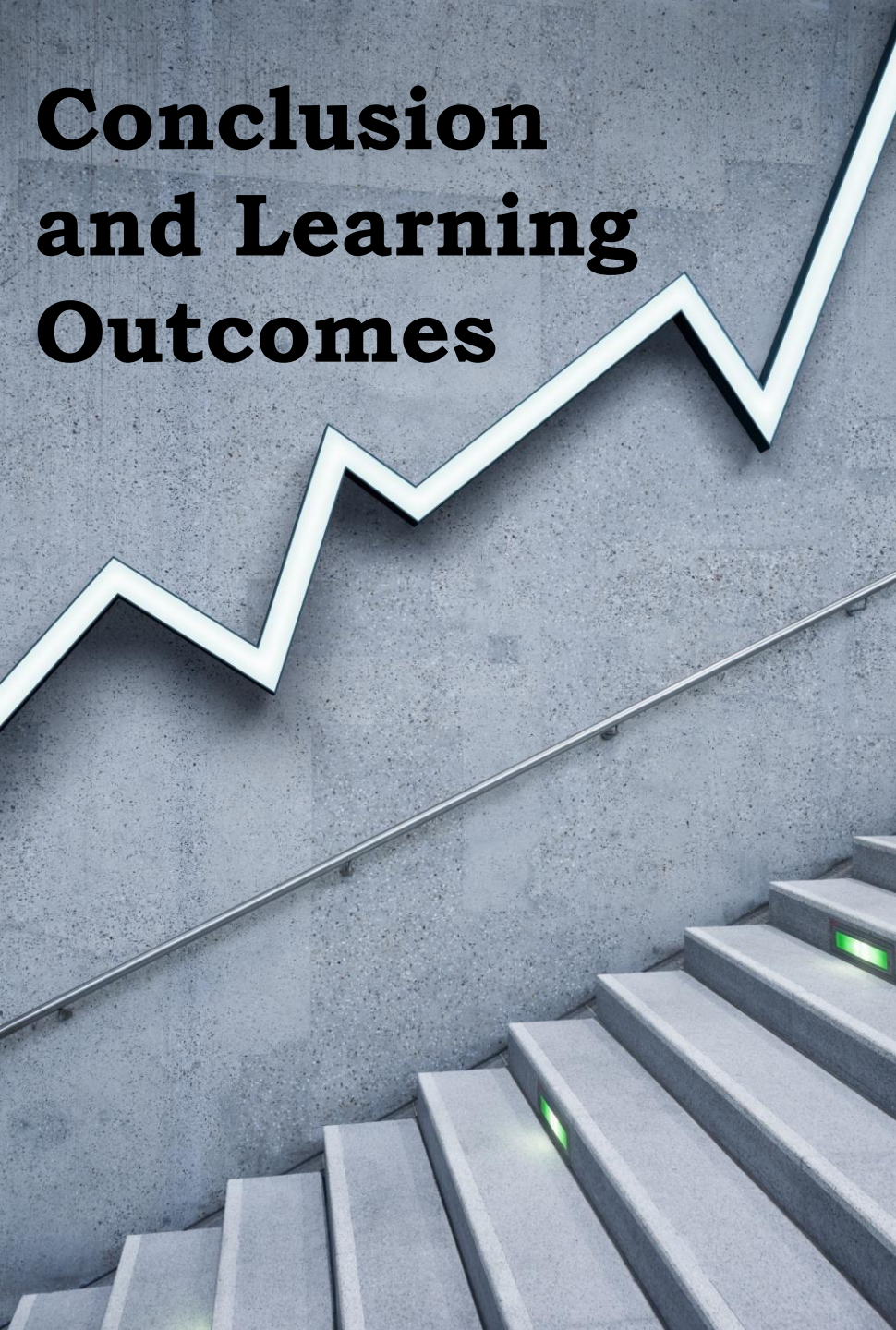
Conclusion and Learning Outcomes

At the end of the project I was able to:

- Learn how to create a database schema design(data modelling)

- Learn how to properly structure queries by using CTEs

- Explore the efficient use of joins and aggregations