

E-Commerce Sales Analysis using SQL & Power BI

Project Overview

This project aims to analyze e-commerce sales data using **SQL** for data extraction, transformation, and insight generation. The dataset used is the **Superstore dataset** (<https://www.kaggle.com/datasets/vivek468/superstore-dataset-final/code>), which contains information on orders, sales, profits, customers, shipping, and more. Additionally, a **Power BI dashboard** was created to visualize key findings interactively.

Dataset Overview

- **Dataset Name:** Superstore Dataset
- **Source:** Kaggle/Public Datasets
- **Number of Records:** 10,000+
- **Fields Included:** Order ID, Product Name, Sales, Profit, Customer Segment, Region, Discount, Order Date, Ship Date, Ship Mode, etc.

Tools Used

- **SQL (MySQL/PostgreSQL)** – Data extraction and analysis
- **Power BI** – Interactive dashboard visualization

1. Prerequisites

You have to create a database before uploading the dataset in MySQL.

Database creating query:

```
CREATE database superstore_sales;
```

You have to select the database to operate in it.

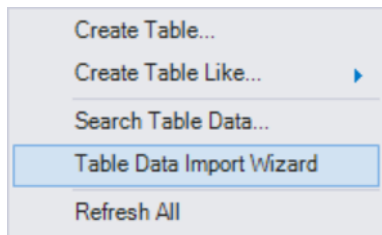
Database selection query:

```
USE superstore_sales;
```

Open the .csv file and create a table matching the columns present in the file.

```
CREATE TABLE superstore(  
  
    Row_ID INT,  
  
    Order_ID VARCHAR(50),  
  
    Order_Date DATE,  
  
    Ship_Date DATE,  
  
    Ship_Mode VARCHAR(50),  
  
    Customer_ID VARCHAR(50),  
  
    Customer_Name VARCHAR(100),  
  
    Segment VARCHAR(50),  
  
    Country VARCHAR(50),  
  
    City VARCHAR(50),  
  
    State VARCHAR(50),  
  
    Postal_Code VARCHAR(20),  
  
    Region VARCHAR(50),  
  
    Product_ID VARCHAR(50),  
  
    Category VARCHAR(50),  
  
    Sub_Category VARCHAR(50),  
  
    Product_Name VARCHAR(255),  
  
    Sales DECIMAL(10,2),  
  
    Quantity INT,  
  
    Discount DECIMAL(5,2),  
  
    Profit DECIMAL(10,2)  
  
);
```

Using Table data import Wizard load the information from .csv file to the created table.



2. Check if the data is loaded correctly

- a. Checking the no of entries.

```
1 • SELECT COUNT(*) from superstore;
2
3
```

Result Grid

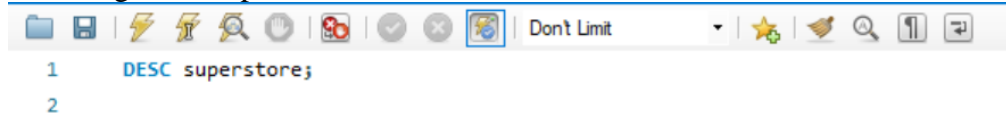
	COUNT(*)
▶	9994

- b. Having a look at the preview of the table

```
1 • SELECT * from superstore limit 5;
2
```

Row_ID	Order_ID	Order_Date	Ship_Date	Ship_Mode	Customer_ID	Customer_Name	Segment	Country	City	State	Postal_Code
▶ 1	CA-2016-152156	2016-11-08	2016-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	Kentucky	42420
2	CA-2016-152156	2016-11-08	2016-11-11	Second Class	CG-12520	Claire Gute	Consumer	United States	Henderson	Kentucky	42420
3	CA-2016-138688	2016-06-12	2016-06-16	Second Class	DV-13045	Darrin Van Huff	Corporate	United States	Los Angeles	California	90036
4	US-2015-108966	2015-10-11	2015-10-18	Standard Class	SO-20335	Sean O Donnell	Consumer	United States	Fort Lauderdale	Florida	33311
5	US-2015-108966	2015-10-11	2015-10-18	Standard Class	SO-20335	Sean O Donnell	Consumer	United States	Fort Lauderdale	Florida	33311

c. Checking the composition of the table

A screenshot of a database tool's 'Result Grid' showing the schema of the 'superstore' table. The table has 17 columns: Row_ID, Order_ID, Order_Date, Ship_Date, Ship_Mode, Customer_ID, Customer_Name, Segment, Country, City, State, Postal_Code, Region, Product_ID, Category, Sub_Category, and Product_Name. The columns are organized into a grid with headers: Field, Type, Null, Key, Default, and Extra. The 'Null' column for all fields is 'YES'. The 'Default' column for all fields is 'NULL'. The 'Extra' column is empty for all fields. The 'Key' column is empty for all fields. The 'Field' column lists the column names. The 'Type' column lists the data types: int, varchar(50), date, and varchar(255).

Field	Type	Null	Key	Default	Extra
Row_ID	int	YES		NULL	
Order_ID	varchar(50)	YES		NULL	
Order_Date	date	YES		NULL	
Ship_Date	date	YES		NULL	
Ship_Mode	varchar(50)	YES		NULL	
Customer_ID	varchar(50)	YES		NULL	
Customer_Name	varchar(100)	YES		NULL	
Segment	varchar(50)	YES		NULL	
Country	varchar(50)	YES		NULL	
City	varchar(50)	YES		NULL	
State	varchar(50)	YES		NULL	
Postal_Code	varchar(20)	YES		NULL	
Region	varchar(50)	YES		NULL	
Product_ID	varchar(50)	YES		NULL	
Category	varchar(50)	YES		NULL	
Sub_Category	varchar(50)	YES		NULL	
Product_Name	varchar(255)	YES		NULL	

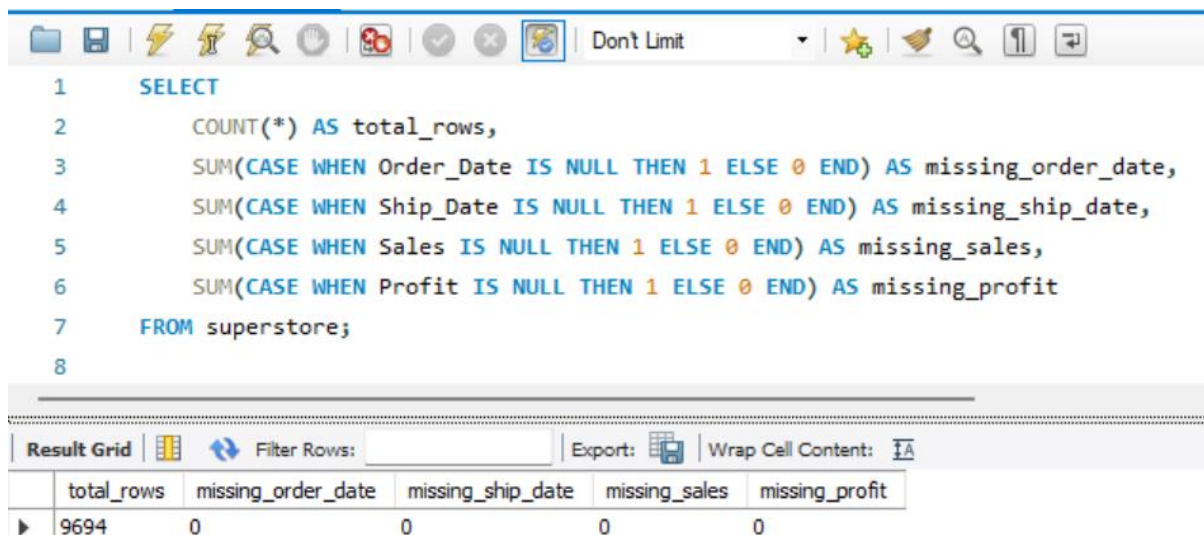
3. Data Cleaning & Preprocessing in SQL

Before analysis, the dataset was cleaned to ensure accurate insights. The following steps were performed:

- Removed duplicate records
- Handled missing values
- Standardized date formats
- Ensured data type consistency (e.g., numeric fields for sales and profit)

Data Cleaning Queries

Checking if there are any NULL values:



The screenshot shows a SQL query editor with a toolbar at the top. The query is as follows:

```

1  SELECT
2      COUNT(*) AS total_rows,
3      SUM(CASE WHEN Order_Date IS NULL THEN 1 ELSE 0 END) AS missing_order_date,
4      SUM(CASE WHEN Ship_Date IS NULL THEN 1 ELSE 0 END) AS missing_ship_date,
5      SUM(CASE WHEN Sales IS NULL THEN 1 ELSE 0 END) AS missing_sales,
6      SUM(CASE WHEN Profit IS NULL THEN 1 ELSE 0 END) AS missing_profit
7  FROM superstore;
8

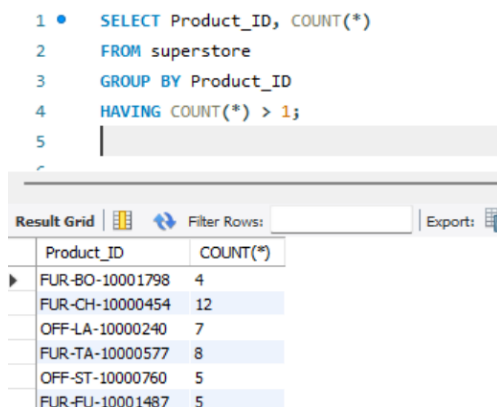
```

Below the query, the result grid is displayed with the following data:

	total_rows	missing_order_date	missing_ship_date	missing_sales	missing_profit
▶	9694	0	0	0	0

We have observed no missing or null values in the table.

Checking if there are any DUPLICATE values:



The screenshot shows a SQL query editor with the following query:

```

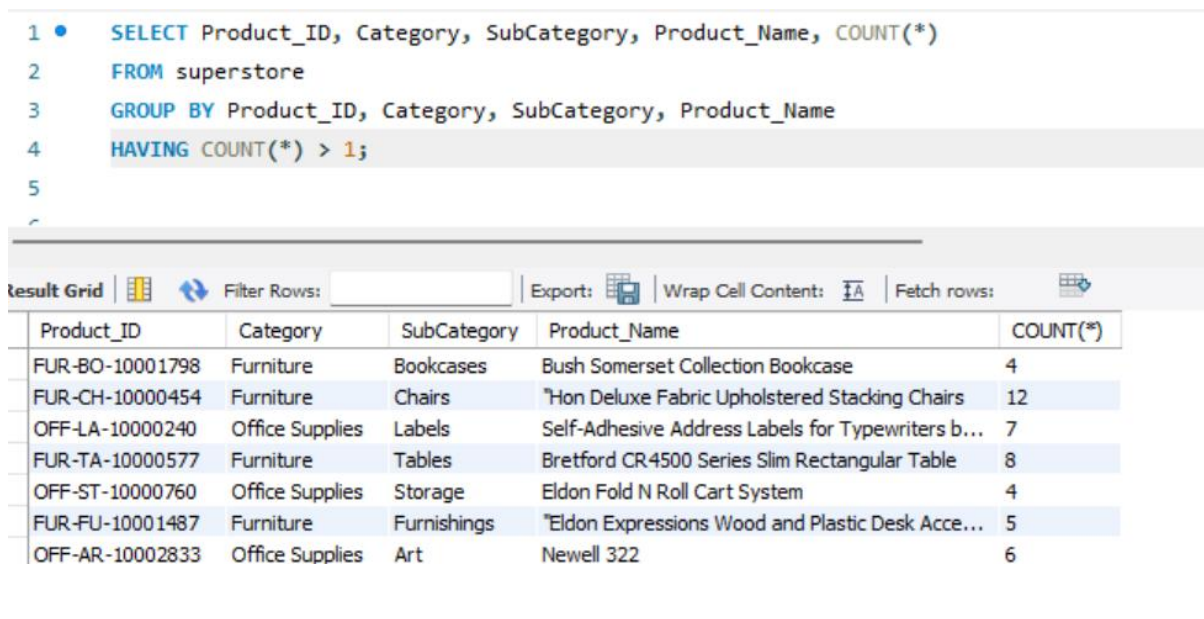
1 • SELECT Product_ID, COUNT(*)
2 FROM superstore
3 GROUP BY Product_ID
4 HAVING COUNT(*) > 1;
5

```

The result grid shows the following data:

	Product_ID	COUNT(*)
▶	FUR-BO-10001798	4
	FUR-CH-10000454	12
	OFF-LA-10000240	7
	FUR-TA-10000577	8
	OFF-ST-10000760	5
	FUR-FU-10001487	5

Identifying the reason behind the duplicates:



The screenshot shows a SQL query editor with the following query:

```

1 • SELECT Product_ID, Category, SubCategory, Product_Name, COUNT(*)
2 FROM superstore
3 GROUP BY Product_ID, Category, SubCategory, Product_Name
4 HAVING COUNT(*) > 1;
5

```

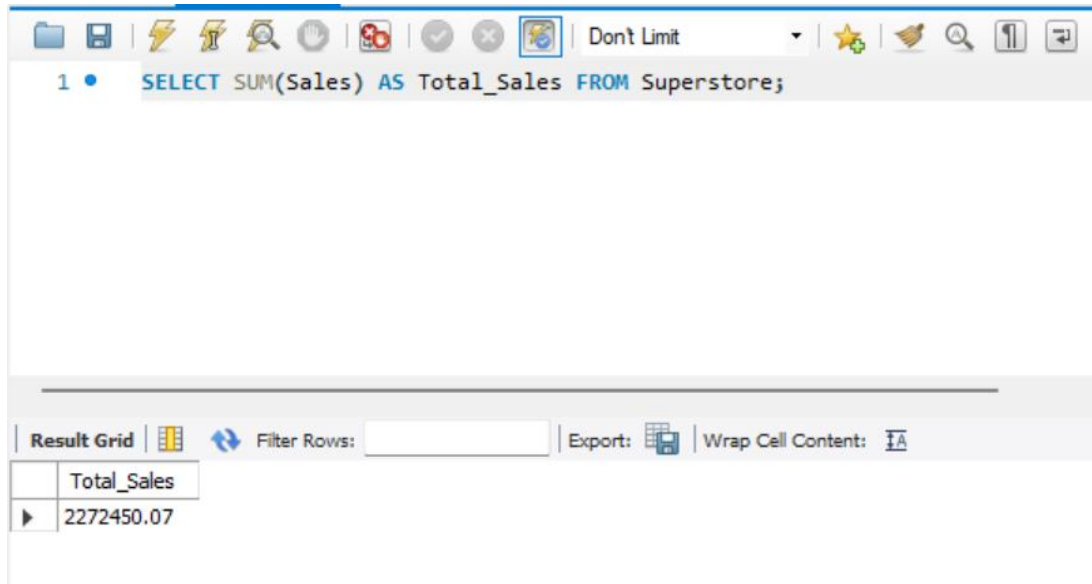
The result grid shows the following data:

	Product_ID	Category	SubCategory	Product_Name	COUNT(*)
	FUR-BO-10001798	Furniture	Bookcases	Bush Somerset Collection Bookcase	4
	FUR-CH-10000454	Furniture	Chairs	"Hon Deluxe Fabric Upholstered Stacking Chairs	12
	OFF-LA-10000240	Office Supplies	Labels	Self-Adhesive Address Labels for Typewriters b...	7
	FUR-TA-10000577	Furniture	Tables	Bretford CR4500 Series Slim Rectangular Table	8
	OFF-ST-10000760	Office Supplies	Storage	Eldon Fold N Roll Cart System	4
	FUR-FU-10001487	Furniture	Furnishings	"Eldon Expressions Wood and Plastic Desk Acce...	5
	OFF-AR-10002833	Office Supplies	Art	Newell 322	6

4. Sales Performance Analysis in SQL

Key Questions & Queries

Q1: What is the total sales revenue generated by the company?



The screenshot shows a SQL query editor interface. The query text is: `SELECT SUM(Sales) AS Total_Sales FROM Superstore;`. Below the query editor, there is a 'Result Grid' section. It contains a table with one column, 'Total_Sales', and one row with the value '2272450.07'. The interface includes various icons for file operations, execution, and formatting.

Total_Sales
2272450.07

Total sales generated by the company is 2272450.07

Q2: How have sales changed over time (monthly/annually)?

```

1 SELECT YEAR(Order_Date) AS Year, MONTH(Order_Date) AS Month, SUM(Sales) AS Monthly_Sales
2 FROM Superstore
3 GROUP BY YEAR(Order_Date), MONTH(Order_Date)
4 ORDER BY Year, Month;
5

```

Result Grid Filter Rows: Export: Wrap Cell Content:

Year	Month	Monthly_Sales
2014	2	4119.84
2014	3	55526.23
2014	4	28139.56
2014	5	23634.66
2014	6	34509.01
2014	7	33500.85
2014	8	27603.51
2014	9	81496.79
2014	10	31394.91
2014	11	78297.26
2014	12	69379.86
2015	1	18085.12
2015	2	11924.26

Q3: Which product category contributes the most to sales?

```

1 • SELECT Category, SUM(Sales) AS Total_Sales
2 FROM Superstore
3 GROUP BY Category
4 ORDER BY Total_Sales DESC;
5

```

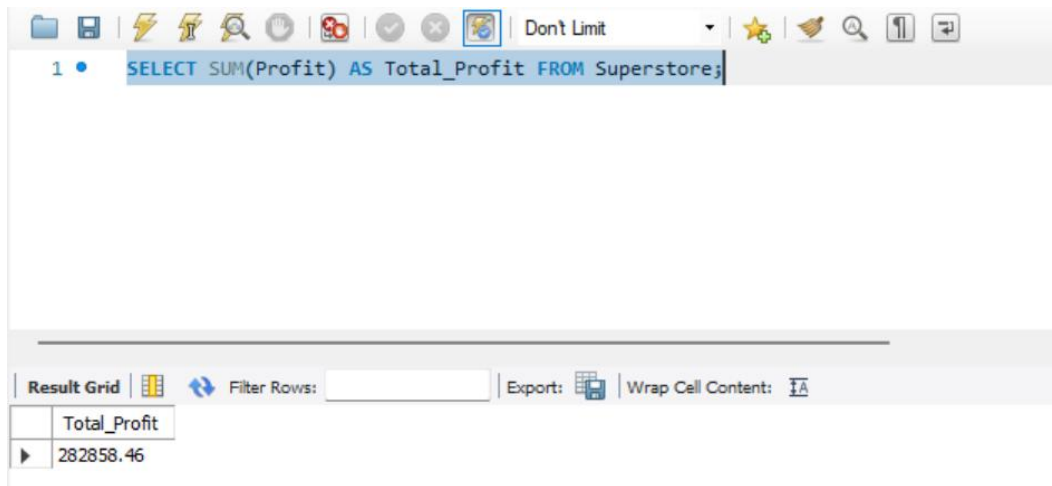
Result Grid Filter Rows: Export: Wrap Cell Content:

Category	Total_Sales
Technology	835900.14
Furniture	733047.06
Office Supplies	703502.87

Technology section contributes most to the sales.

5. Profitability Insights

Q4: What is the total profit earned by the company?



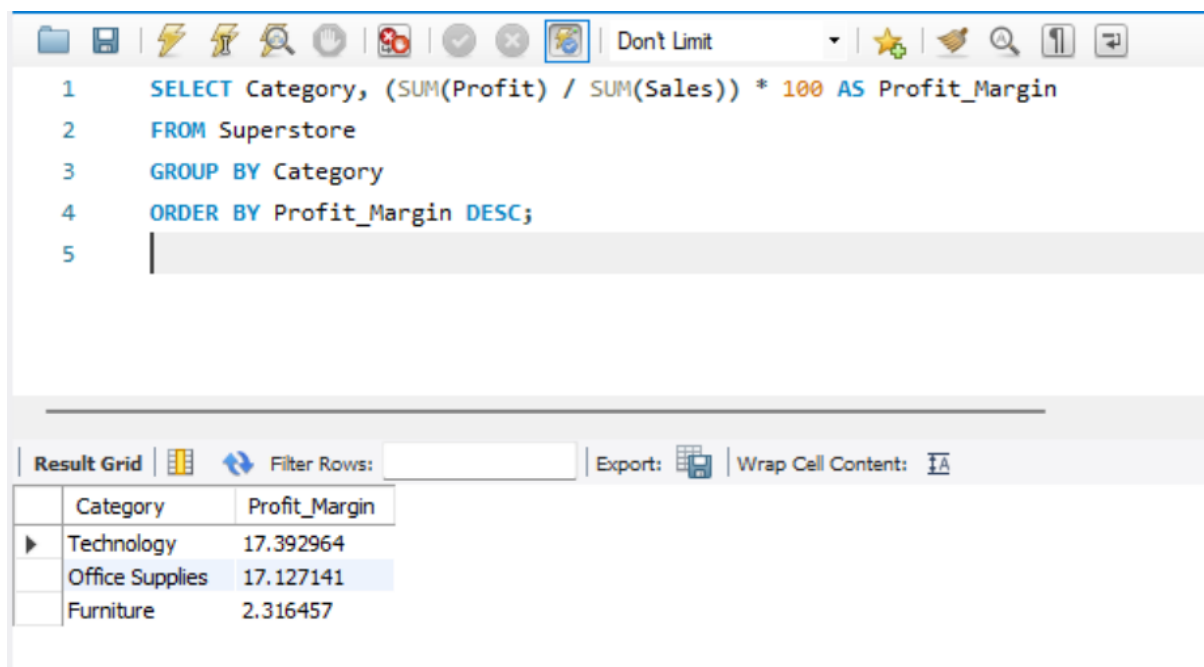
1 • `SELECT SUM(Profit) AS Total_Profit FROM Superstore;`

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	Total_Profit
▶	282858.46

*Total Profit generated by the company is **282858.46***

Q5: Which product category has the highest profit margin?



1 `SELECT Category, (SUM(Profit) / SUM(Sales)) * 100 AS Profit_Margin`
 2 `FROM Superstore`
 3 `GROUP BY Category`
 4 `ORDER BY Profit_Margin DESC;`
 5

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

	Category	Profit_Margin
▶	Technology	17.392964
	Office Supplies	17.127141
	Furniture	2.316457

***Technology** category has the highest profit margin*

Q6: Which region is the most profitable?


```
1 • SELECT Region, SUM(Profit) AS Total_Profit
2 FROM Superstore
3 GROUP BY Region
4 ORDER BY Total_Profit DESC;
5
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
	Region	Total_Profit	
▶	West	106021.47	
	East	90672.04	
	South	46036.00	
	Central	40128.95	

West section has generated the most profit.

6. Customer Insights

Q7: Who are the top 10 customers by total sales?

1	•	SELECT Customer_Name, SUM(Sales) AS Total_Spent
2		FROM Superstore
3		GROUP BY Customer_Name
4		ORDER BY Total_Spent DESC
5		LIMIT 10;

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Customer_Name	Total_Spent		
Sean Miller	25043.07		
Tamara Chand	19017.85		
Raymond Buch	15117.35		
Tom Ashbrook	14595.62		
Adrian Barton	14355.61		
Sanjit Chand	14142.34		
Ken Lonsdale	14071.92		
Hunter Lopez	12873.30		
Sanjit Engle	12209.44		
Christopher Conant	12129.08		

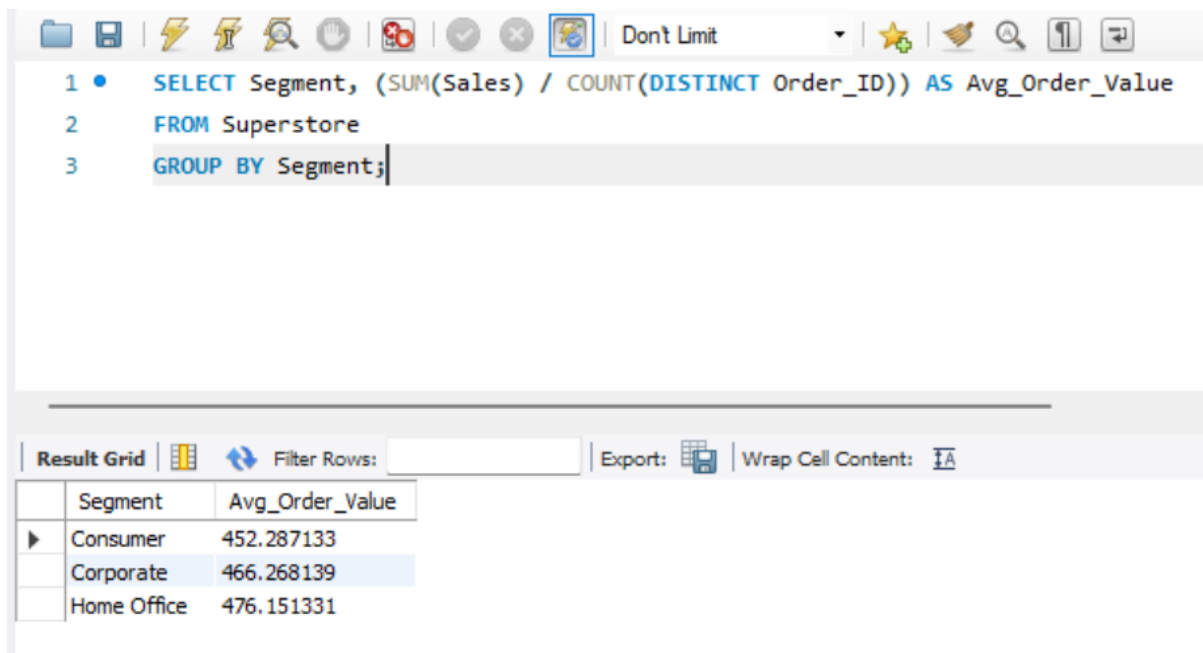
Q8: Which customer segment contributes the most to sales?

1	•	SELECT Segment, SUM(Sales) AS Total_Sales
2		FROM Superstore
3		GROUP BY Segment
4		ORDER BY Total_Sales DESC;

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Segment	Total_Sales		
Consumer	1150166.18		
Corporate	696604.60		
Home Office	425679.29		

Consumer section contributes most to the database.

Q9: What is the average order value (AOV) for each segment?



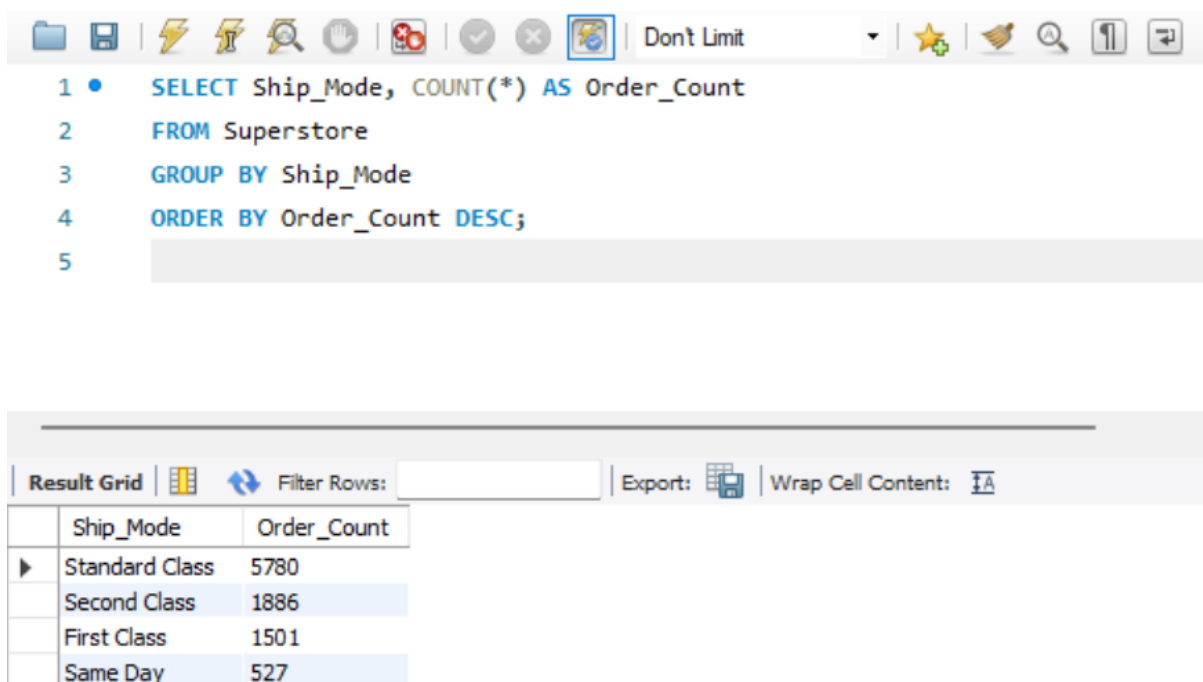
```
1 • SELECT Segment, (SUM(Sales) / COUNT(DISTINCT Order_ID)) AS Avg_Order_Value
2 FROM Superstore
3 GROUP BY Segment;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [fA](#)

	Segment	Avg_Order_Value
▶	Consumer	452.287133
	Corporate	466.268139
	Home Office	476.151331

7. Shipping & Order Trends

Q10: What are the most common shipping methods used?



```
1 • SELECT Ship_Mode, COUNT(*) AS Order_Count
2 FROM Superstore
3 GROUP BY Ship_Mode
4 ORDER BY Order_Count DESC;
5
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [fA](#)

	Ship_Mode	Order_Count
▶	Standard Class	5780
	Second Class	1886
	First Class	1501
	Same Day	527

Standard Class is the most common form of shipping method

Q11: Which shipping mode is the fastest (based on average shipping time)?

```
1 • SELECT Ship_Mode, AVG(DATEDIFF(Ship_Date, Order_Date)) AS Avg_Shipping_Time
2 FROM superstore
3 GROUP BY Ship_Mode
4 ORDER BY Avg_Shipping_Time;
5
6
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Ship_Mode	Avg_Shipping_Time		
Same Day	0.0442		
First Class	2.1827		
Second Class	3.2380		
Standard Class	5.0065		

***SAME DAY** method is the fastest mode of shipping*

Q12: Are there any delays in shipping?

```
1 • SELECT COUNT(*) AS Delayed_Orders
2 FROM Superstore
3 WHERE Ship_Date > Order_Date;
4
```

Result Grid	Filter Rows:	Export:	Wrap Cell Content:
Delayed_Orders			
9191			

*There are **9191** delayed orders.*

Power BI Dashboard Design

After extracting insights from SQL, a **Power BI dashboard** was designed to visualize key findings.

KPIs Included:

- **Total Sales**
- **Total Profit**
- **Top 10 Customers by Sales**
- **Sales Trend Over Time**
- **Regional Sales & Profit Breakdown**
- **Most Profitable Product Categories**
- **Shipping Mode Performance**

Dashboard Features:

- ☐ Interactive filters for **date range, region, and category**
 - ☐ Visual breakdown of **profitability by product segment**
 - ☐ Drill-down features to explore **customer purchase behavior**
-

Conclusion & Final Insights

- **Profitability Strategy:** Certain product categories have high sales but low profit margins. Optimizing pricing and discounts can improve revenue.
- **Shipping Efficiency:** Faster shipping methods need to be prioritized to reduce delivery delays.
- **Customer Segmentation:** High-value customers should be targeted with loyalty programs to increase sales.