Sales Analytics Using SQL

SQL Query Project Documentation

Prepared by: Aparna Sharmaa
Role: Data Scientist/Data Analyst (Fresher)
Date: January 2025

Project Overview:

This project focuses on analyzing retail sales data using SQL to extract business insights. The dataset includes details such as Invoice Number, Product Details, Quantity, Price, Customer Information, and Country.

Key skills demonstrated in this project:

- ☐ Data cleaning and preprocessing
- ☐ Writing SQL queries for descriptive analytics
- ☐ Using aggregation, window functions, and conditional logic
- ☐ Preparing data insights for visualization tools like Power BI

Dataset Summary

Column Name	Description
InvoiceNo	Unique transaction ID
StockCode	Product code
Description	Product name
Quantity	Number of units purchased
InvoiceDate	Date and time of transaction
UnitPrice	Price per product unit
CustomerID	Unique customer ID
Country	Customer's country

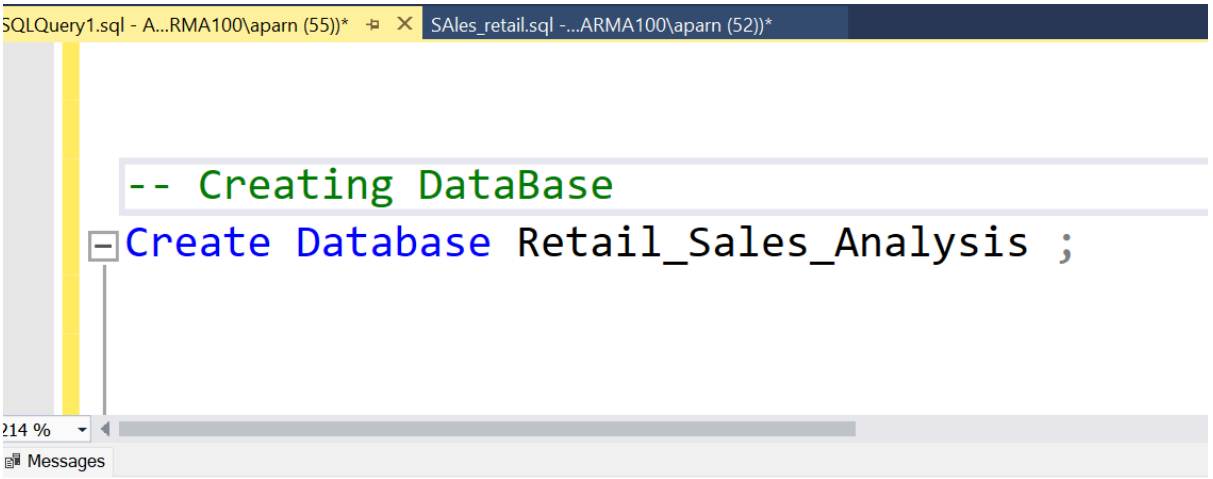
✅ Tools Used

Microsoft SQL Server

Power BI

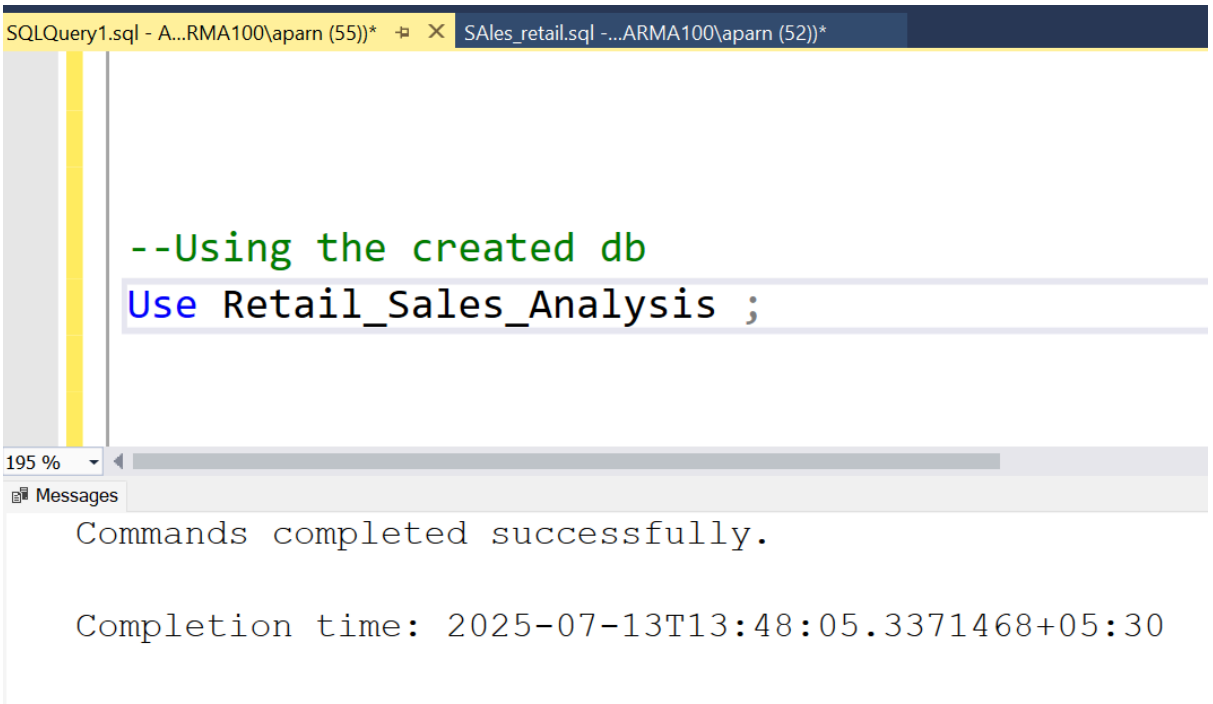
Create Database:

Create Database Retail_Sales_Analysis ;



To use the database:

Use Retail_Sales_Analysis ;



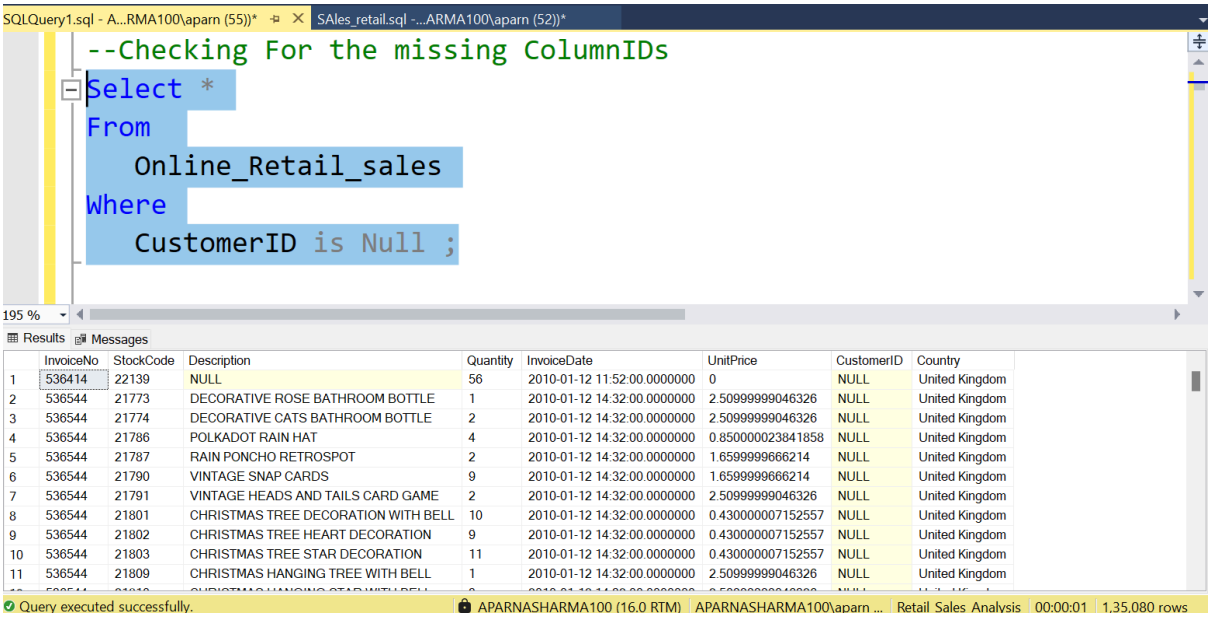
Data Cleaning Steps

Before performing sales analysis, the following SQL data cleaning steps were applied to ensure accuracy and reliability:

1. Remove Rows with Missing Customer IDs

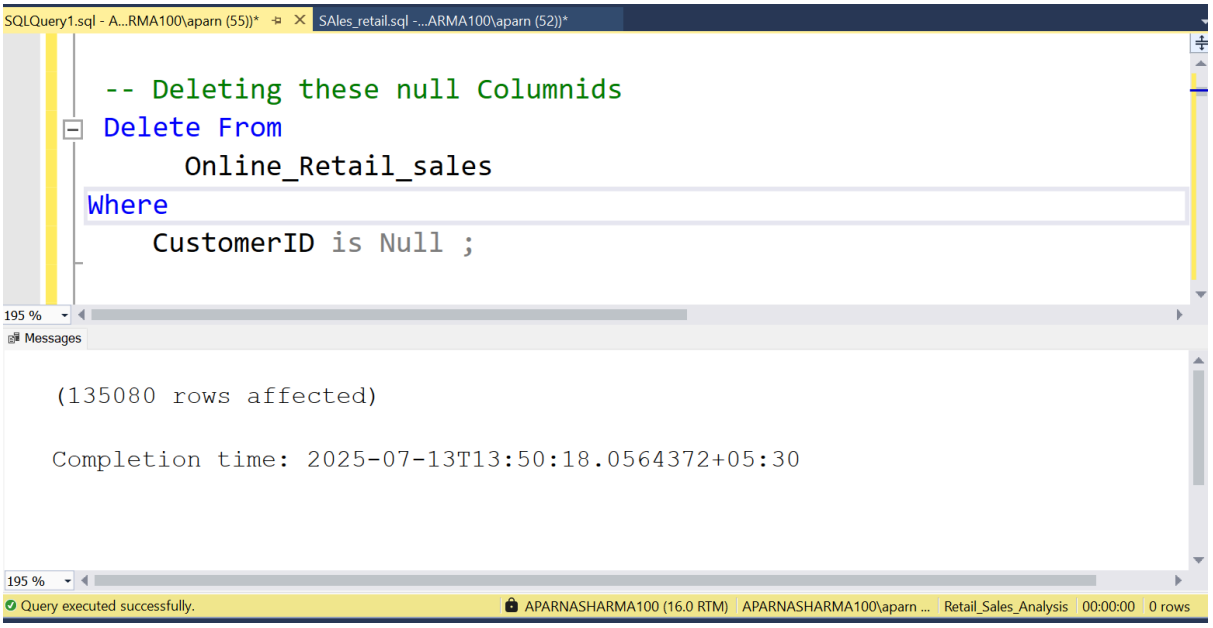
Firstly I checked for the rows with missing customerIds using :

```
Select *
From
  Online_Retail_sales
Where
  CustomerID is Null ;
```



Rows without customer IDs were considered incomplete and removed.

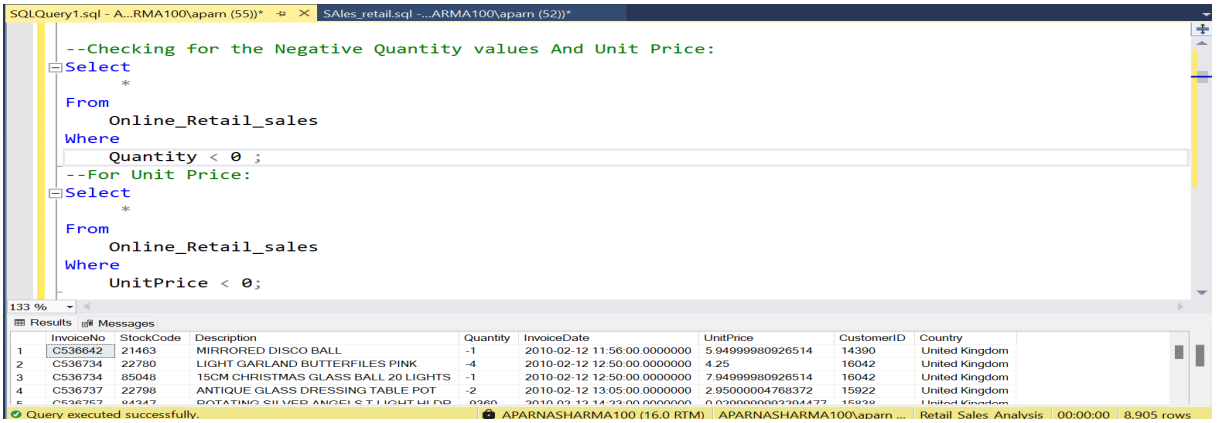
```
Delete From
     Online_Retail_sales
Where
   CustomerID is Null ;
```



2. Handle Negative Quantity Values

Rows with negative quantity and unit price were identified .

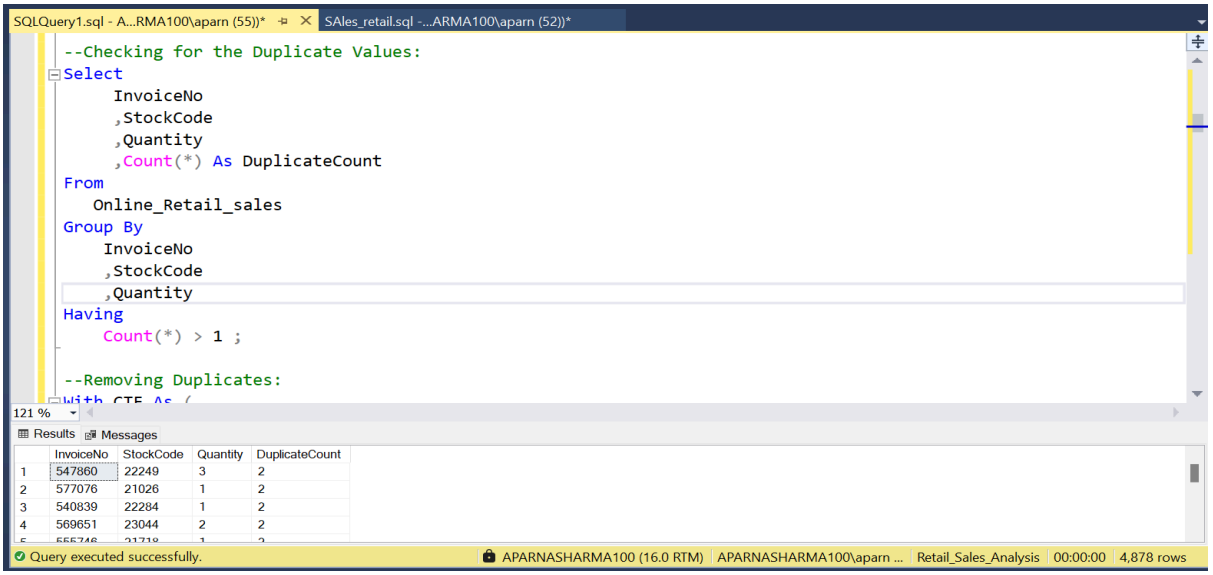
```
Select  *
From
   Online_Retail_sales
Where
   Quantity < 0 ;

--For Unit Price:
Select  *
From
   Online_Retail_sales
Where
   UnitPrice < 0;
```



3.Remove Duplicate Records

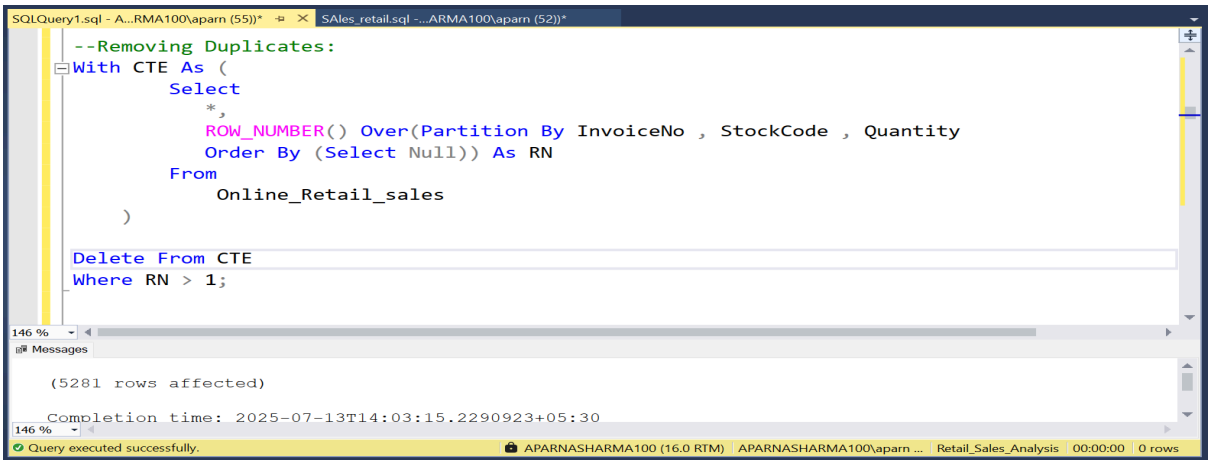
Duplicate records were identified based on InvoiceNo, StockCode, and Quantity. Duplicates were removed using SQL's window function ROW_NUMBER():

```
--Checking for the Duplicate Values:
Select
    InvoiceNo
          ,StockCode
          ,Quantity
          ,Count(*) As DuplicateCount
From
  Online_Retail_sales
Group By
   InvoiceNo
          ,StockCode
          ,Quantity
Having
   Count(*) > 1 ;
```




```
--Removing Duplicates:
With CTE As (
      Select
                  *,
                  ROW_NUMBER() Over(Partition By InvoiceNo , StockCode , Quantity
                  Order By (Select Null)) As RN
               From
                  Online_Retail_sales
        )
```

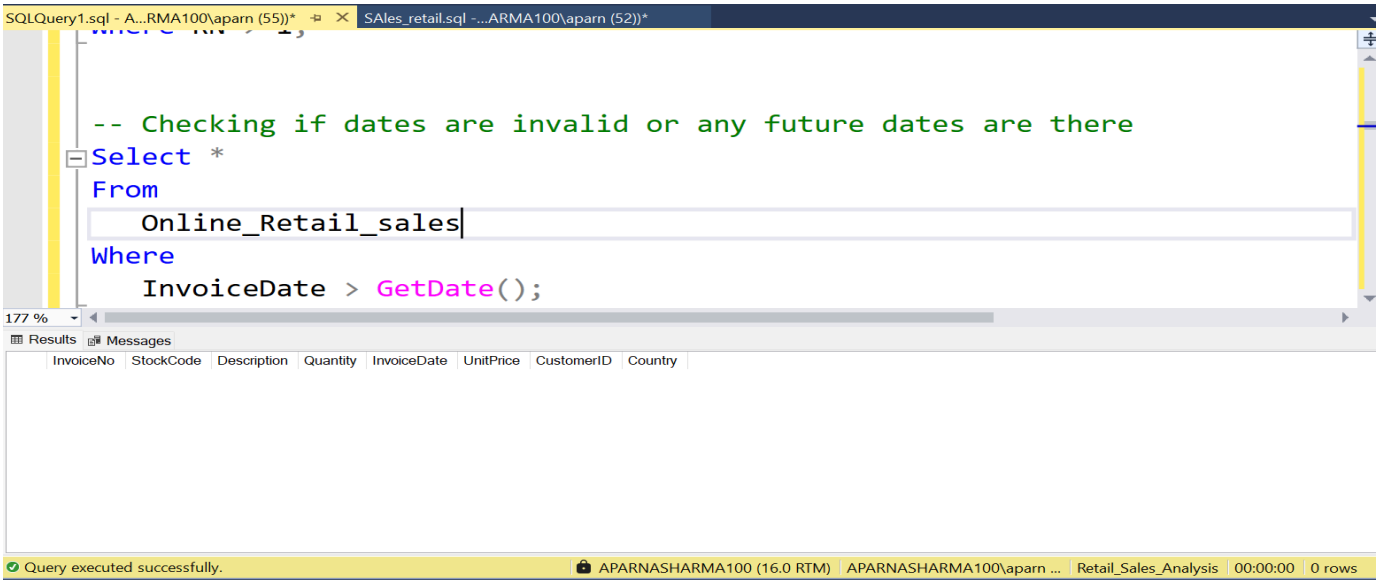
Delete From CTE
Where RN > 1;



4. Validate Invoice Date

Dates were checked to ensure there were no future-dated transactions:

SELECT *
FROM
Online_Retail_sales
WHERE
InvoiceDate > GETDATE();

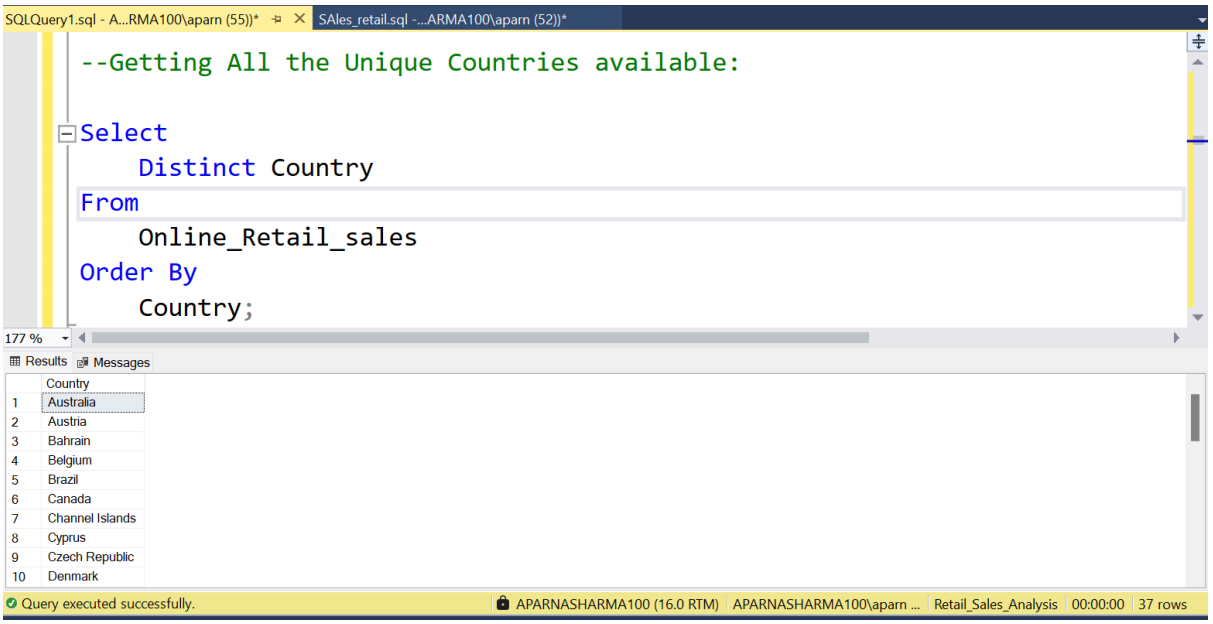


6. Standardize Country Names

All unique country values were reviewed for consistency:

SELECT
DISTINCT Country
FROM

```
    Online_Retail_sales
ORDER BY
    Country;
```



Note:

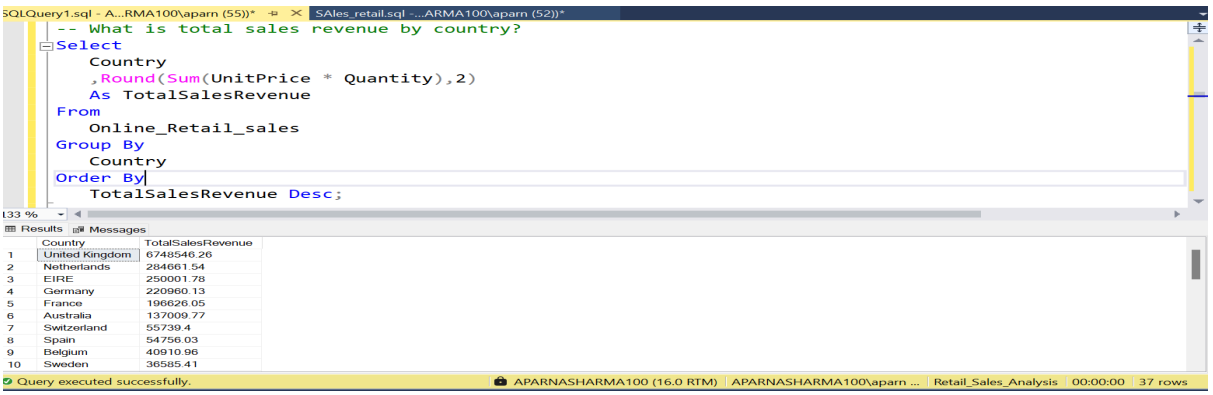
These cleaning steps helped in preparing clean and reliable sales data for further analysis using SQL queries, Power BI dashboards.

3 SQL Business Analysis Queries

The following queries were performed on the cleaned dataset to extract key business insights.

1. Total Sales Revenue by Country:

```
Select
  Country
  ,Round(Sum(UnitPrice * Quantity),2)
  As TotalSalesRevenue
From
  Online_Retail_sales
Group By
  Country
Order By
   TotalSalesRevenue Desc;
```



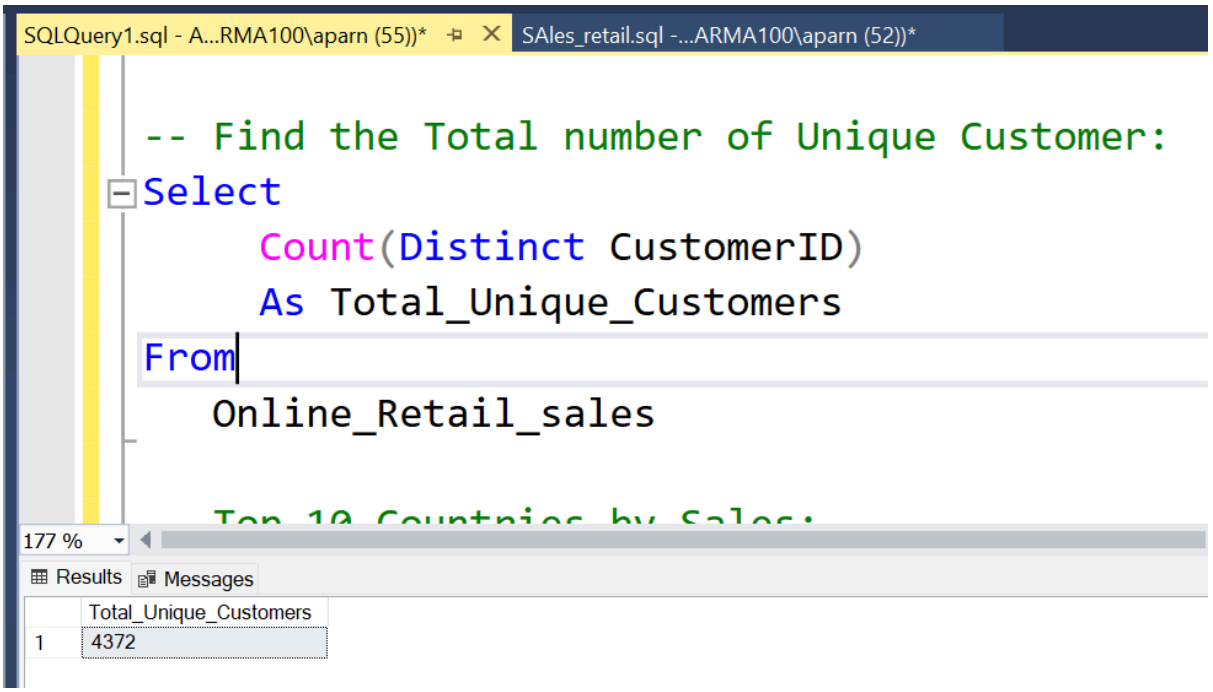
Insight:

Identifies which countries contribute the most to the business's revenue.

Top 5 are : UK , Netherlands , EIRE , Germany and France.

2. Total Number of Unique Customers:

```
Select
    Count(Distinct CustomerID)
        As Total_Unique_Customers
From
  Online_Retail_sales;
```



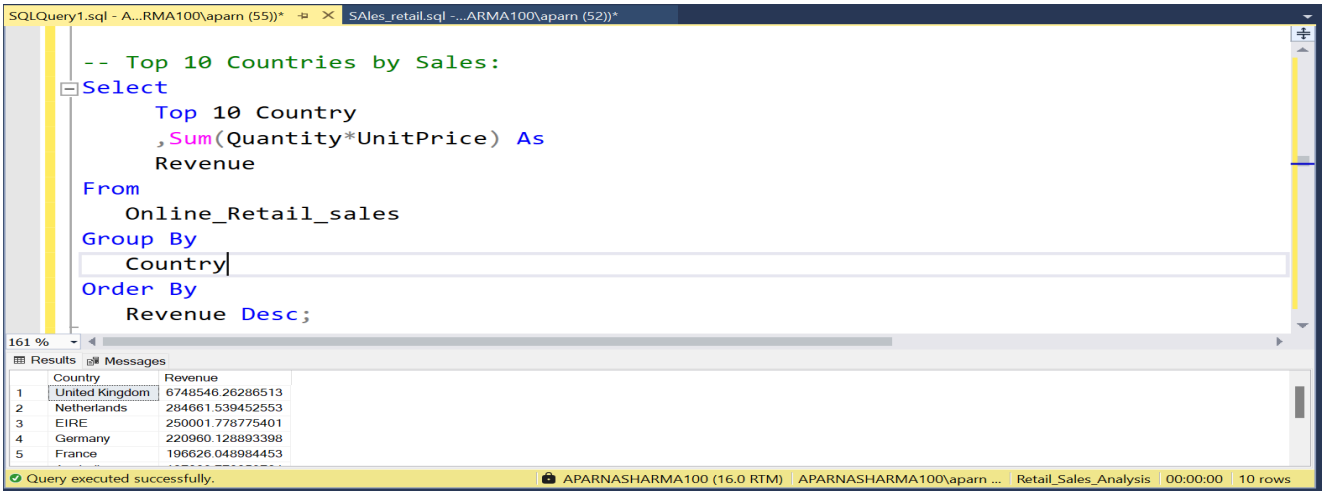
Insight:

Helps measure the active customer base.

3. Top 10 Countries by Sales :

```
Select
    Top 10 Country
        ,Sum(Quantity*UnitPrice) As
```

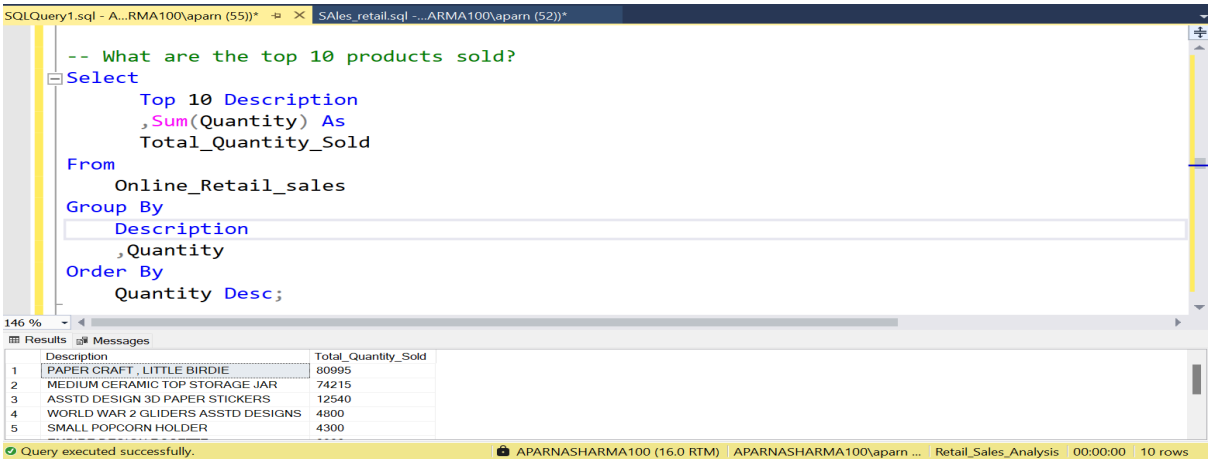
```
        Revenue
From
  Online_Retail_sales
Group By
  Country
Order By
  Revenue Desc;
```



Insight:
Useful for targeting marketing or business expansion.

4. Top 10 Products Sold :

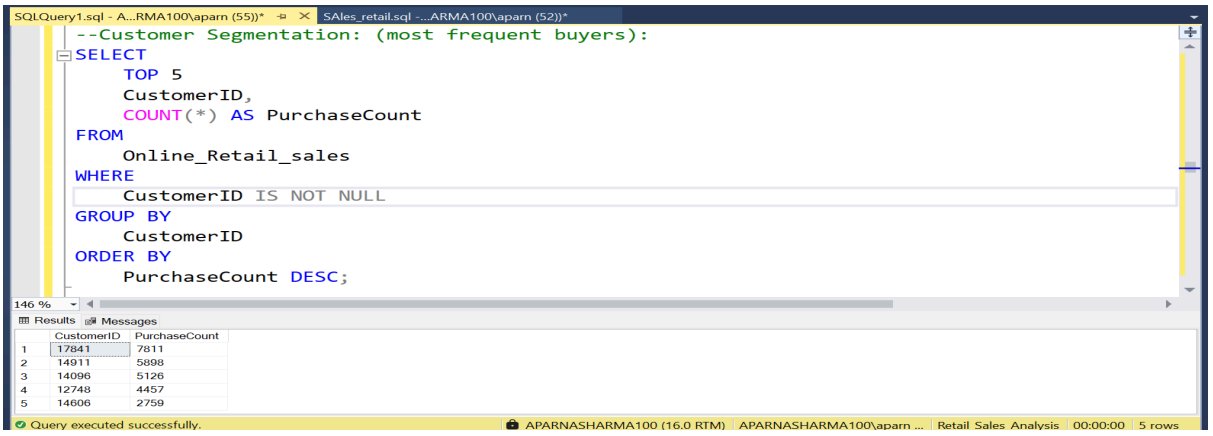
```
Select
    Top 10 Description
        ,Sum(Quantity) As
        Total_Quantity_Sold
From
  Online_Retail_sales
Group By
  Description
       ,Quantity
Order By
  Quantity Desc;
```



Insight:
Shows best-selling products.

5. Customer Segmentation: Most Frequent Buyers

```
SELECT
   TOP 5
   CustomerID,
   COUNT(*) AS PurchaseCount
FROM
   Online_Retail_sales
WHERE
   CustomerID IS NOT NULL
GROUP BY
   CustomerID
ORDER BY
   PurchaseCount DESC;
```

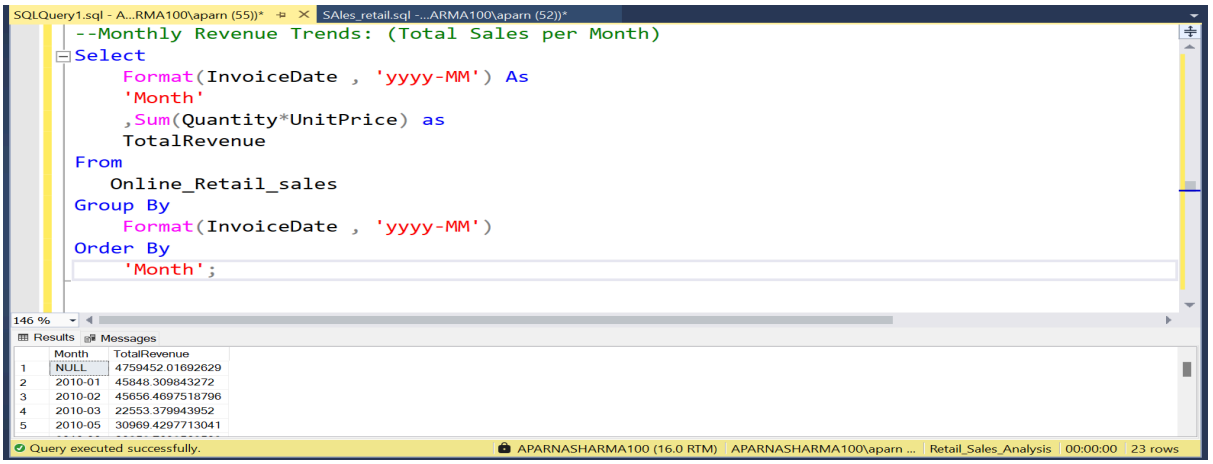


Insight:
Helps identify loyal or high-frequency customers.

6. Monthly Revenue Trends

```
Select
   Format(InvoiceDate , 'yyyy-MM') As
```

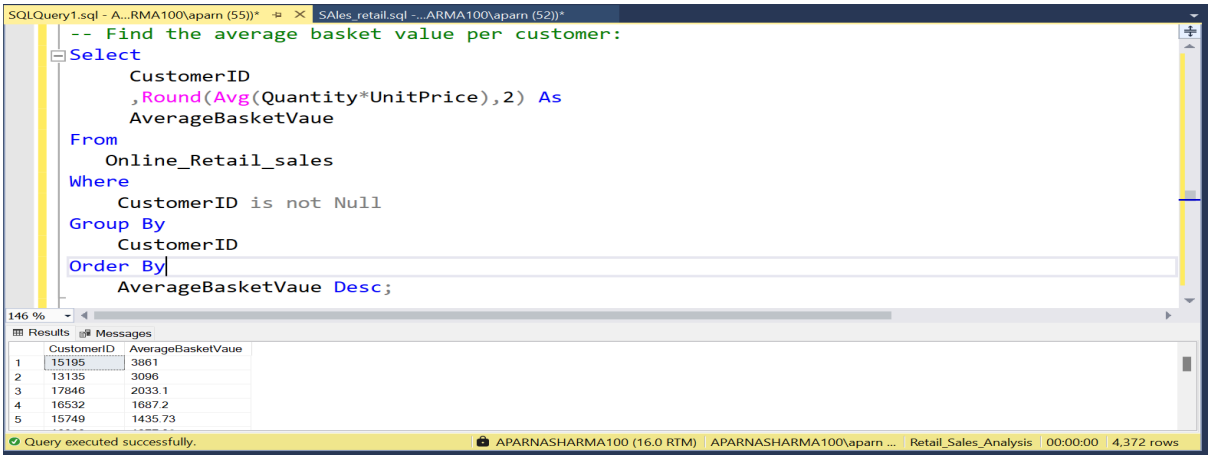
```
        'Month'
    ,Sum(Quantity*UnitPrice) as
        TotalRevenue
From
  Online_Retail_sales
Group By
   Format(InvoiceDate , 'yyyy-MM')
Order By
   'Month';
```



Insight:
Helps spot seasonal sales patterns.

7. Average Basket Value Per Customer

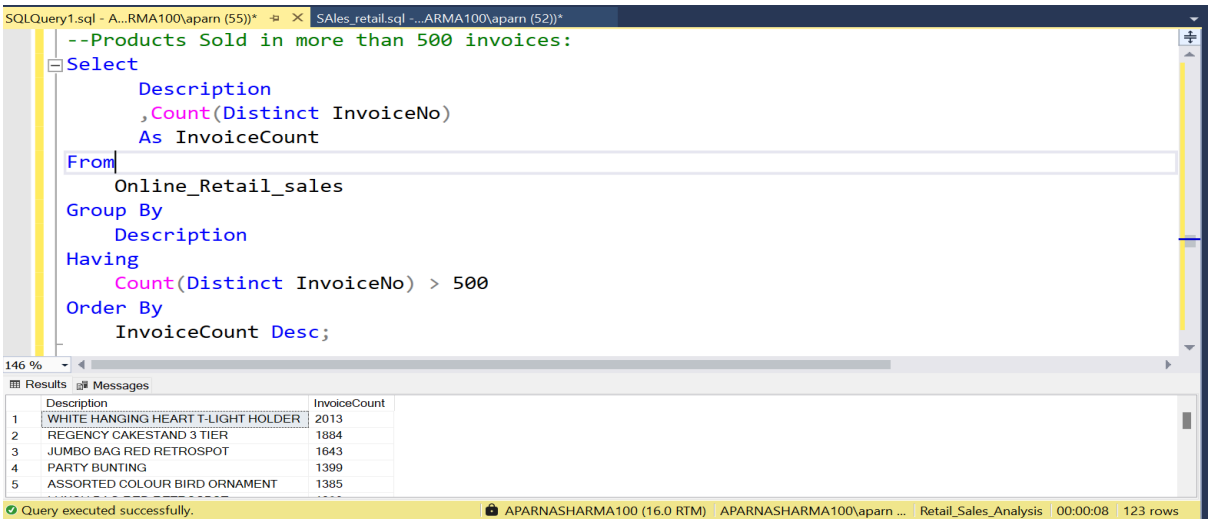
```
Select
    CustomerID
        ,Round(Avg(Quantity*UnitPrice),2) As
        AverageBasketVaue
From
  Online_Retail_sales
Where
   CustomerID is not Null
Group By
   CustomerID
Order By
   AverageBasketVaue Desc;
```



Insight:
Useful for understanding customer spending habits.

8. Products Sold in More Than 500 Invoices

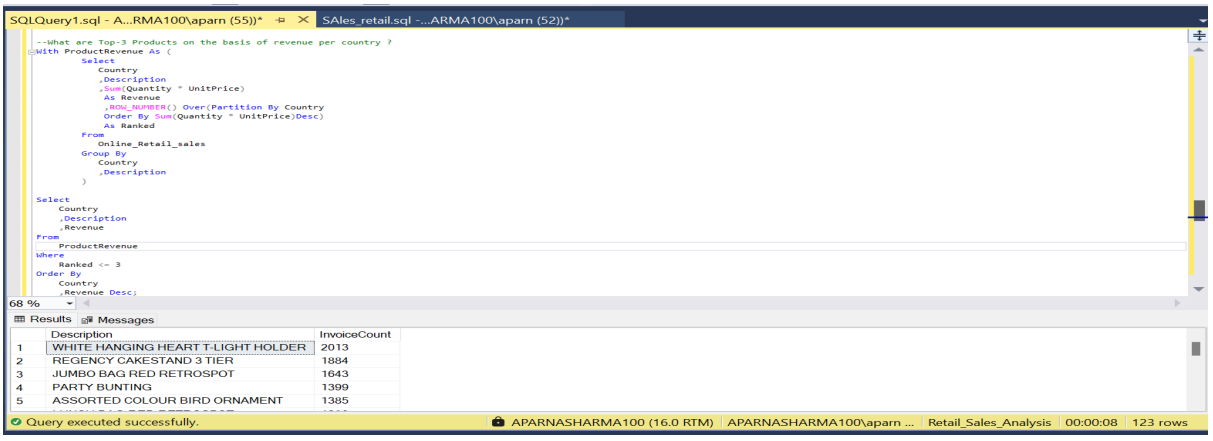
```
Select
    Description
        ,Count(Distinct InvoiceNo)
        As InvoiceCount
From
   Online_Retail_sales
Group By
   Description
Having
   Count(Distinct InvoiceNo) > 500
Order By
   InvoiceCount Desc;
```



Insight:
Identifies consistently popular products.

9. Top 3 Products by Revenue Per Country

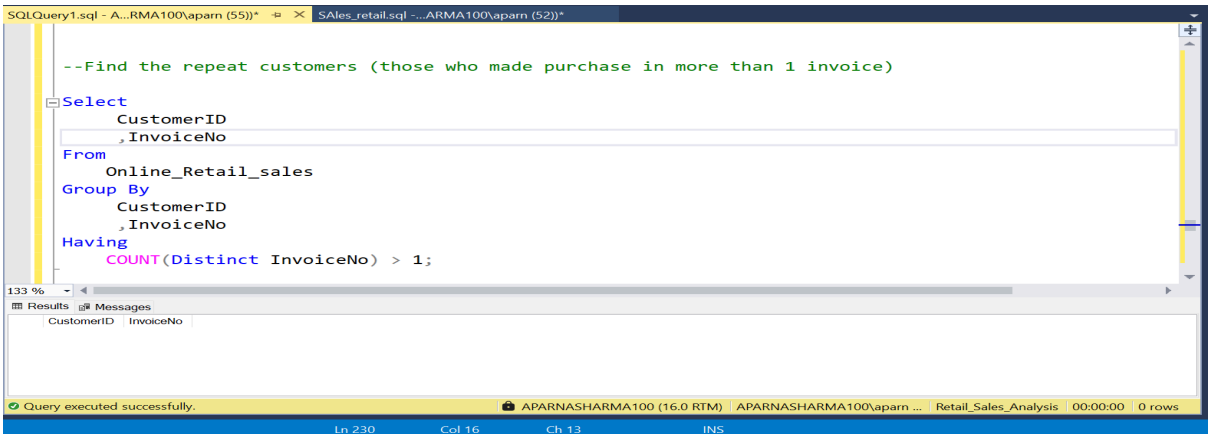
```
With ProductRevenue As (
        Select
          Country
          ,Description
                ,Sum(Quantity * UnitPrice)
                 As Revenue
                            ,ROW_NUMBER() Over(Partition By Country
                            Order By Sum(Quantity * UnitPrice)Desc)
                            As Ranked
        From
          Online_Retail_sales
        Group By
          Country
                            ,Description
                    )

Select
    Country
          ,Description
          ,Revenue
From
    ProductRevenue
Where
    Ranked <= 3
Order By
    Country
          ,Revenue Desc;
```



Insight:
Helps prioritize product distribution and inventory by region.

10. Repeat Customers

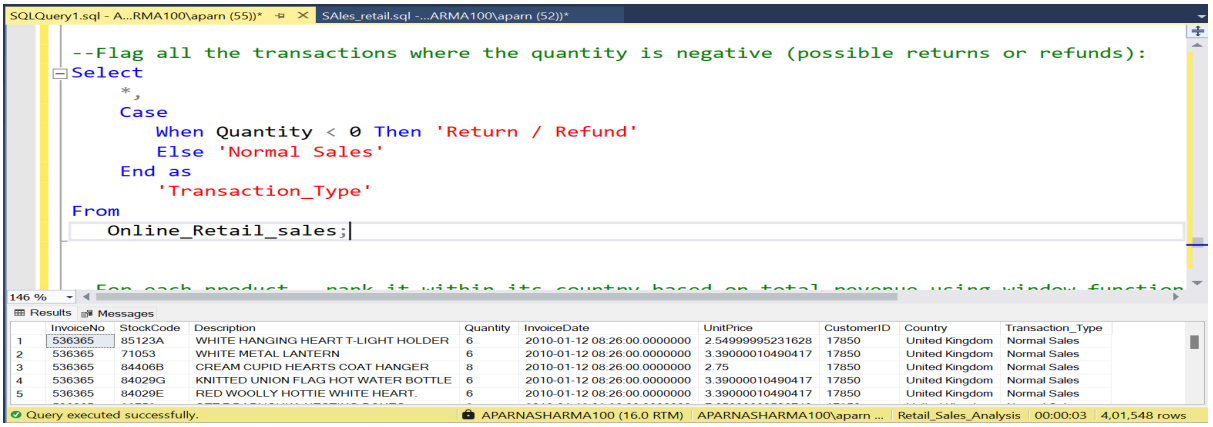
```
Select
    CustomerID
        ,InvoiceNo
From
   Online_Retail_sales
Group By
    CustomerID
        ,InvoiceNo
Having
   COUNT(Distinct InvoiceNo) > 1;
```



Insight:
Helps measure customer retention rate.

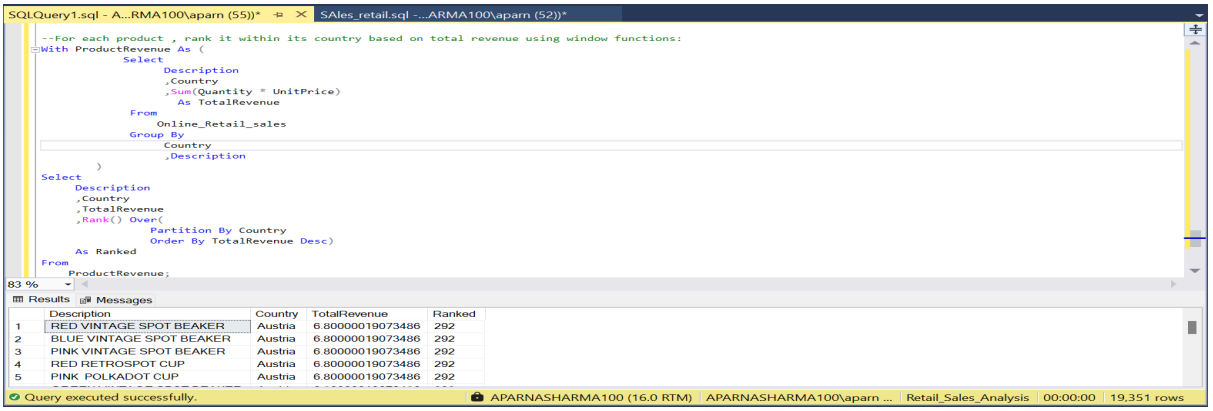
11. Flag Transactions with Negative Quantity (Returns or Refunds)

```
Select
   *,
        Case
          When Quantity < 0 Then 'Return / Refund'
          Else 'Normal Sales'
        End as
          'Transaction_Type'
From
  Online_Retail_sales;
```



12. Product Ranking by Revenue Within Each Country

```
With ProductRevenue As (
        Select
            Description
                ,Country
                ,Sum(Quantity * UnitPrice)
                  As TotalRevenue
         From
            Online_Retail_sales
         Group By
            Country
                ,Description
              )

Select
    Description
        ,Country
        ,TotalRevenue
        ,Rank() Over(
                Partition By Country
                Order By TotalRevenue Desc)
        As Ranked
From
   ProductRevenue;
```



Insight:
Provides product popularity ranking for regional business strategy.

Conclusion:

Project Summary and Key Learnings

This project involved cleaning and analyzing retail sales data using Microsoft SQL Server.

Key insights derived:

- ❖ Revenue distribution by country and product
- ❖ Customer segmentation: identifying top buyers and repeat customers
- ❖ Seasonal sales trends
- ❖ Return/refund detection using SQL case logic
- ❖ Product ranking with window functions

Skills demonstrated:

- ➔ SQL aggregation and grouping
- ➔ Window functions: ROW_NUMBER, RANK
- ➔ Data cleaning using SQL (removing duplicates, handling missing values)
- ➔ Business insights preparation for dashboards (Power BI)

References :

Dataset: Online Retail Dataset (Publicly available version from Kaggle/UCI Repository)

SQL Server Documentation

Power BI Official Guide.