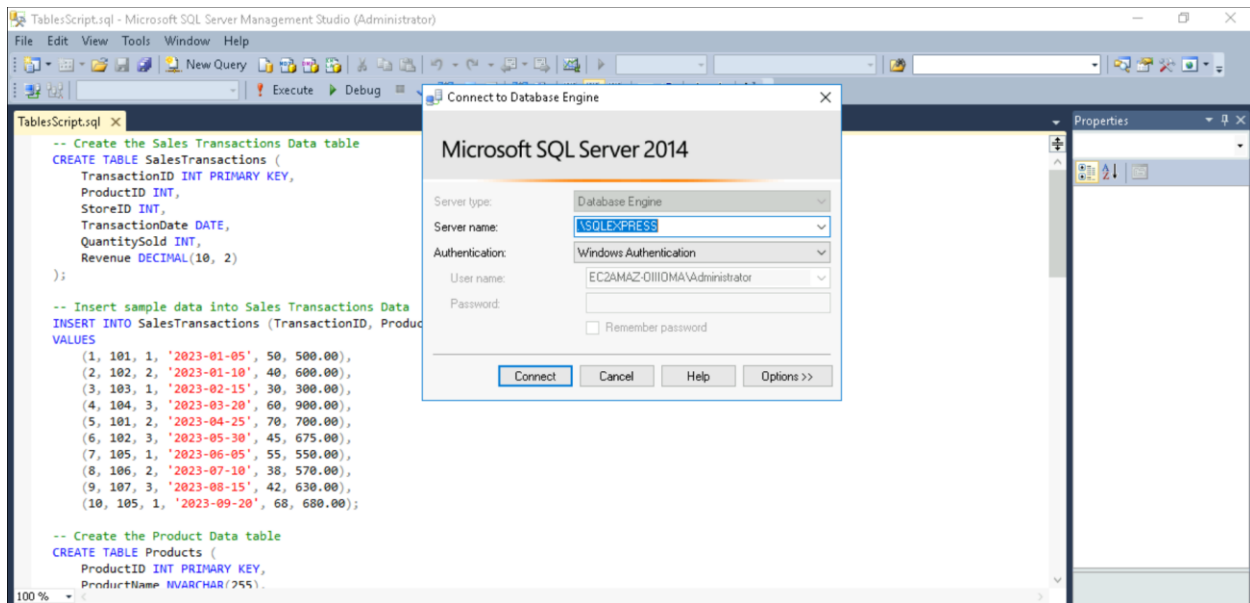


Hands-on Assessment

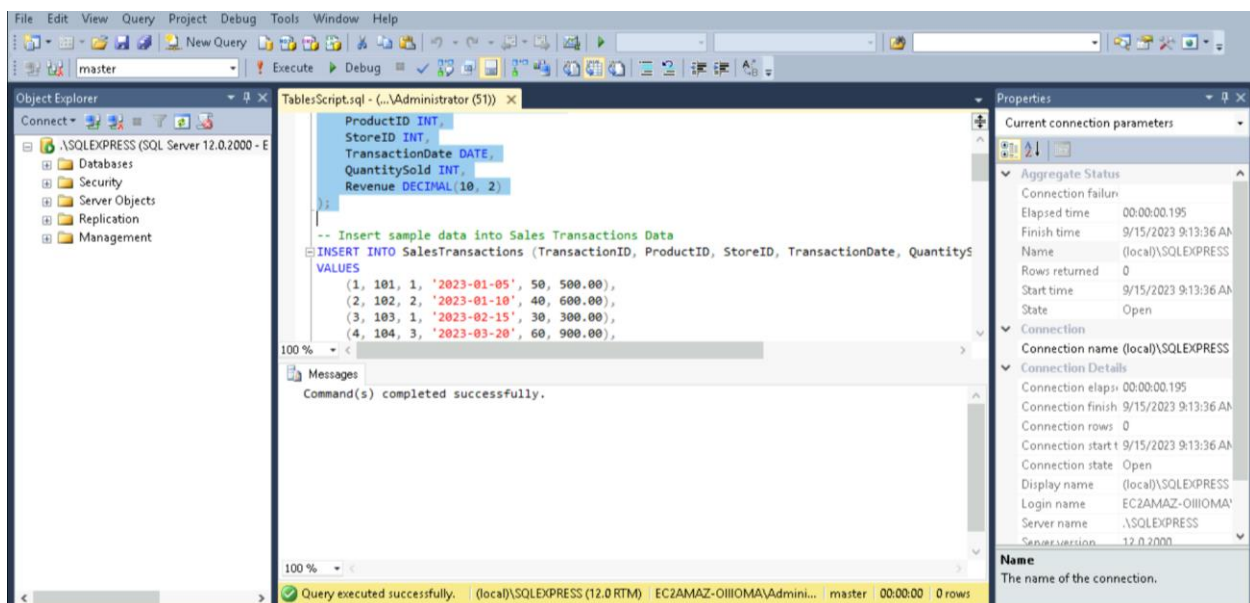
Power BI

Requirement 1: Data Loading

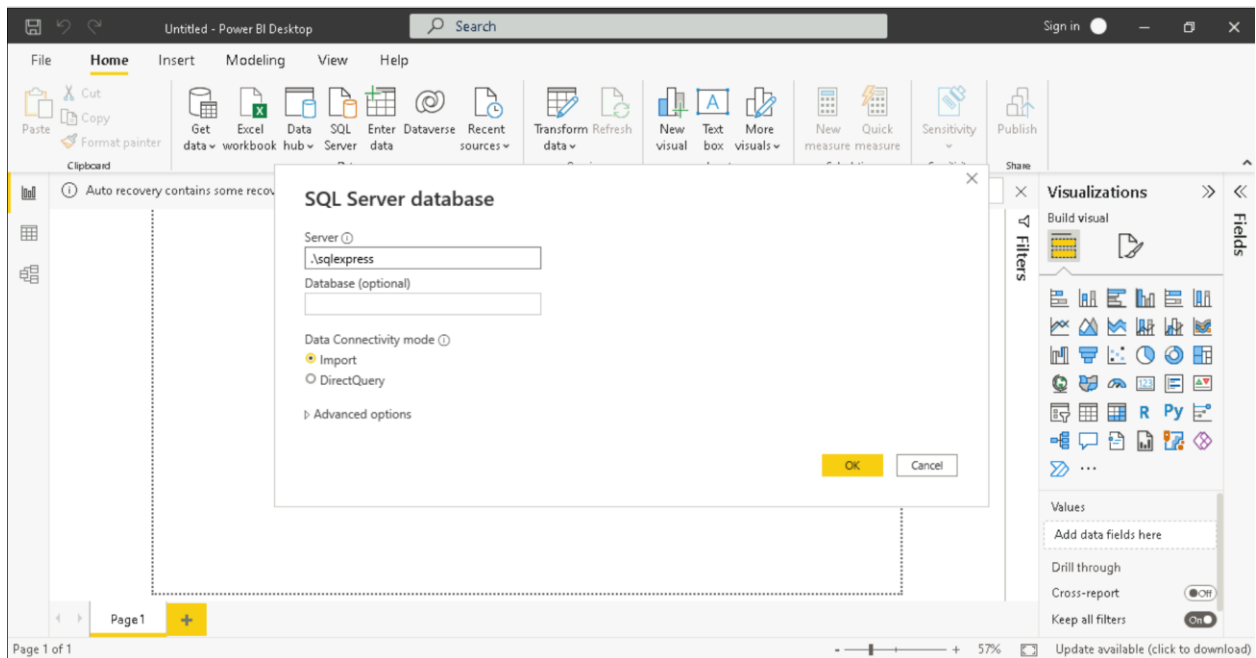
Open GitHub link and download the data file



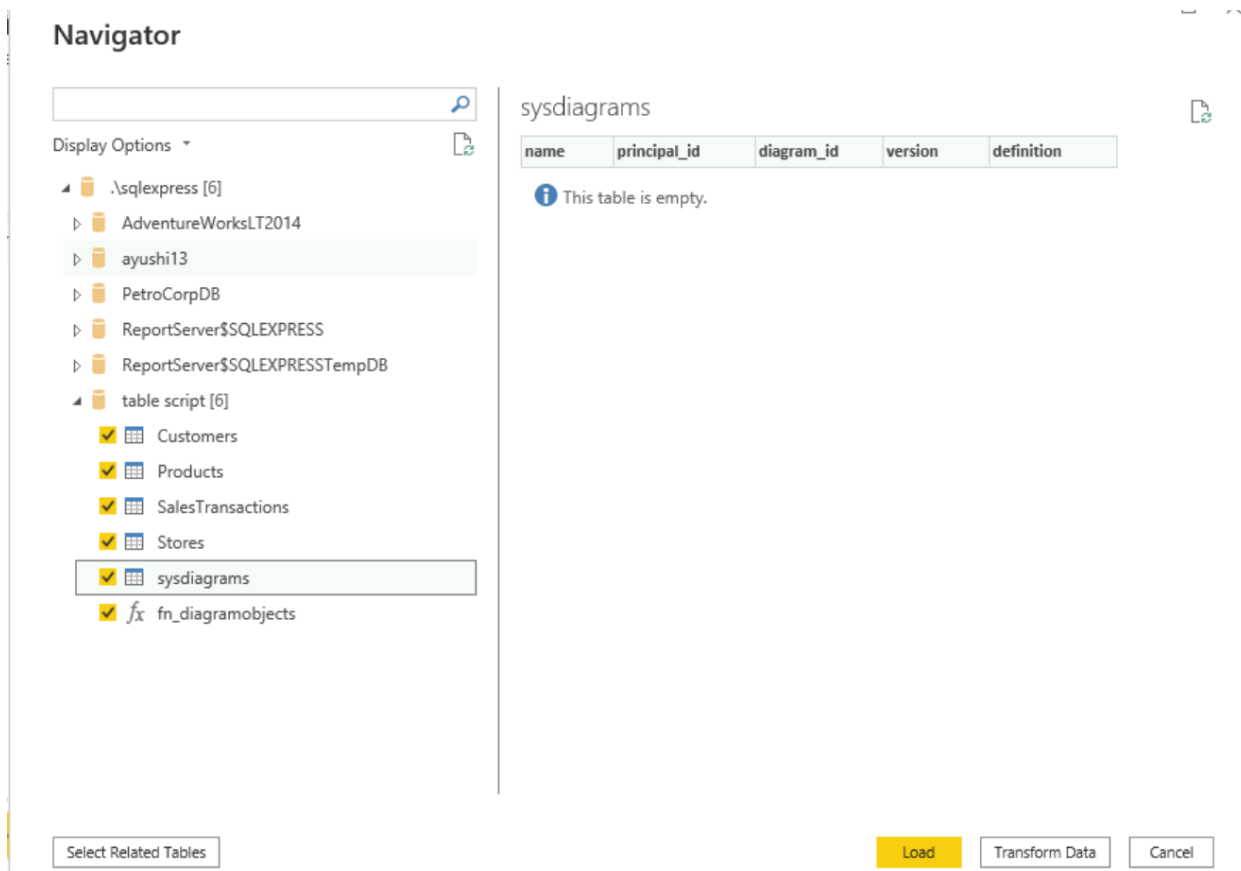
Create a new database and run all the queries.



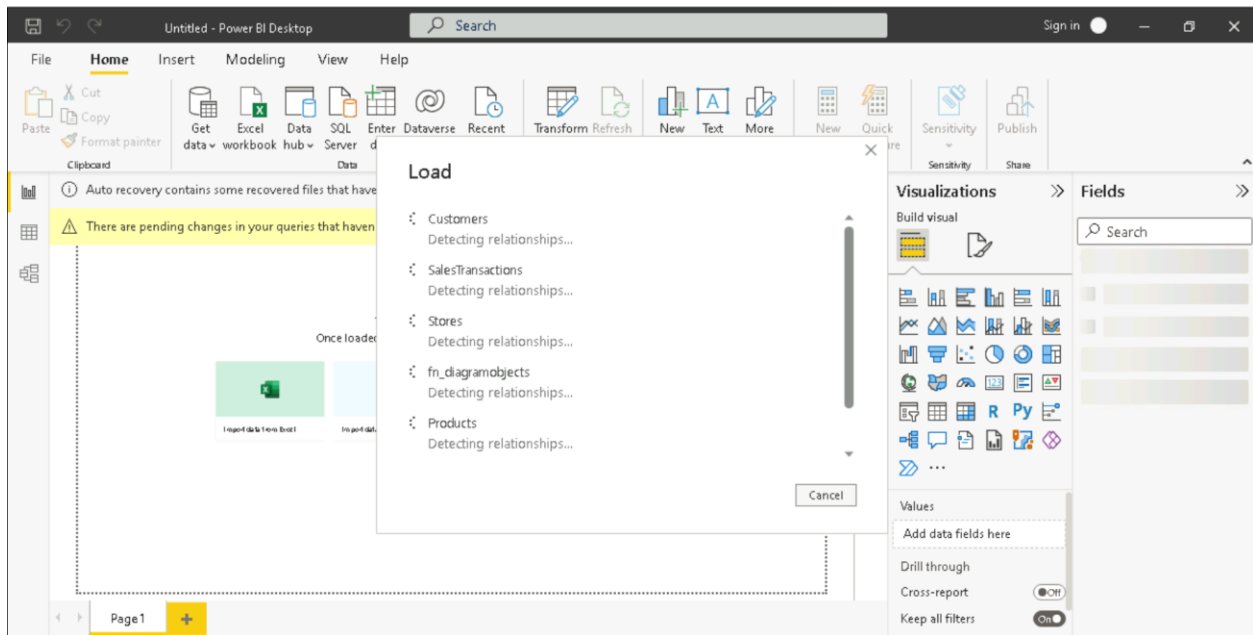
Connecting Power Bi with SQL server.



Loading database in Power Bi



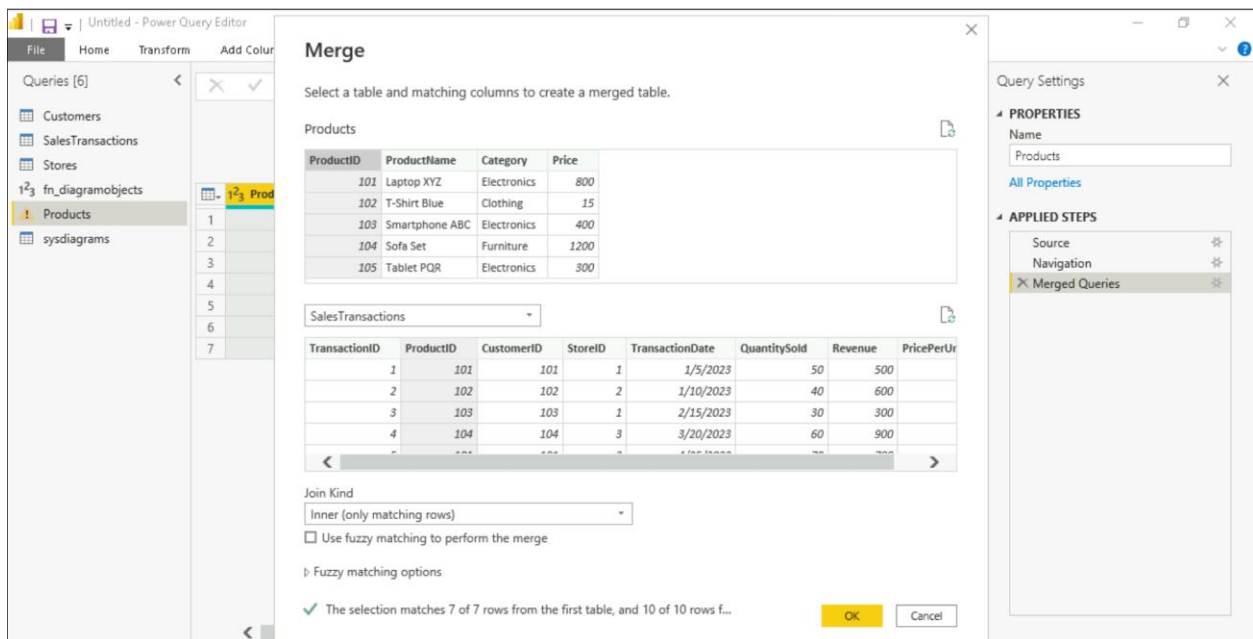
Data Loading is done.



Requirement 2: Data Transformation

- Merge Tables:

Merging tables using inner join



Sales and Product table merged

The screenshot shows the Power Query Editor interface. The main query is defined as a nested join of the 'Products' and 'SalesTransactions' tables. The resulting table has 7 columns: ID, ProductName, Category, Price, and SalesTransactions. The 'SalesTransactions' column contains a 'Table' icon, indicating it's a nested table. The 'Query Settings' pane on the right shows the query name 'Products' and the applied steps: Source, Navigation, and Merged Queries.

ID	ProductName	Category	Price	SalesTransactions
1	101 Laptop XYZ	Electronics	800	Table
2	102 T-Shirt Blue	Clothing	15	Table
3	103 Smartphone ABC	Electronics	400	Table
4	104 Sofa Set	Furniture	1200	Table
5	105 Tablet PQR	Electronics	300	Table
6	106 Jeans Black	Clothing	40	Table
7	107 Chair	Furniture	80	Table

- Clean Data:

Original format of Phone in Customers

The screenshot shows the Power Query Editor interface with a query named 'Customers'. The query is defined as a table script. The resulting table has 5 columns: CustomerID, CustomerName, Email, and Phone. The 'Phone' column contains phone numbers in a standard format, including area codes in parentheses.

CustomerID	CustomerName	Email	Phone
1	101 Customer A	customerA@email.com	(123) 456-7890
2	102 Customer B	customerB@email.com	(234) 567-8901
3	103 Customer C	customerC@email.com	(345) 678-9012
4	104 Customer D	customerD@email.com	(456) 789-0123
5	105 Customer E	customerE@email.com	(567) 890-1234
6	106 Customer F	customerF@email.com	(678) 901-2345
7	107 Customer G	customerG@email.com	(789) 012-3456

Replace ' (' and ') ' with a blank

Queries [6] < X ✓ fx = #\"table script\"{Schema=\"dbo\",Item=\"Customers\"}}{Data1}

Replace Values

Replace one value with another in the selected columns.

Value To Find
{

Replace With

Advanced options

OK Cancel

Queries [6] < X ✓ fx = Table.ReplaceValue(dbo.Customers.\"(\".\"\".Replacer.ReplaceText.{\"Phone\"})

Replace Values

Replace one value with another in the selected columns.

Value To Find
}

Replace With

Advanced options

OK Cancel

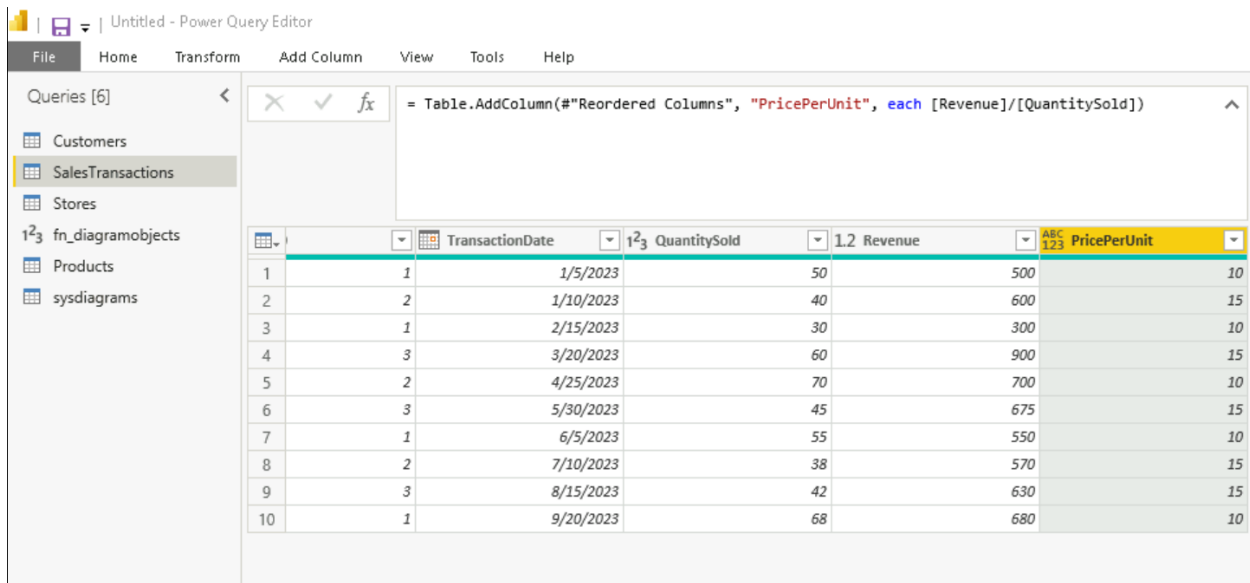
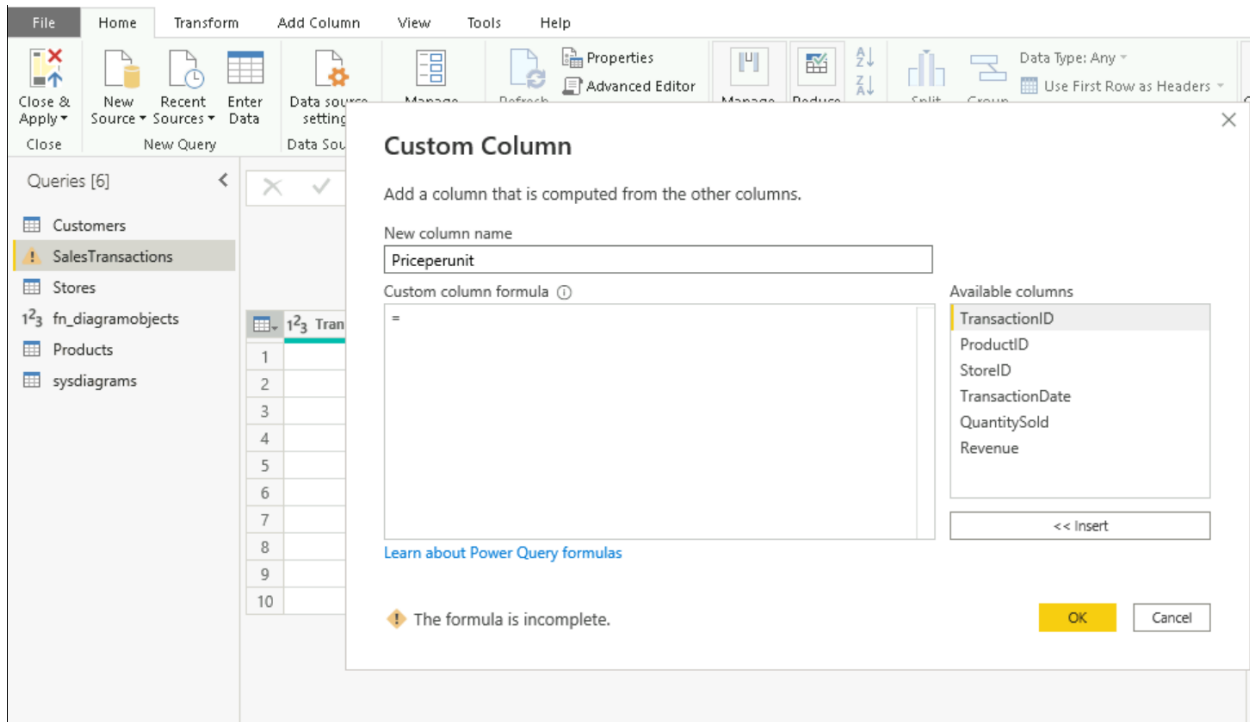
Data is cleaned and here is the new format.

Queries [6] < X ✓ fx = Table.ReplaceValue(#\"Replaced Value\", \"\", \"\", Replacer.ReplaceText, {\"Phone\"})

	CustomerID	CustomerName	Email	Phone
1	101	Customer A	customerA@email.com	123 456-7890
2	102	Customer B	customerB@email.com	234 567-8901
3	103	Customer C	customerC@email.com	345 678-9012
4	104	Customer D	customerD@email.com	456 789-0123
5	105	Customer E	customerE@email.com	567 890-1234
6	106	Customer F	customerF@email.com	678 901-2345
7	107	Customer G	customerG@email.com	789 012-3456

- **Create Calculated Columns:**

Create a new column price per unit and used the formula to derive value.



- **Filter Data:**

Creating a filter on Revenue column .

The screenshot shows the Power Query Editor interface. A 'Filter Rows' dialog box is open, allowing the user to apply filter conditions to the rows in the table. The dialog has two tabs: 'Basic' and 'Advanced'. The 'Basic' tab is selected. Under 'Keep rows where 'Revenue'', there are two conditions: 'is greater than' 500 and 'is less than' 900. The background table has columns: TransactionDate, QuantitySold, Revenue, and PricePerUnit. The 'Revenue' column is highlighted in yellow. The 'Query Settings' pane on the right shows the 'Name' as 'SalesTransactions' and the 'APPLIED STEPS' list includes 'Source', 'Navigation', 'Added Custom', 'Reordered Columns', and 'Added Custom1'.

	TransactionDate	QuantitySold	Revenue	PricePerUnit
1	1/5/2023	50	500	10
2	1/10/2023	40	600	15
			300	10
			900	15
			700	10
			675	15
			550	10
			570	15
			630	15
			680	10

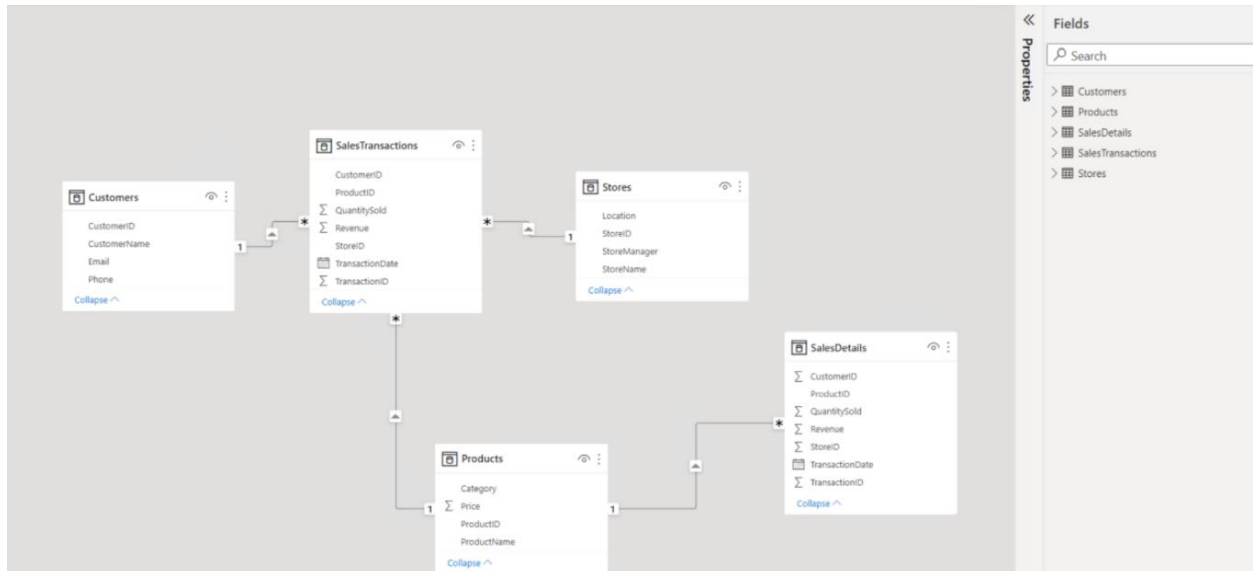
Revenue column values are now filtered.

The screenshot shows the Power Query Editor interface after applying the filter. The 'Query Settings' pane on the right shows the 'APPLIED STEPS' list updated to include 'Filtered Rows'. The background table now only contains the rows where the 'Revenue' value is between 500 and 900.

	TransactionDate	QuantitySold	Revenue	PricePerUnit
1	1/10/2023	40	600	15
2	4/25/2023	70	700	10
3	5/30/2023	45	675	15
4	6/5/2023	55	550	10
5	7/10/2023	38	570	15
6	8/15/2023	42	630	15
7	9/20/2023	68	680	10

Requirement 3: Data Modelling

- Create Relationships:



- Create Hierarchies:

The screenshot shows a data model tool interface. On the left, a table view displays data for the Stores table:

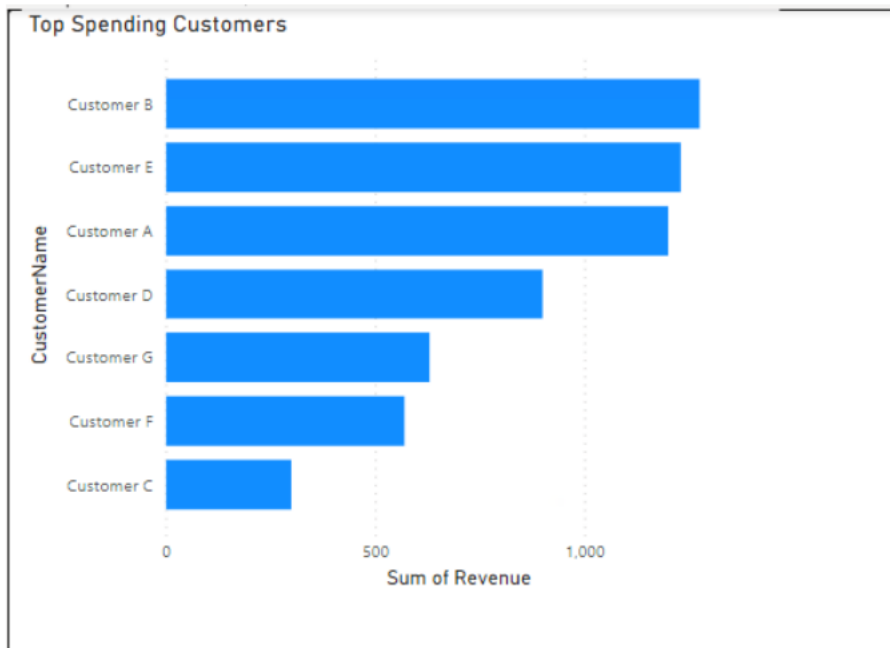
StoreID	StoreName	Location	StoreManager	Address	Country
1	Downtown	New York	John Smith	Downtown, New York	USA
2	Uptown	Los Angeles	Jane Doe	Uptown, Los Angeles	USA
3	Suburbia	Chicago	Mike Johnson	Suburbia, Chicago	USA

On the right, the 'Fields' pane shows a hierarchy for the Stores table:

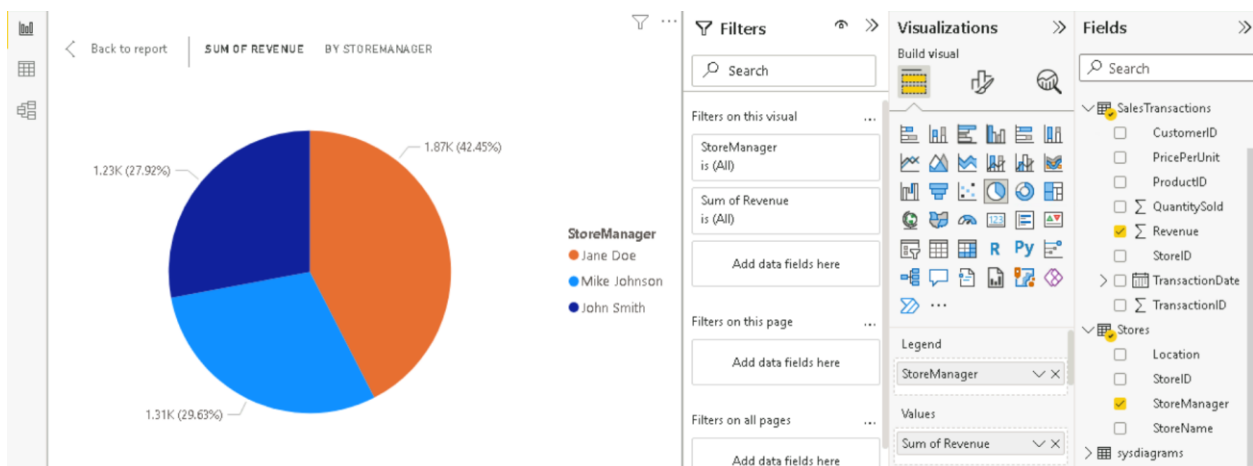
- store
 - SalesDetails
 - StoreID
 - SalesTransactions
 - StoreID
 - Store
 - Address
 - Country
 - Location
 - StoreID
 - StoreManager
 - StoreName
 - StoreName Hierarchy
 - StoreName

Requirement 4: Business Queries and Analysis

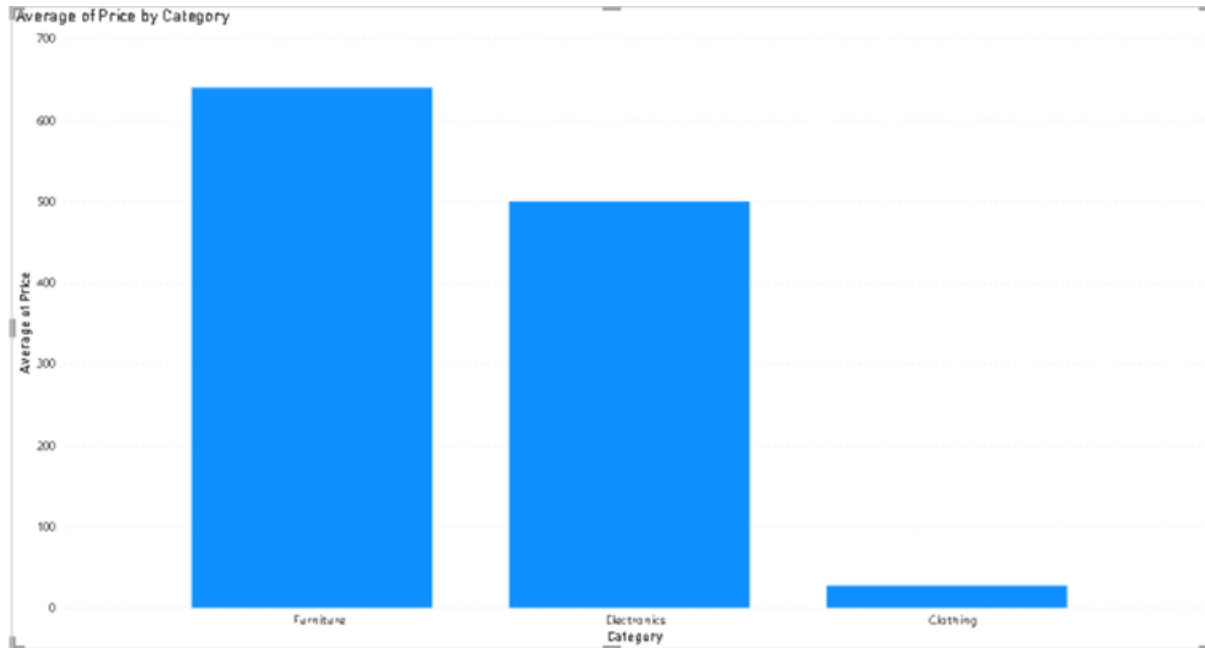
1. Who are the top-spending customers based on their total purchase amount?



2. How is sales revenue distributed among different store managers?

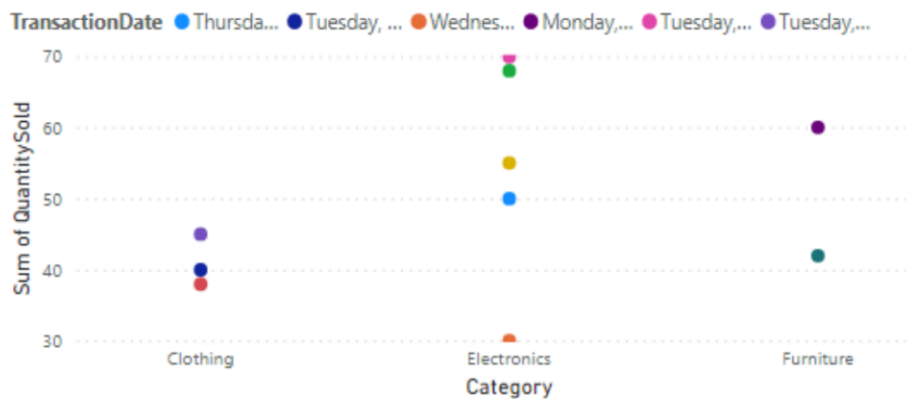


3. What is the average price of products in each category?

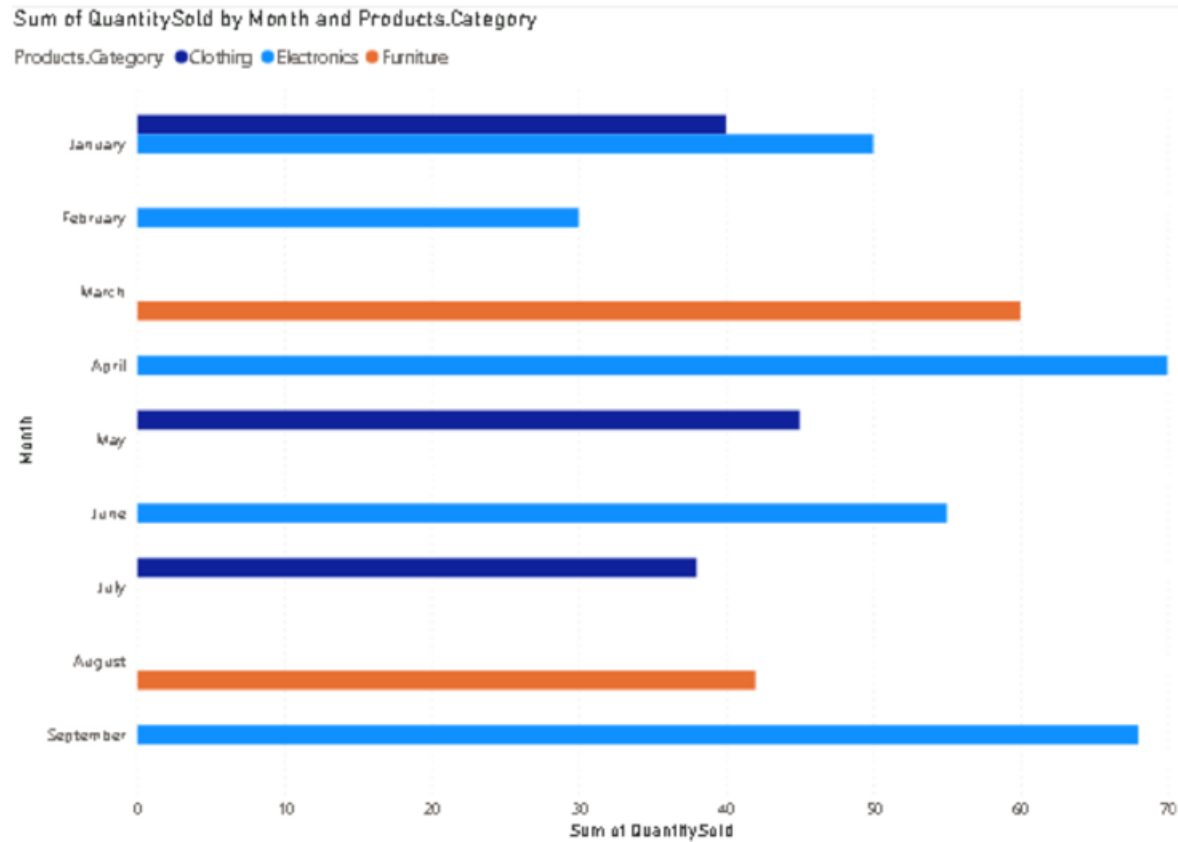


4. Are there specific days of the week when sales are higher?

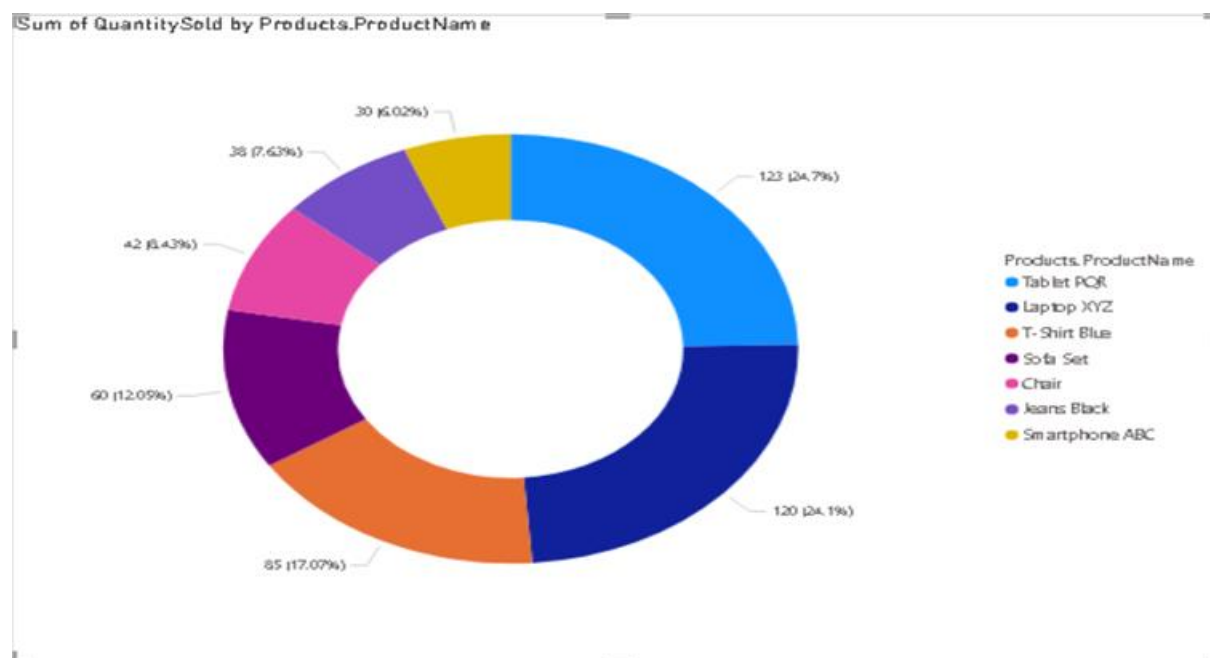
Sales Trends by product category on daily basis



5. How do sales trends vary by product category on a monthly basis?

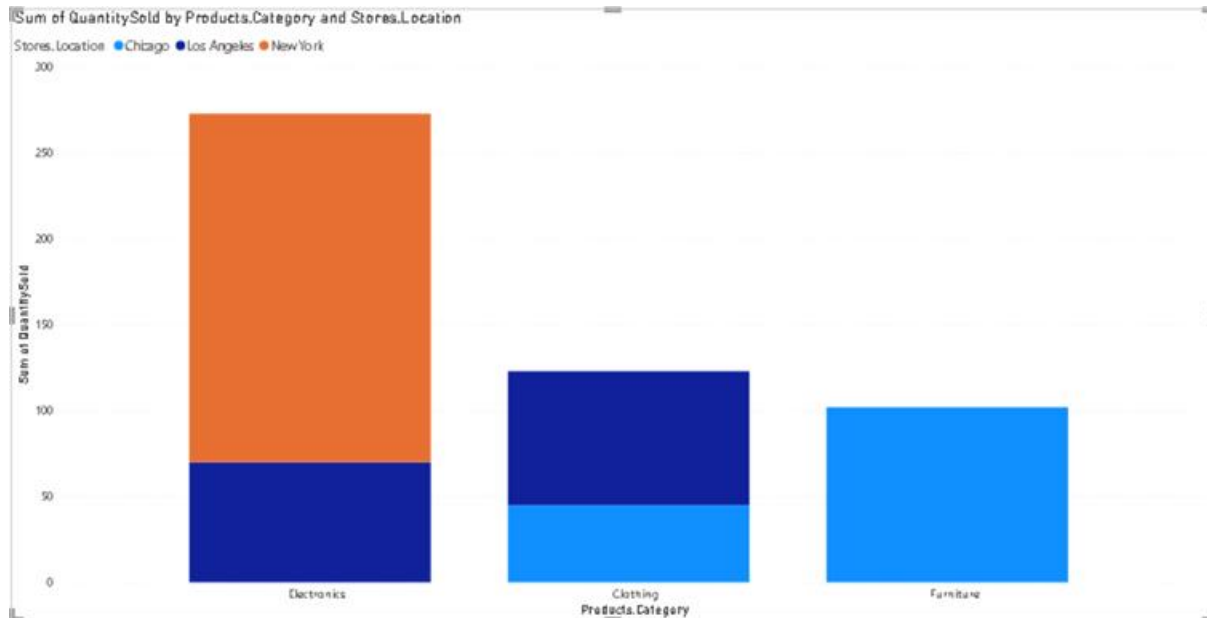


6. What percentage of products account for 80% of total sales revenue?

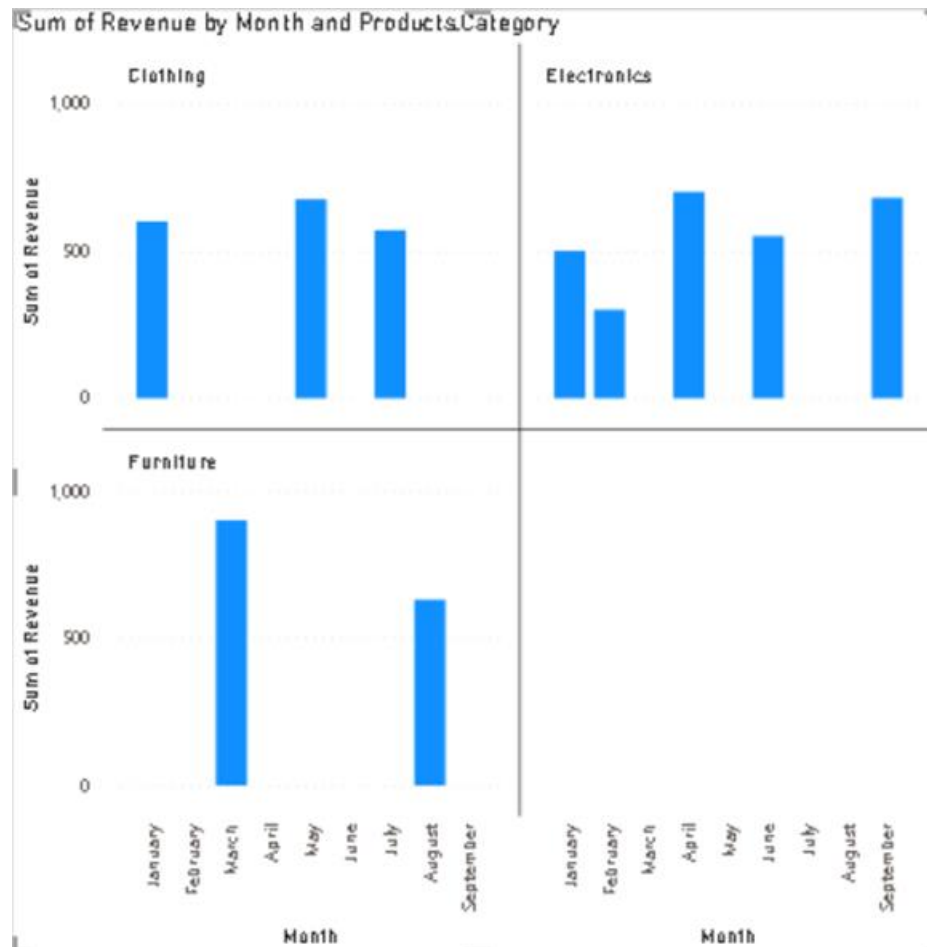


7. Are there any trends in repeat customer purchases?

8. Which product categories perform best at each store location?



9. Are there any seasonal patterns or trends in sales for specific products or categories?



10. Can customers be segmented into high, medium, and low-value segments based on their purchase history.

Requirement 5: Data Insights and Recommendations –

- **Analyze Patterns:** Identify patterns and trends in the data, such as seasonality or regional variations and show it or mark it.
- **Generate Insights:** Provide actionable insights based on your analysis. For example, suggest increasing marketing efforts for the most profitable product category.
- **Create Visual Stories:** Use storytelling techniques to communicate your findings effectively in the reports and dashboards.