POWER QUERY, DATA MODEL, POWER PIVOT TABLE, DAX, DASHBOARD

I used the Business Intelligence (BI) tools in **Microsoft Excel** which are **Power Query, Data Models, Power Pivot Tables and DAX**. The Excel file for my project can be viewed in **Github** saved as **Supermarket Data Model.xlsx** Also the **dataset** is saved in this folder as various **CSV files**.

My project involves using Business Intelligence (BI) tools in Excel to analyse data for a Supermarket. The dataset includes information about customers, products, stores, geographic locations and sales transactions for product quantities over a two year time period 1997 and 1998.

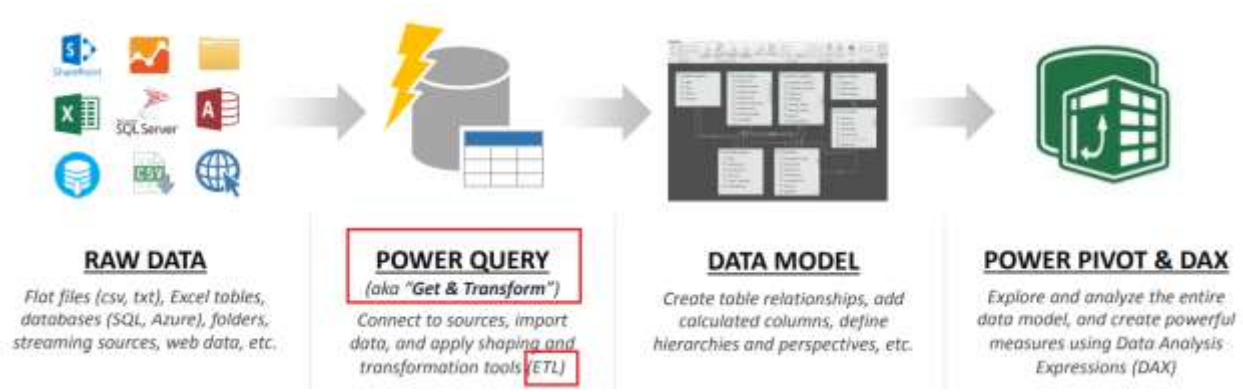
Power Query was used as an ETL (Extract, Transform, Load) tool to transform the dataset and to structure the data into various tables.

Data Modelling was used to create relationships between the tables.

Power Pivot Tables were used for data analysis.

DAX formulas were used to create Measures in the Power Pivot Tables, and Calculated Columns in the Data Model.

Dashboards were created to provide interactive data visualizations



POWER QUERY

Power Query is an **ETL (Extract, Trasform, Load)** tool found within Excel (and other BI tools such as Power BI and Azure Data Lake Storage). The Power Query Editor can be used to connect to and **extract data from various different sources** such as SQL Server, MySQL, CSV files, PDF files and websites on the internet. The advantage of using Power Query is that we can **import several million rows of data into Excel** whereas an ordinary Excel file has a limit of storing 1 million rows of data.

This data is then cleaned and transformed using the Power Query Editor's transformation functions.

The cleaned data can then be loaded into a Data Model in Excel to perform data analysis.

Some of the **transformations** I performed on the data for this project includes changing data types of columns, renaming column headers, deleting unwanted columns, splitting column data, formatting the number columns (e.g as currency) and text columns (as uppercase letters), sorting columns, filtering columns and merging columns.

I also used the Date Time tool to create a Calendar Table disclosing columns of dates by year, month name and by day.

I created a Conditional Column by applying the conditional test IF – THEN- ELSE

I also used the function "Append Queries" to stack 2 tables of data relating to transactions of quantities sold. Each table had a different years of data for 1997 and 1998.

As a result the dataset was transformed into various separate tables as follows:

- **Table Customer_Lookup** – This table contains data on customer demographics such as customer gender, customer address & city, cutomer occupation and income level.

- **Table Product_Lookup** – This table contains data about product brands, product prices and cost.

- **Table Store_Lookup** – This table contains data about various stores such as if they belong to a large Supermarket or if they are small grocery stores, and store location by city

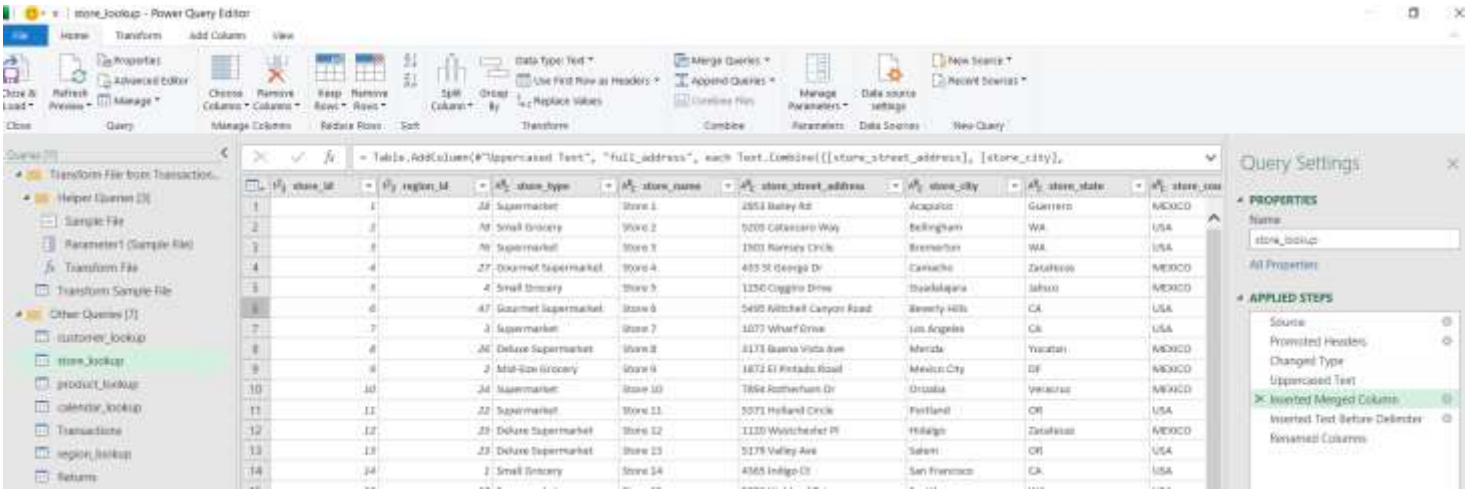
- **Table Region_Lookup** – This table contains data about the location of the sales by region

- **Table Calendar_Lookup** – This table contains data on dates of sales transactions. The dates are displyed in different columns by year, month, day, and week.

- **Table Transactions** – This table contains data on transactions of quantities sold, the customer id number, store id number, product id number and date

- **Table Returns** - This table contains data on transactions of quantities returned, the store id number, product id number and date.

The above mentioned tables were then **loaded** into Excel's **Data Model.**



DATA MODEL

One of the features of Excel is that it enables us to build Data Models. We can load data in the form of tables into the Data Model which is stored in Excel's memory.

Data Modelling is defined as saving data from different sources into tables and then creating **relationships** between those tables. The relationships form connections between the tables based on a common field column (which are called Primary Keys and Foreign Keys).

The **Lookup Tables** contain text to describe the data. For example Table Customer_Lookup contains data about the customer's city location, occupation and income level. The Lookup Tables contain **Primary Keys**.

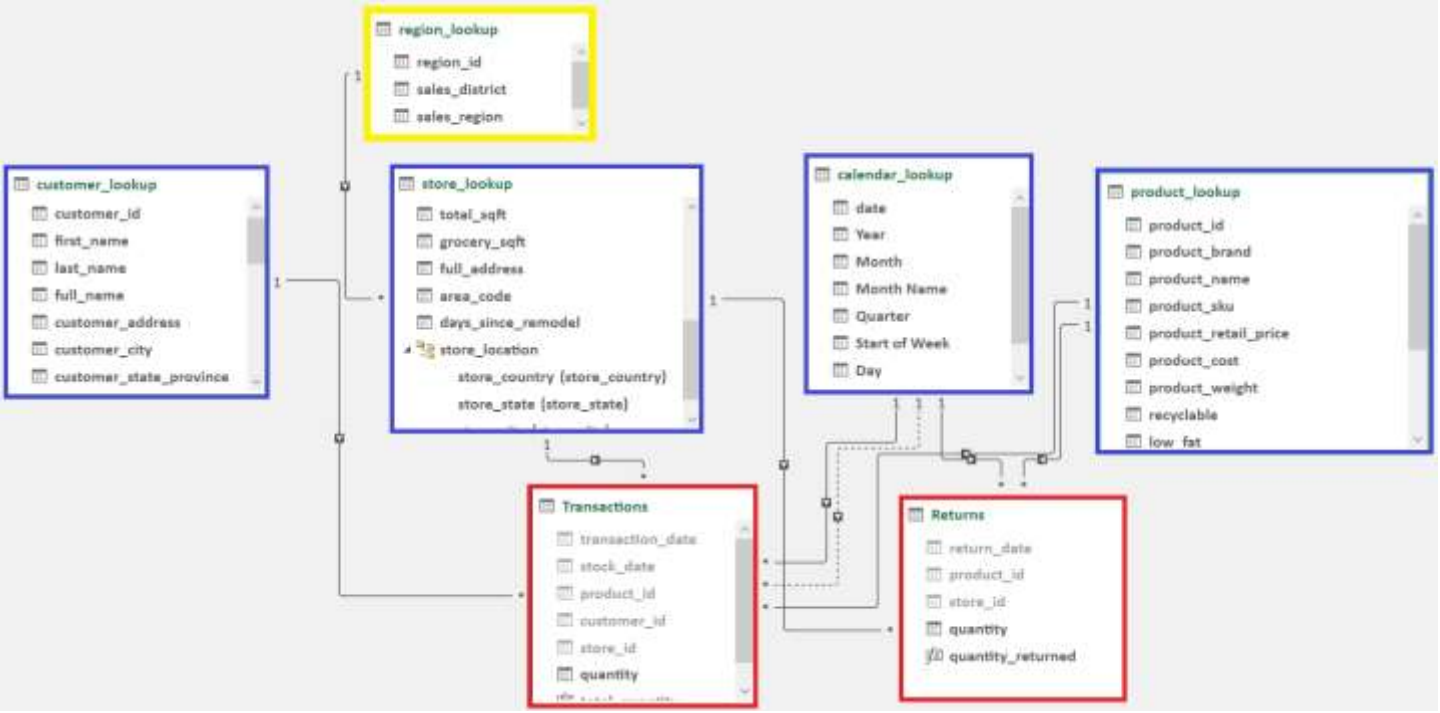
The **Data Tables** contain quantitative values. For example Table Transactions contains data about the quantity of products sold. The Data Tables contain **Foreign Keys**.

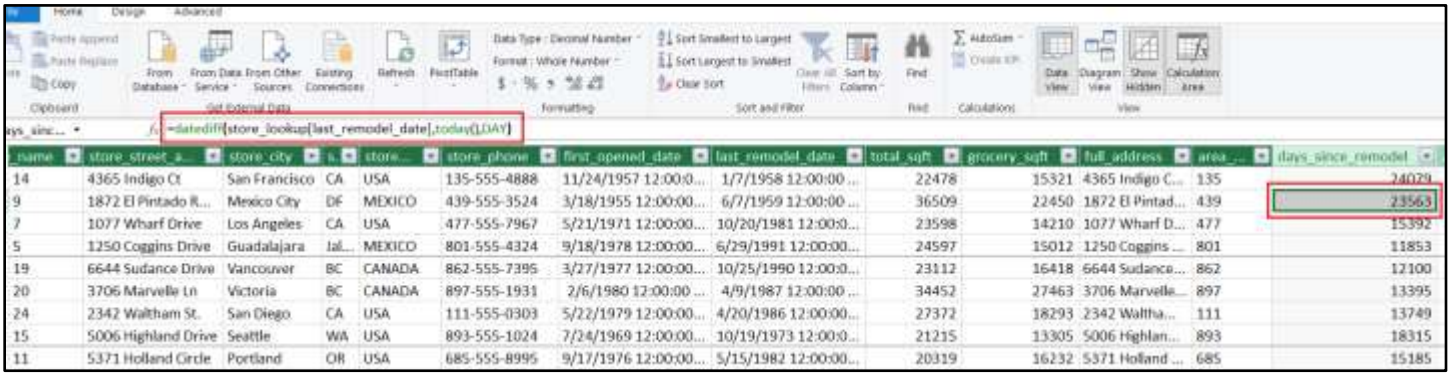
In my data model I used only the **One-To-Many cardinality** to create relationships between the Lookup Tables and Data Tables.

The advantage of using a Data Model to create a Power Pivot Table is that this enables us to use **multiple separate tables in a single Power Pivot Table**. If we did not use a Data Model then we must consolidate multiple tables into a single large table by using formulas like VLookup. This is time consuming and prone to errors.

Another advantage of using a Data Model to create a Power Pivot Table is that this enables us to create multiple tables from **various different data sources**. For example data from websites, PDF files, SQL Server or CSV files.

The Data Model I created was based on a **Snowflake Schema** (as shown below).





name	store_street_a...	store_city	s...	store...	store_phone	first_opened_date	last_remodel_date	total_sqft	grocery_sqft	full_address	area...	days_since_remodel
14	4365 Indigo Ct	San Francisco	CA	USA	135-555-4888	11/24/1957 12:00:0...	1/7/1958 12:00:00 ...	22478	15321	4365 Indigo C...	135	24029
9	1872 El Pintado R...	Mexico City	DF	MEXICO	439-555-3524	3/18/1955 12:00:00...	6/7/1959 12:00:00 ...	36509	22450	1872 El Pintad...	439	23563
7	1077 Wharf Drive	Los Angeles	CA	USA	477-555-7967	5/21/1971 12:00:00...	10/20/1981 12:00:0...	23598	14210	1077 Wharf D...	477	15392
5	1250 Coggins Drive	Guadalajara	Jal...	MEXICO	801-555-4324	9/18/1978 12:00:00...	6/29/1991 12:00:00...	24597	15012	1250 Coggins ...	801	11853
19	6644 Sudance Drive	Vancouver	BC	CANADA	862-555-7395	3/27/1977 12:00:00...	10/25/1990 12:00:0...	23112	16418	6644 Sudance...	862	12100
20	3706 Marvelle Ln	Victoria	BC	CANADA	897-555-1931	2/6/1980 12:00:00 ...	4/9/1987 12:00:00 ...	34452	27463	3706 Marvelle...	897	13395
24	2342 Waltham St.	San Diego	CA	USA	111-555-0303	5/22/1979 12:00:00...	4/20/1986 12:00:00...	27372	18293	2342 Waltha...	111	13749
15	5006 Highland Drive	Seattle	WA	USA	893-555-1024	7/24/1969 12:00:00...	10/19/1973 12:00:0...	21215	13305	5006 Highlan...	893	18315
11	5371 Holland Circle	Portland	OR	USA	685-555-8995	9/17/1976 12:00:00...	5/15/1982 12:00:00...	20319	16232	5371 Holland ...	685	15185

POWER PIVOT TABLE

A **Power Pivot Table** is the same as a normal **Pivot Table** except that the **source data** for the Power Pivot Table is the **Data Model** whereas the source data for a normal Pivot Table is a **single table** on the Excel worksheet.

The **advantages** of using a **Power Pivot Table** instead of a normal Pivot Table are:

- we can upload **large datasets** for millions of rows into a data model whereas a worksheet can only have a limit of 1 million rows.

- **data models** used in Power Pivot Tables can use **tables from various different data sources (web, pdf, SQL)** which can be combined through **table relationships** to enable better data analysis. If we instead used a single Excel worksheet as a data source (as is done for a normal Pivot Table) then we need to use VLookup and Index Match to stitch the various data sources together into a single table.

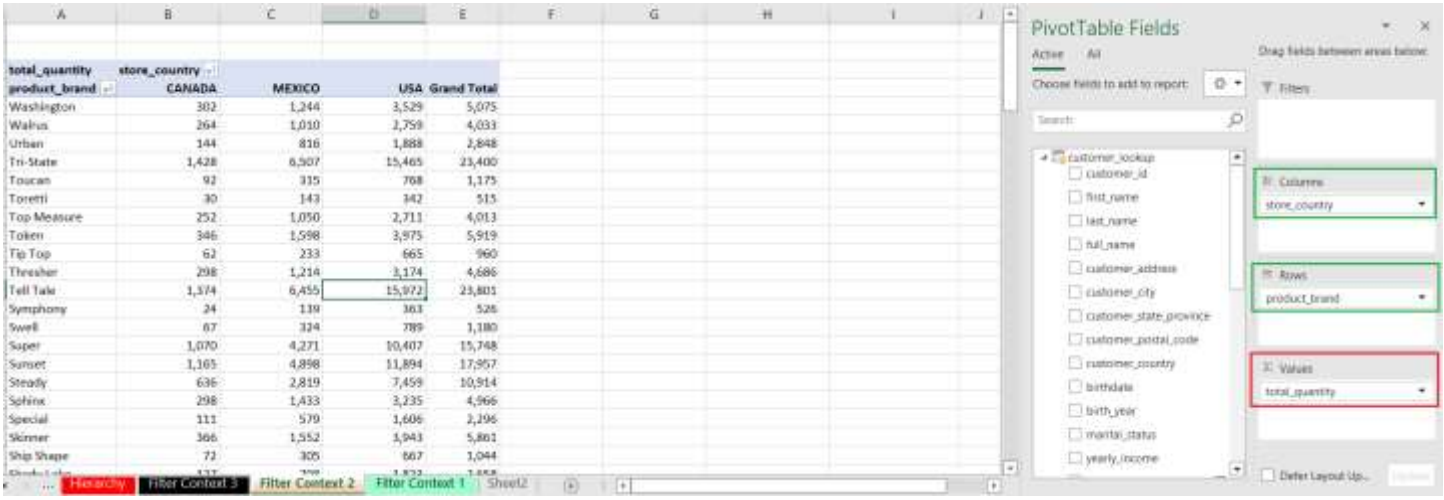
- we can use **DAX** (data analysis expressions) formulas to create **Measures** and **Calculated Columns** in Power Pivot Tables and Data Models. In comparison a normal Pivot Tables use Calculated Fields which are not as powerful.

In this Data Model I created a **Calculated Column** in table Customer_Lookup with the following **DAX formula**:
= DATEDIFF (customer_lookup[birthdate],today(),YEAR)
This calculated the age of each customer by using the **function DATEDIFF()**

In the Power Pivot Table I created a **Measure** in the table Transactions with the following **DAX formula**:
Total_Quantity: = SUM(Transactions[quantity])
This calculated the total quantity of products sold by using the **function SUM()**

Filter context refers to the fields have been dragged into the Power Pivot Table **Panes for Rows, Columns, Filters & Slicers**. They act as a set of "coordinates" which will impact the calculation of the field in the **Pane Values**. **Measures** are based on Filter Context.

In the Excel file **Supermarket Data Model.xlsx** I created examples of Power Pivot Tables with Filter Context in tabs named **"Filter Context 1", "Filter Context 2"** and **"Filter Context 3"**



DAX

DAX stands for **Data Analysis Expressions** and is the formula language used to create metrics to analyze data and provide insights.

The DAX functions are used to create **Calculated Columns** in tables of the Data Model, and **Measures** in the Power Pivot Table. The Calculated Columns are used to filter data whereas Measures are used to aggregate data values (for example SUM, COUNT, AVERAGE).

The DAX functions work with relational datababases and enable us to create complicated metrics which simple formulas are not able to achieve. DAX can be used to nest several Measures to create powerful **Measure Trees**.

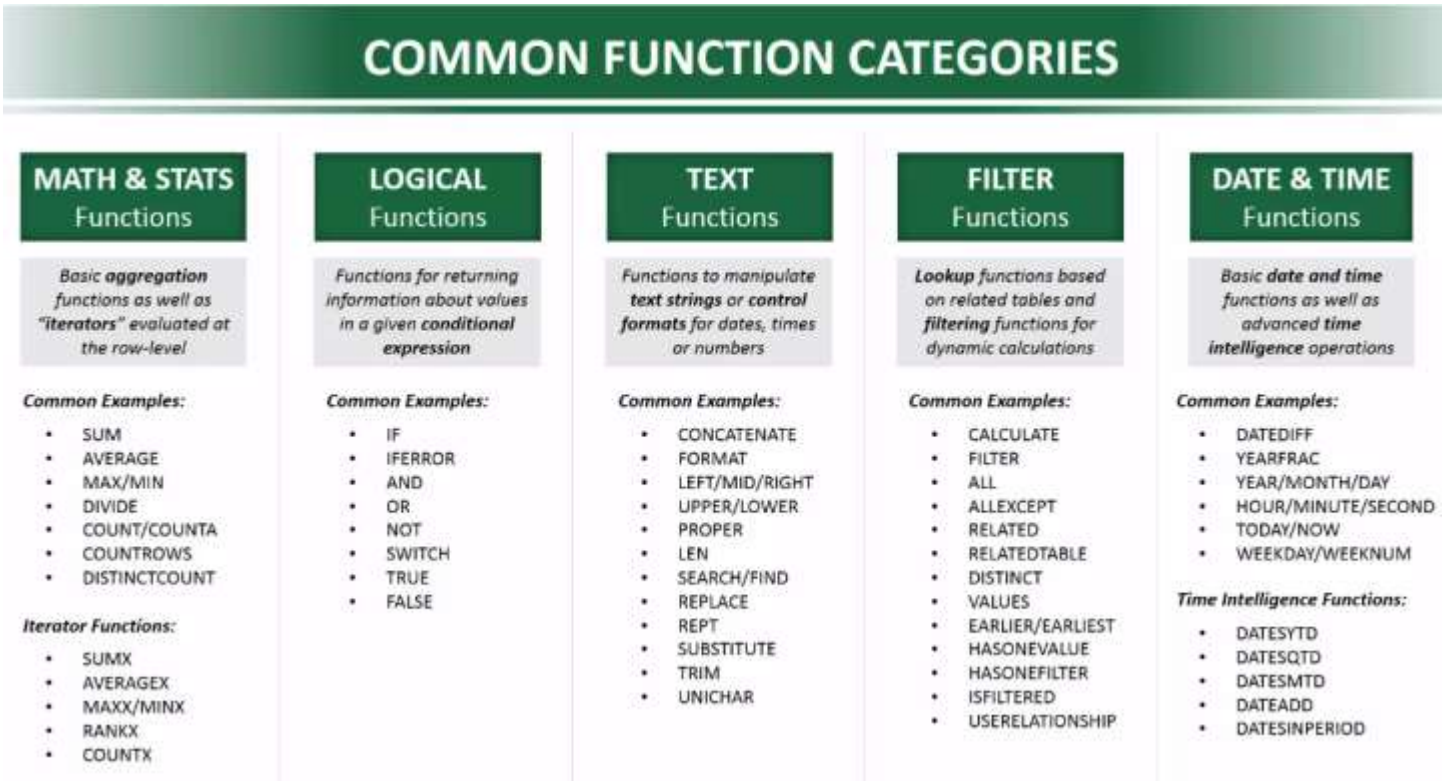
Some of the DAX function categories are shown in the diagram below. These include:

- Time Intelligence Functions - DATESYTD( ), DATESMTD( ), DATESQTD( ), DATESINPERIOD( )
- Iterator Functions - SUMX( ), COUNTX( ), RANKX( )
- Filter Functions - CALCULATE( ), FILTER( ), ALL( ), RELATED( ), DISTINCT( )
- Logical Functions - IF( ), NOT( ), AND( ), OR( )
- Statistical Functions - SUM( ), COUNT( ), DISTINCTCOUNT( ), COUNTROWS( )
- Text Functions - CONCATENATE( ), LEFT/RIGHT( ), UPPER/LOWER( ), REPLACE( )
- Date & Time Functions - DATEDIFF( ), YEAR/MONTH/DAY( ), WEEKDAY/WEEKNUM( )

For my project I used DAX Measures to create a **Disconnected Slicer** for a Power Pivot Table. The functions used were CALCULATE( ) with a nested FILTER ( ) function.

I used **RANKX( )** the **Iterator X function** to create a Measure which **ranks** the field **product_name** based on another field **Total Revenue Measure**. This shows the **product names which generated the highest revenue by rank order.**

In addition I used **Time Intelligence functions** such as **DATESYTD( ), DATESMTD( )** and **DATESQTD( )** to **display trends in data values over a time period** such as every day in the year 1998.



price_threshold

1 | 2 | 3 | 4

product_brand	Count Transactions	Threshold Selected	Transactions in Price Threshold
ADJ	198	2	
Akron	356	2	356
American	2,384	2	653
Amigo	326	2	
Applause	355	2	
Atomic	1,345	2	656
BBB Best	5,254	2	3249
Best	714	2	339
Best Choice	6,000	2	2164
Better	4,073	2	2449
Big City	355	2	355
Big Time	5,797	2	2578
Bird Call	3,224	2	1818
Black Tie	374	2	374
Blue Label	3,665	2	1852
Blue Medal	808	2	167

product_brand	product_name	product_retail_price	total_quantity	Total Revenue (Measure)	Product Rank (by Revenue)
ADJ	ADJ Rosy Sunglasses	2.76	620	$1,711	235
Akron	Akron Eyeglass Screw	1.76	581	$1,023	852
	Akron City Map	1.74	529	$920	963
American	American Sliced Turke	3.17	548	$1,737	219
	American Sliced Ham	2.76	599	$1,653	266
	American Roasted Ch	2.97	531	$1,577	329
	American Pimento Lo	2.76	558	$1,540	360
	American Turkey Hot	2.74	544	$1,491	394
	American Low Fat Bol	2.87	511	$1,467	412
	American Corned Bee	2.65	512	$1,357	541
	American Low Fat Col	2.27	563	$1,278	621
	American Chicken Ho	2.52	486	$1,225	662
	American Foot-Long H	2.14	514	$1,100	785
	American Potato Sala	1.55	477	$739	1,179
	American Cole Slaw	0.89	544	$484	1,353
	American Beef Bologr	0.78	531	$414	1,411
	American Sliced Chick	0.59	505	$298	1,520
Amigo	Amigo Scallops	2.79	563	$1,571	341
	Amigo Lox	2.96	449	$1,329	570

Year 1998

date	total_quantity	YTD total_quantity	MTD total_quantity	QTD total_quantity
5/18/1998	204	209,767	27,640	72,689
5/19/1998	1,784	211,551	29,424	74,473
5/20/1998	1,306	212,857	30,730	75,779
5/21/1998	2,473	215,330	33,203	78,252
5/22/1998	535	215,865	33,738	78,787
5/23/1998	1,026	216,891	34,764	79,813
5/24/1998	2,044	218,935	36,808	81,857
5/25/1998	1,306	220,241	38,114	83,163
5/26/1998	1,629	221,870	39,743	84,792
5/27/1998	2,583	224,453	42,326	87,375
5/28/1998	822	225,275	43,148	88,197
5/29/1998	1,906	227,181	45,054	90,103
5/30/1998	31	227,212	45,085	90,134
5/31/1998		227,212	45,085	90,134
6/1/1998	810	228,022	810	90,944
6/2/1998	1,405	229,427	2,215	92,349
6/3/1998	1,811	231,238	4,026	94,160
6/4/1998	378	231,616	4,404	94,538
6/5/1998	1,629	233,245	6,033	96,167
6/6/1998	2,562	235,807	8,595	98,729

DASHBOARD & SLICERS

I created a **Dashboard** to enable data visualization. This dashboard shows **sales analysis by product brands, customer demographics, geographic location and seasons (in months).**

The **Tab "Dashboard & Slicers"** displays various charts which are linked to Power Pivot Tables. Also the **Slicers** enable users to filter and interact with the data in these charts. The data source for this project is from the course "Microsoft Excel Business Intelligence w/ Power Query & DAX" by Maven Analytics.

The **Bar Chart** shows sales quantity by product brands

The **Pie Chart** shows sales quantity by cities where customers are located.

The **Stacked Bar Chart** shows sales quantity by customers categorised into income levels and their city location.

The **Line Chart** over time shows the sales quantity by month for the years 1997 and 1998. This displays the seasons which had the highest sales quantities.

The **Slicers** are connected to all these charts in the dashboard. The Slicers enable users to filter if the customers are home owners or if the customers have children in their family. Another Slicer filters the dashboard by product name. Also the **Timeline Slicer** can filter the charts by months in 1997 and 1998.