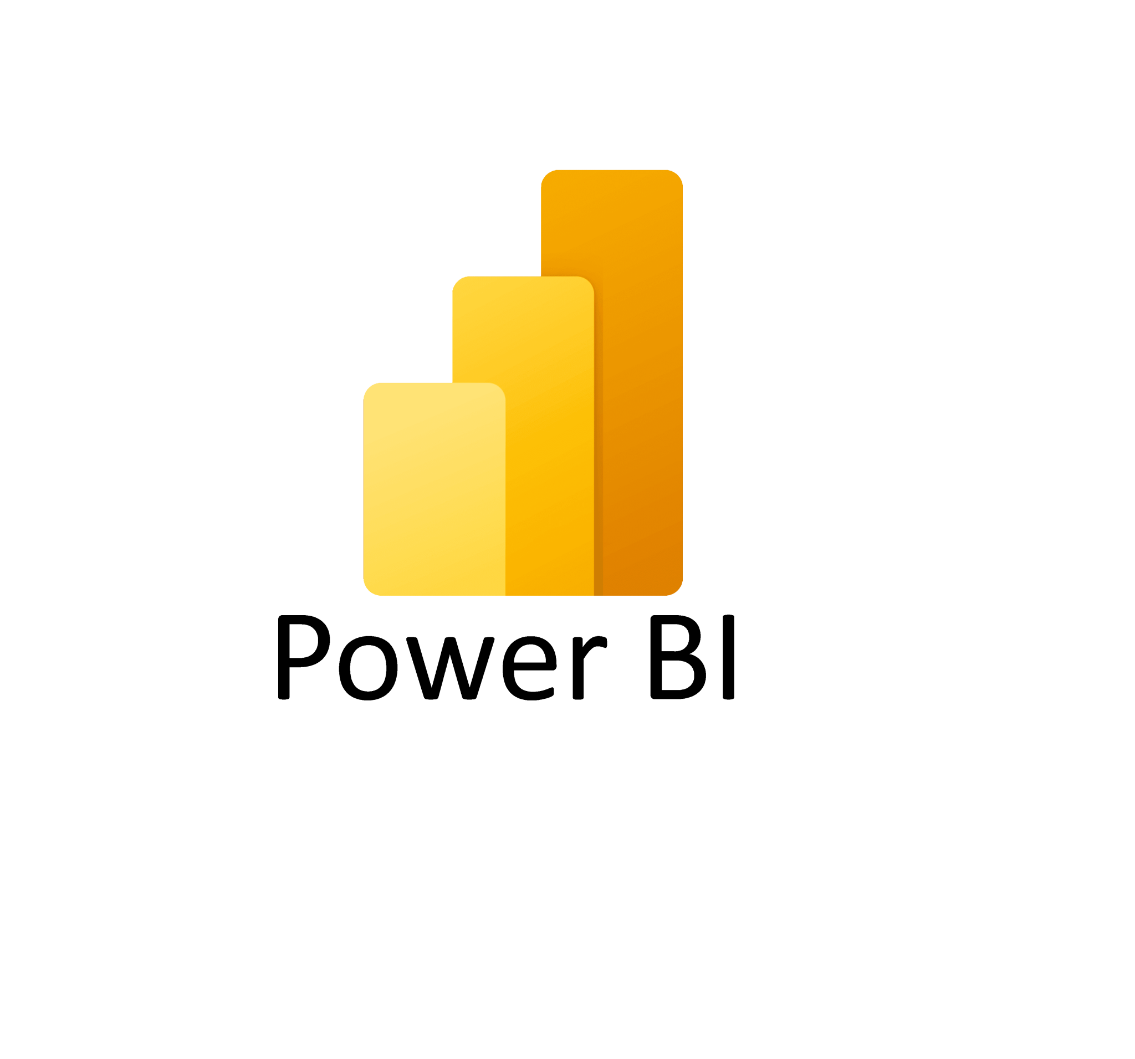
A logo for a video game

AI-generated content may be incorrect.

**Powered by Bokamoso Coders: Microsoft Power BI**

****

PARTICIPANT HANDBOOK

INSTRUCTOR-LED TRAINING

Course Version: 1

Course Duration: 4 hours

Material Number: 50160449

# Contents

Course Overview

|  |  |  |
| --- | --- | --- |
| 1 | Unit 1: | Course Introduction |
|  |  |  |
|  |  |  |
| 11 |  | Welcome & Objectives |
| 13 |  | Why Power BI? (business use case) |
| 53 | Unit 2: | Set-up and Installation |
|  |  |  |
| 55 |  | Creating a Microsoft PowerBI Account |
| 61 |  | Installing PowerBI |
| 66 |  | PowerBI walkthrough |
| 111 | Unit 3: | Connecting and shaping Data |
|  |  |  |
| 112 |  | Lesson: Importing different Files  Lesson: Connecting to different databases.  Lesson: Import data from folders |
| 115 |  |  |
| 131 | Unit 5: | Configuring Charts & Tables |
|  |  |  |
| 133 |  | Lesson: Configuring Tables |
| 138 |  | Lesson: Configuring Charts |
| 147 |  | Exercise 5: Configure Charts and Text Widgets |
| 167 |  | Lesson: Using Conditional Formatting |
| 169 |  | Exercise 6: Use Conditional Formatting (Optional) |
| 185 | Unit 6: | Sorting & Filtering |
|  |  |  |
| 187 |  | Lesson: Sorting Data in Stories |
| 188 |  | Lesson: Filtering Data in Stories |
| 197 |  | Exercise 7: Filtering |
| 217 |  | Exercise 8: Use Measure and Time Input Controls for a Time |
|  |  | Series and Scatterplot |

# Course Overview

TARGET AUDIENCE

This course is intended for the following audiences:

* Application Consultant
* Technology Consultant
* Business User
* Business Analyst

**UNIT 1**

**Getting Started with PowerBi**

### Lesson 1

Getting Started with PowerBI 3

UNIT OBJECTIVES

* Introduction to Power BI and its Ecosystem
* Explore the Power BI Interface
* Discuss Real-World Applications

Unit 1

Lesson 1

## What is Power BI?

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* What Power BI is and how it fits with Microsoft Fabric
* The key differences between Power BI Desktop and Power BI service
* How to get started with Power BI step-by-step

### **Introducing PowerBI**

Power BI is Microsoft’s Business Intelligence tool used for connecting, transforming,

and visualizing data. The purpose is to turn raw data into meaningful insights through

interactive dashboards and reports.

**Real-World Examples:**

* Retail: Sales dashboards showing performance by region.
* HR: Attrition dashboards tracking employee trends.
* Finance: Budget vs. Actual spend monitoring.

 **Power BI and Microsoft Fabric**

**What is Microsoft Fabric?**

Microsoft Fabric is an end-to-end analytics platform that brings together all the data tools an organization needs — from data movement, storage, and engineering to real-time analytics and BI reporting.

**Key Components of Fabric**

* Fabric unifies different workloads under one umbrella:
* **Data Factory** → For data integration and pipelines.
* **Synapse Data Engineering** → For big data analytics with Spark.
* **Synapse Data Science** → For advanced ML & AI experiments.
* **Synapse Data Warehousing** → For enterprise-scale relational analytics.
* **Real-Time Analytics** → For streaming data insights.
* **Power BI** → For visualization, storytelling, and sharing insights.

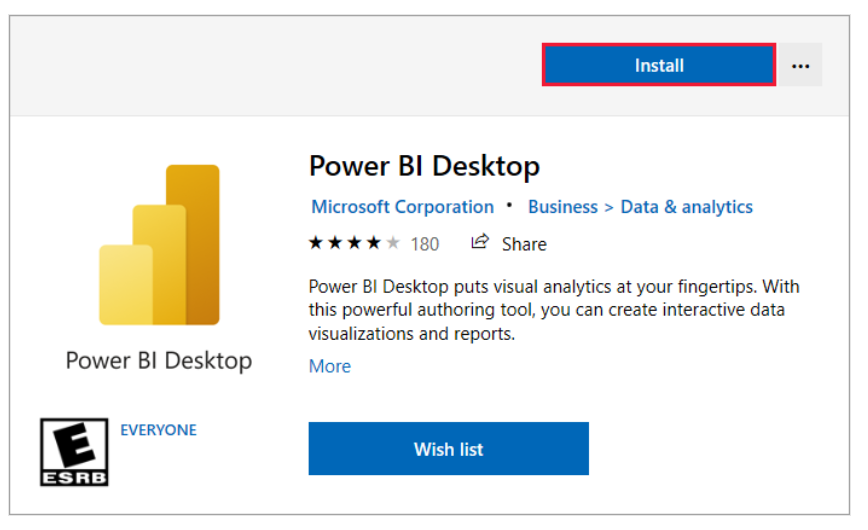
Power BI is the front door of Microsoft Fabric — its what business users see, but under the hood, Fabric connects everything else.

**Why this matters:**

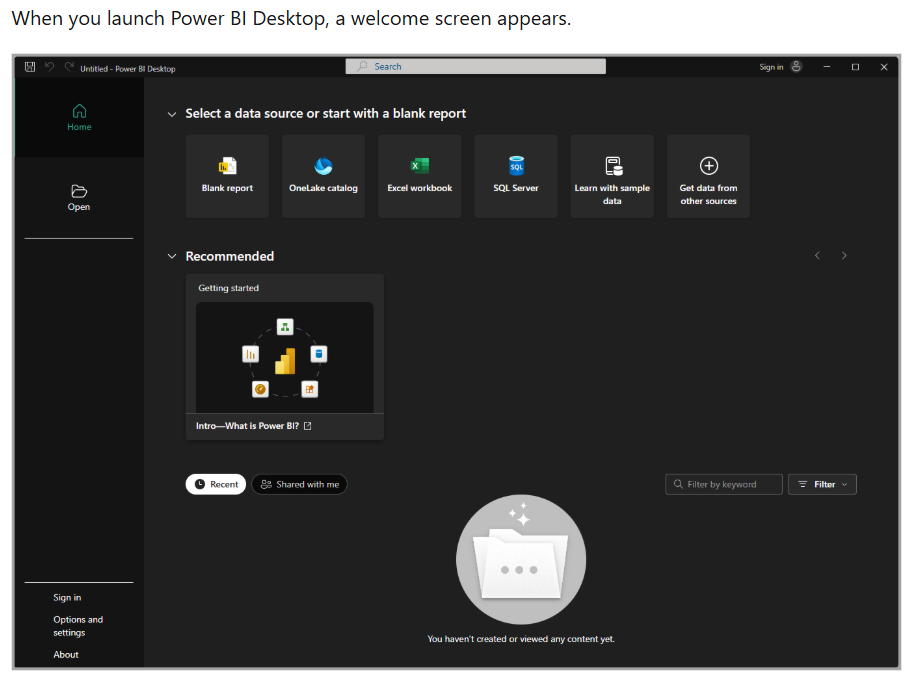
* Power BI isn’t just a reporting tool anymore — it’s part of a much larger data ecosystem.
* With Fabric, your reports can connect directly to OneLake (the unified data lake of Fabric) instead of depending only on Excel, SQL, or CSV files.
* This makes solutions scalable, cloud-native, and enterprise-ready.

**Installing PowerBI**

After you've landed on the **Power BI Desktop** page of the Microsoft Store, select **Install**



When you launch this Screen appears



Connecting to PowerBI datasources

**UNIT 2**

**Connecting to PowerBI data sources**

Lesson 1

Introductions to data Connections 11

Lesson 2

Connecting to File-based data sources 13

Lesson 3

Connecting to Databases 21

Lesson 4

Connecting to Cloud & Online Services 45

Lesson 5

Connecting to Web & API’s 45

UNIT OBJECTIVES

* Explain the different categories of data sources available in Power BI.
* Connect Power BI Desktop to file-based data sources such as Excel, CSV, JSON, and PDF.
* Connect to Microsoft Fabric using Lakehouse, Warehouse, and One Lake Data Hub.
* Apply best practices when connecting to data, including selecting relevant tables, setting data types, and handling credentials securely.

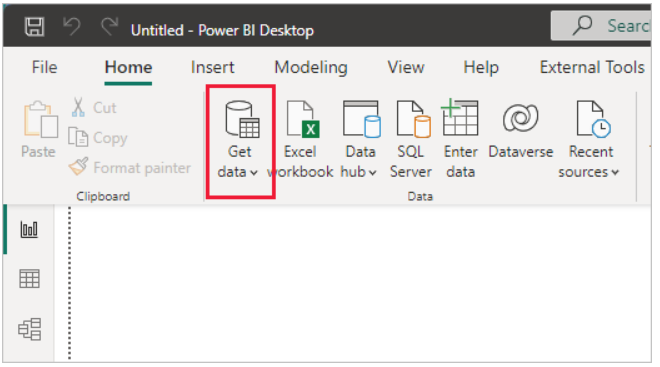
1. **What is a Data Source in Power BI?**

In Power BI, a data source is the location from which your report retrieves its data. Data sources can be:

* + - * A file on your computer (Excel, CSV, JSON)
      * A database (SQL Server, Oracle, MySQL, etc.)
      * A cloud service (Azure, SharePoint, OneDrive, Salesforce, etc.)
      * An online feed (Web, OData, APIs)
      * Microsoft Fabric Lakehouses and Warehouses

1. **The “Get Data” Button**

All connections start with the Get Data button on the Home ribbon in Power BI Desktop.



Since this data source is an Excel file, select **Excel** from the **Get Data** window, then select the **Connect** button.

1. **Common Connection Scenarios**

**a) Files**

* Useful for individual analysis or small projects
* Examples: Excel workbooks, CSV exports from a system

**b) Databases**

* Used for large, structured datasets stored in systems like SQL Server or Oracle
* Offers options for Import, DirectQuery, or Live Connection

**c) Cloud & Online Services**

* Connect directly to services like SharePoint, Dynamics, or Salesforce
* Ideal for teams and organizations storing data in the cloud

**d) Microsoft Fabric**

* Power BI integrates natively with **OneLake** in Fabric
* You can connect to Lakehouses and Warehouses for enterprise-scale analytics

**e) Web & APIs**

* Import public data tables from websites (e.g., population stats, stock prices)
* Advanced users can connect to APIs via OData feeds or custom queries

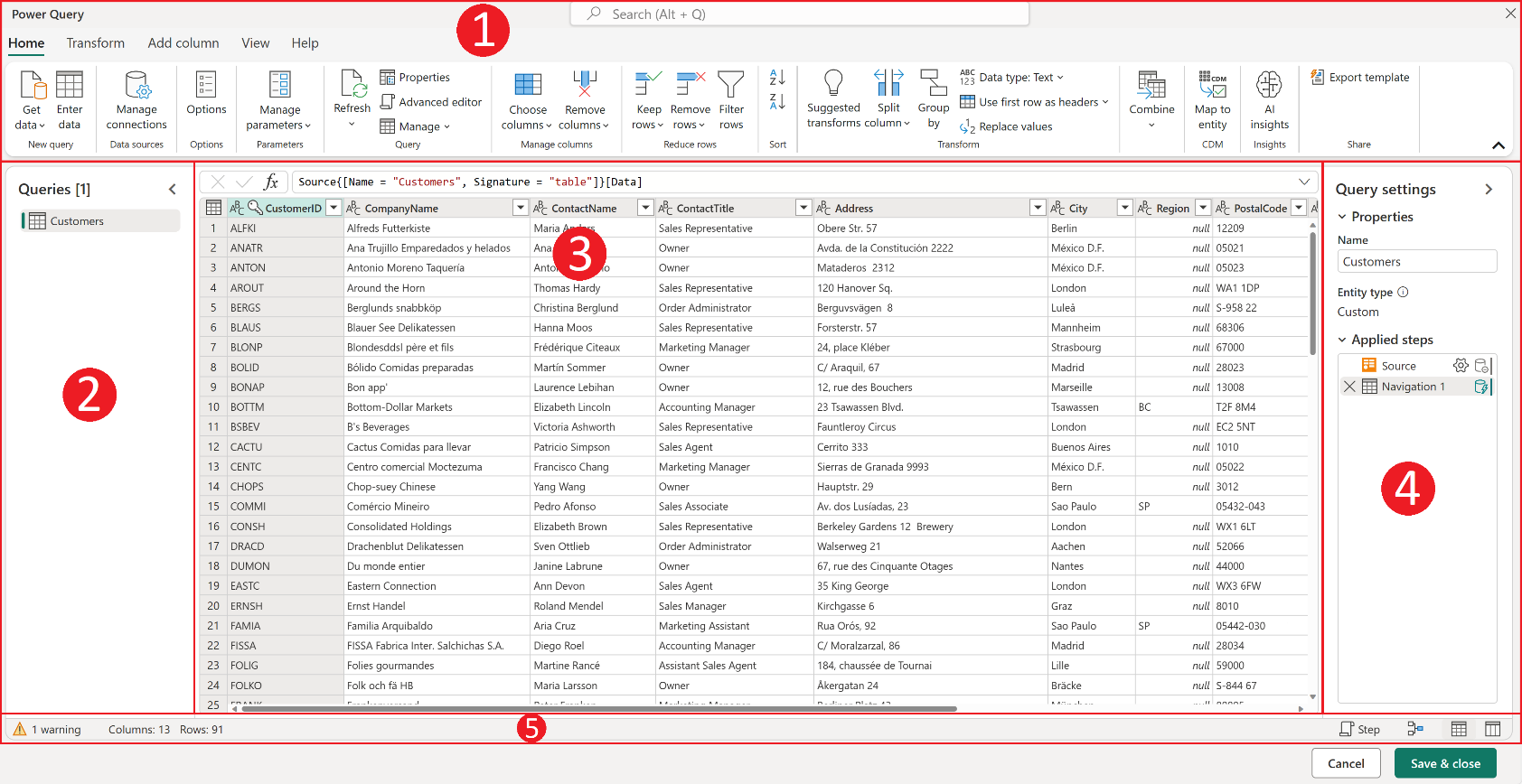
### **What is Power Query?**

Power Query is a tool inside Power BI used to clean, shape, and transform data before analysis. Think of it as a “data cleaning kitchen” before serving your dashboard.

**Opening Power Query**

* + - * Load the dataset: Apocalypse Food Prep → Purchase Tracker sheet.
* In Power BI Desktop → Home → Get Data → Excel → Select file → Load.
* Click Transform Data → Power Query Editor opens**.**

**Exploring Power Query Editor**

The Power Query editor represents the Power Query user interface. In this user interface, you can add or modify queries, manage queries by grouping or adding descriptions to query steps, or visualize your queries and their

### **Exploring the Power Query Editor**

* Ribbon → tools for transforming data.
* Queries Pane → shows "Purchase Tracker".
* Data Preview Grid → shows rows from the Excel sheet.
* Applied Steps Pane → logs every action (like a recipe).

**2. Importing & Transforming Data**

**Basic Cleanup**

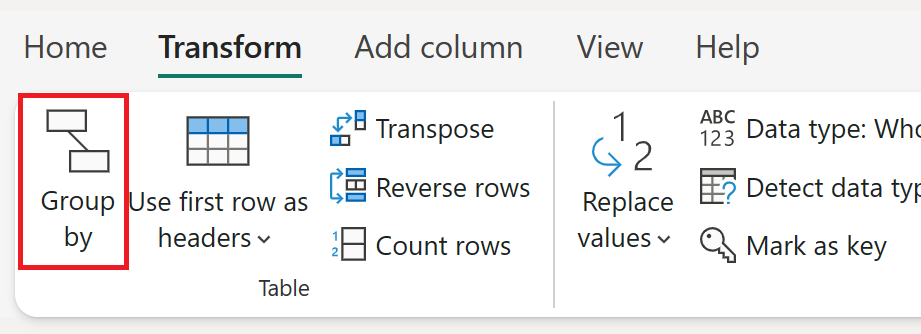
Home → Remove Top Rows → Delete the blank/title rows at the top.

A screenshot of a computer

AI-generated content may be incorrect.

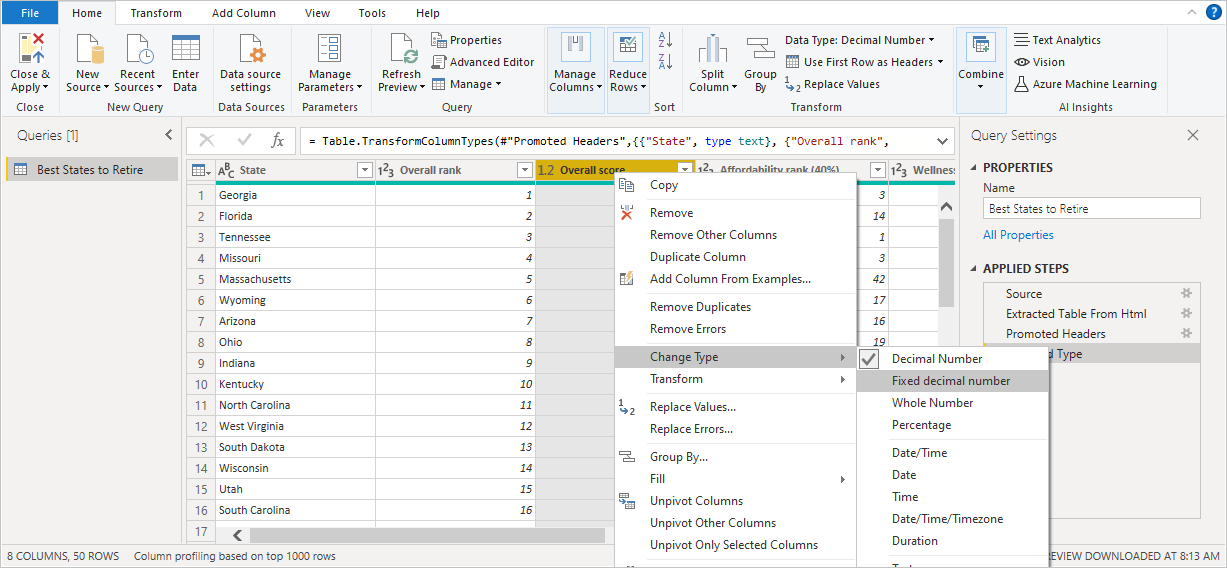
**Steps to use the First Row as Headers**

* In Power Query Editor, go to the Home tab on the ribbon.
* Click Use First Row as Headers under the Transform group.
* The first row will move into the header position, making your dataset cleaner and easier to work with.



**Steps to Change a Column’s Data type**

* 1. In the Power Query Editor, select the column you want to adjust.
* In our dataset, for example, the Product Cost values are numeric.
  1. Go to the Data Type dropdown (on the Home tab, top menu).
  2. Choose the correct data type from the list. Common ones include:
* Decimal Number → for prices, amounts, or values with decimals.
* Whole Number → for counts or quantities.
* Date → for date columns like Date\_Purchased.
* Text → for product names or store names.
  1. Once applied, you’ll see a new step added in the Applied Steps pane called Changed Type.



**3. Changing Column Types in Power Query**

Every column in your dataset needs the correct data type so Power BI knows how to handle it. For example:

* Dates must be stored as Date to create timelines.
* Numbers must be Whole Numbers or Decimals to allow calculations.
* Text fields (like Store or Product Name) must be text, so they’re not treated as numbers.

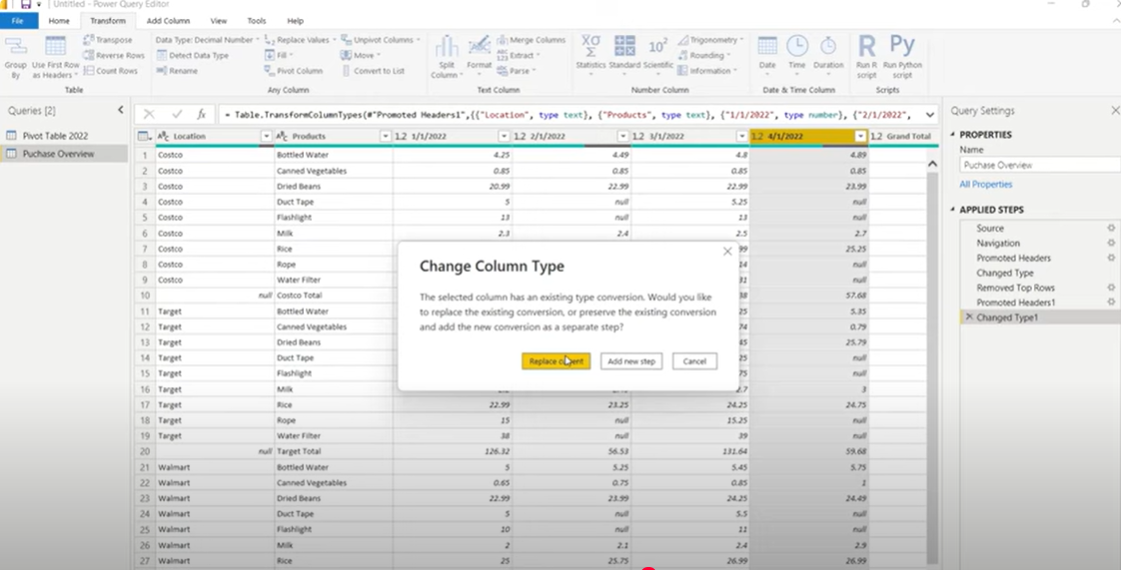
If the data type is wrong, you may get errors or misleading results in your visuals.

**Steps to Change a Column’s Data Type:**

1. Select the column you want to update.

In our example, we’re updating Product Cost columns to Decimal Number.

1. On the Home tab, click the **Data Type dropdown.**
2. Select the correct type (Date, Decimal, Whole Number, Text, etc.).



1. Filtering Data in Power Query

Filtering helps you focus only on the rows you need for analysis. This is especially useful when:

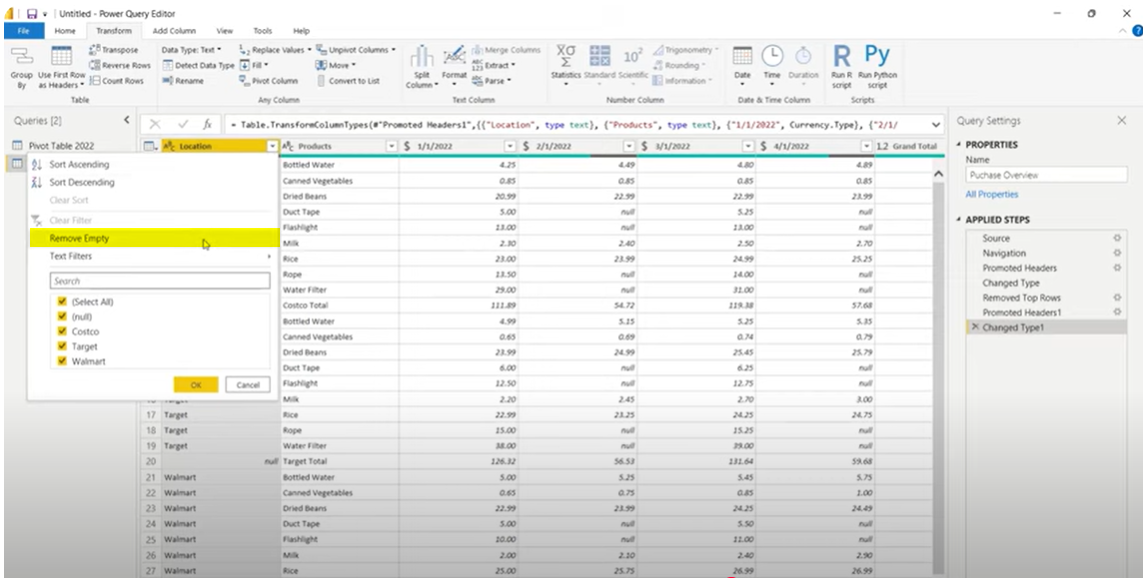
* You want to remove irrelevant values (e.g., Grand Total rows).
* You want to work with data from specific categories (e.g., only one store).
* You want to clean out blank or null entries.

Steps to Filter Data in Power Query:

1. Select the column you want to filter.

In this example, the column is Location.

1. Click the dropdown arrow next to the column name.
2. A filter menu appears (like in Excel). From here, you can:
   * + - * Select/deselect items.
         * Remove blanks (null values).
         * Apply text filters (e.g., contains, begins with)
3. Click OK to apply the filter.



Text Filter: Does Not Contain

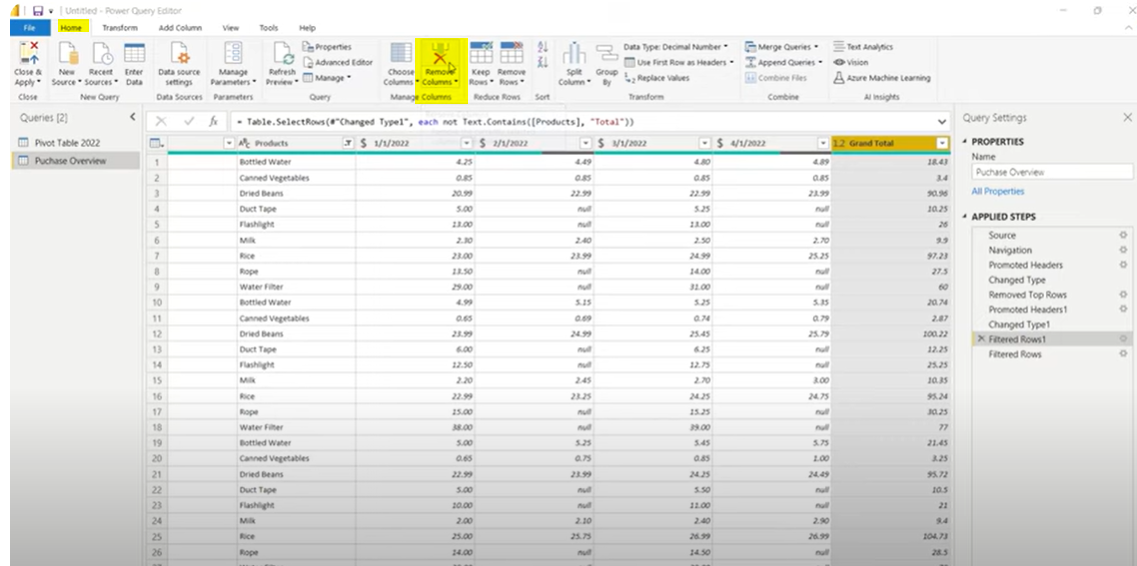
A screenshot of a computer

AI-generated content may be incorrect.

**Removing Unnecessary Columns**

After cleaning the rows, the next step is to simplify the dataset by keeping only the columns you need.

* Select the unwanted columns (e.g., Grand Total).
* On the Home tab, click Manage Columns → Remove Columns.



**Pivoting and Unpivoting Columns**

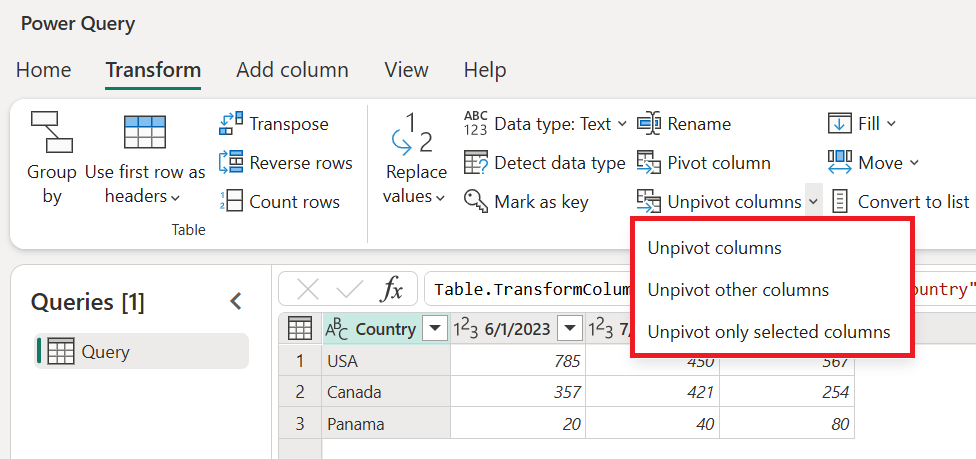
* + - Unpivot → turns columns into rows (useful when dates or categories are spread across columns, but you need them stacked in one column)
    - Pivot → turns rows into columns (useful for creating summary tables).

**Unpivoting Columns**

1. Select the columns you want to keep as identifiers (Location and Products).
2. On the Transform tab, click Unpivot Columns.
3. The date columns are transformed into two new fields:

* Attribute → holds the original column names (dates).
* Value → holds the corresponding product cost.

Now your dataset is tidy: each row represents a purchase with Location, Product, Date, and Value.



**Renaming Columns for Clarity**

To make the dataset easier to understand, rename $ Value → **Product Cost** and **Location** → **Store.** Clear

names improve readability and ensure consistency when building visuals in Power BI.

Unit 4

Lesson 1

## How to Create and Manage Relationships in PowerBI

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Understand what relationships are in Power BI.
* Create and edit relationships between tables.
* Manage cardinality and cross-filtering settings.

### Dataset: Apocalypse Food prep- Relationship Tutorial

**What is a Relationship?**

* A connection between two tables, based on a common column (key).
* Example: Sales table (ProductID) linked to Products table (ProductID).

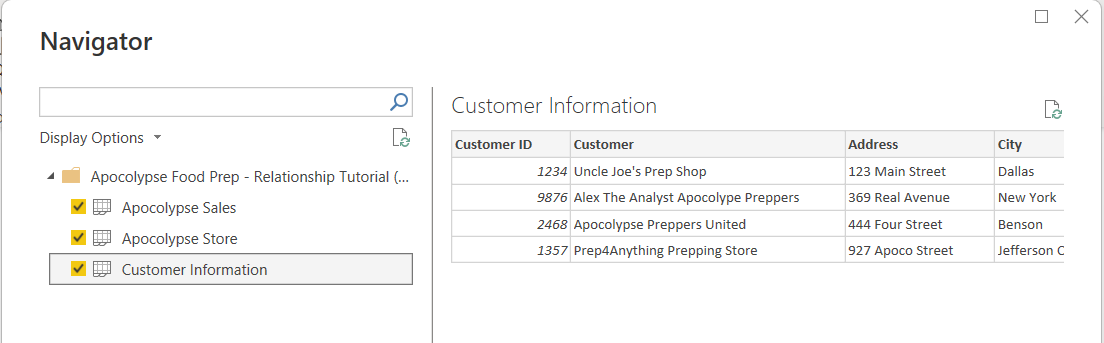
**Why Relationships Matter**

* They allow you to build reports that combine fields from different tables.
* Prevent duplicate data and keep your model efficient.

### **Creating Relationships**

**Step 1: Import Data**

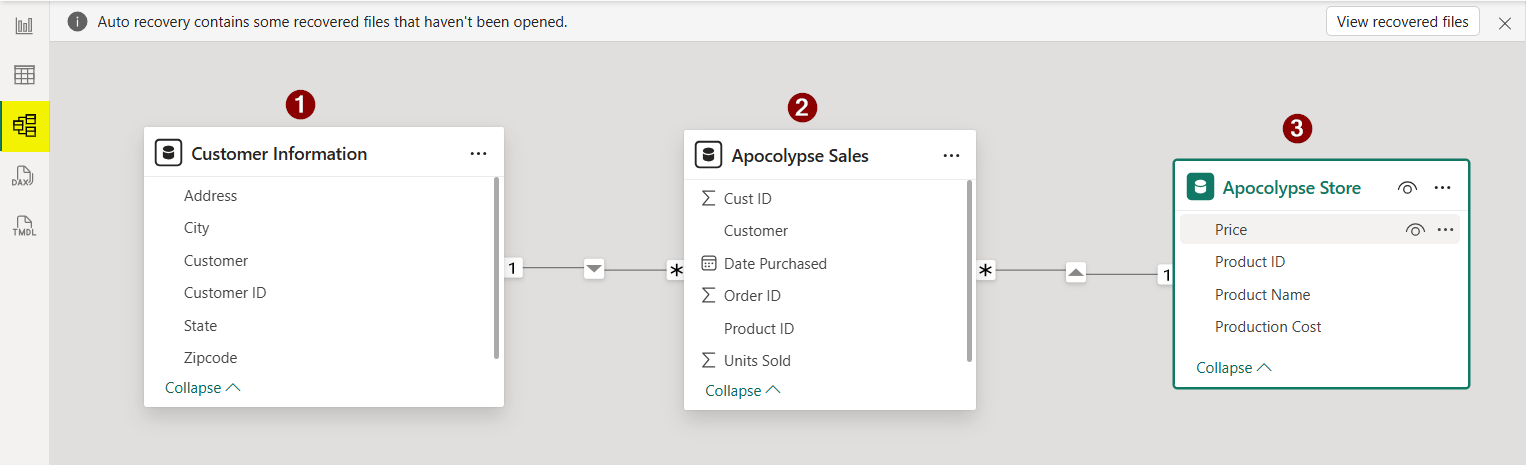
* From the Home tab → Get Data → Excel, select your file.
* In the **Navigator window**, tick all three sheets
* Customer Information
* Apocolypse Sales
* Apocolypse Store
* Use Click Load to bring them into Power BI.



**Step 2: Switch to Model View**

* On the left sidebar, click the Model icon to open the model view.
* Arrange the tables so they are easy to see:

1. Customer Information on the left.
2. Apocolypse Sales in the center.
3. Apocolypse Store on the right.



**Step 3: Create Relationships**

Power BI often detects relationships automatically, but you can also create them manually.

The model now shows:

* One-to-Many (1:\*) between Customer Information → Apocolypse Sales.
* One-to-Many (1:\*) between Apocolypse Store → Apocolypse Sales.

This is a **star schema:** Sales is the fact table in the middle, connected to Customer and Store dimension tables.

**Tip for learners:** Always check that relationships match the real-world logic (e.g., one customer can place many sales, one store product can appear in many sales)

**Editing Relationships and Understanding Cardinality**

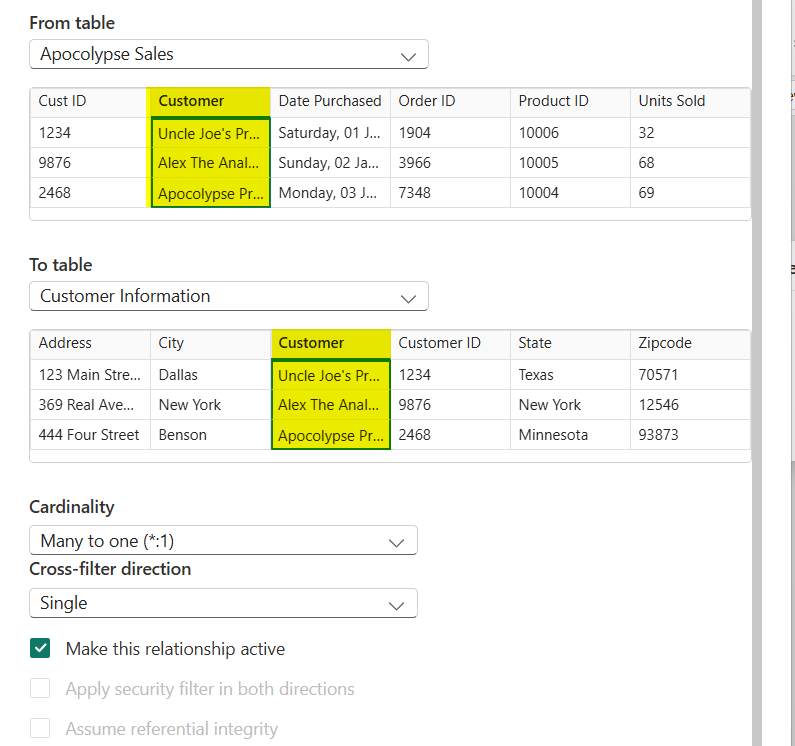
When Power BI detects relationships, it sometimes selects the wrong column to join tables. In this case, Power BI automatically used the Customer field to link the Apocolypse Sales and Customer Information tables. However, using names is risky because they can repeat or be misspelled.

**Step 1: Adjusting the Relationship Fields**

* Open the relationship settings by double-clicking the line connecting the two tables.
* Change the relationship so it uses **Customer ID** instead of Customer Name.
* Select **Cust ID** in Apocolypse Sales and **Customer ID** in Customer Information.
* Click **OK** to apply.

This ensures the relationship is based on a unique identifier (Customer ID), which is more reliable for analysis.

This ensures the relationship is based on a unique identifier (Customer ID), which is more reliable for analysis.



**Understanding Cardinality**

Cardinality defines how data in one table relates to data in another.

1. **One-to-Many (1:\*)** → Most common.
   * + - Example: One customer can make many sales.

* Power BI uses this when linking dimension tables (like Customer Information) to fact tables (like Apocolypse Sales).

1. **One-to-One (1:1)** → Used when both tables have unique matching records.

* Example: Employee table linked to Badge table, one record each.

1. **Many-to-One (\*:1)** → The same as One-to-Many but viewed from the opposite direction (depending on which table you start from).

1. **Many-to-Many (:)** → Both tables contain duplicate values for the key.

* Usually avoided because it can cause ambiguous results.

💡 Tip: Use One-to-Many relationships wherever possible for clean, reliable data models.

A screenshot of a computer

AI-generated content may be incorrect.

Understanding Cross Filter Direction (Example: Sales and Store)

Cross-filtering determines how filters flow between related tables in Power BI — it defines whether one table can influence another when you create visuals.

**Single Direction (Recommended Default)**

In a Single cross-filter, the filter flows from the “one” side to the “many” side of the relationship.

Example: Filtering by Customer Name affects Sales, but not vice versa.

Both Directions (Bidirectional Filtering)

Filters move both ways — each table can affect the other.

Useful for lookup or bridge tables but can slow performance or create ambiguity.

***Example:*** *When to Use Bidirectional Filtering*

**Aggregated Reporting Across Both Tables**

Suppose you have visuals showing:

* **Total Sales by Store Location**, and
* **Stores by Total Sales Performance** (e.g., Top 5 and Bottom 5 stores).

In this case, you want **filters to flow both ways** — from Sales to Store and from Store to Sales — so that:

* When a user clicks on a store in the Store Performance chart, the Sales by Location chart updates automatically.
* Likewise, when a user selects a region or sales record in the Sales chart, it highlights or filters the related store.

This bidirectional setup keeps both visuals synchronized and provides a more interactive dashboard experience.

Unit 5

Lesson 1

## How to use Dax in PowerBI

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* + Understand what DAX is, why it’s used in Power BI, and how to create basic measures and calculated columns using DAX formulas.

### **What is Dax?**

DAX (Data Analysis Expressions) is the formula language used in Power BI, Power Pivot, and Analysis Services. It allows you to create new information from your existing data model — such as totals, averages, counts, percentages, and more complex metrics.

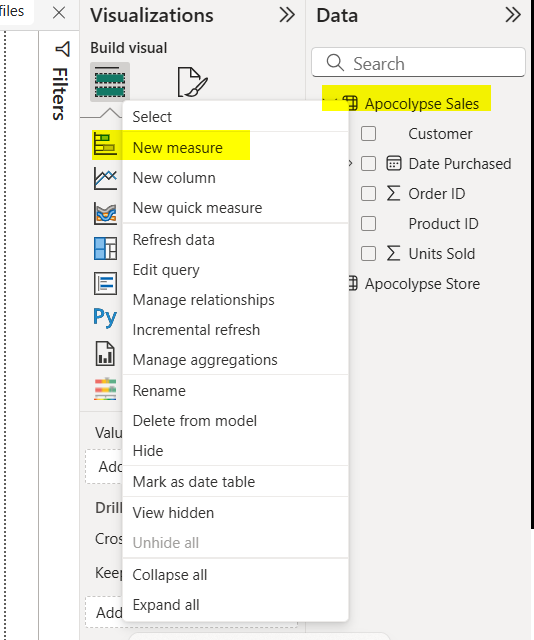
Think of DAX like Excel formulas but designed for relational data and filter contexts in Power BI.

**Basic DAX Functions Every Beginner Should Know**

|  |  |  |
| --- | --- | --- |
| Function | Description | Example |
| SUM() | Adds up all numbers in a column | SUM('Sales'[Revenue])` |
| AVERAGE() | Calculates the mean | AVERAGE('Sales'[Revenue]) |
| COUNT () | Counts number of rows with values | COUNT ('Sales'[Order ID]) |
| DISTINCTCOUNT() | Counts unique values | DISTINCTCOUNT('Sales'[Customer ID]) |
| MIN() / MAX() | Finds the smallest or largest value | MAX('Sales'[Price]) |
| DIVIDE() | Safe division (avoids divide by zero error) | `DIVIDE([Total Sales], [Total Orders])` |
|  |  |  |

**Step 1: Creating a New Measure**

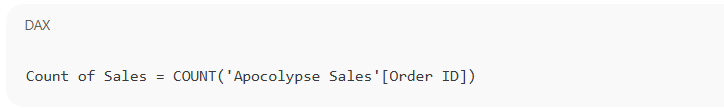
* 1. **Load the Dataset**
* Make sure the Apocolypse Sales and Apocolypse Store tables are already loaded into Power BI.
  1. **Open the Report View**
* On the left sidebar, click the Report icon (the one with the bar chart).
* This is where you build your visuals and create DAX calculations.
  1. **Select the Table for the Measure**
* In the Data pane, right-click on Apocolypse Sales.
* Choose New Measure from the dropdown menu (as shown in the screenshot).
  1. **A Formula Bar Appears**
* At the top of the report canvas, you’ll now see a formula bar.
* This is where you’ll type DAX formulas (Data Analysis Expressions) to create calculated metrics like totals, averages, and percentages.



**A screenshot of a computer

AI-generated content may be incorrect.**

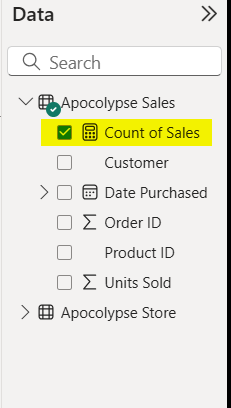
**Step 2: Writing Your First DAX Measure**



1. Press **Enter** to create the measure.

* Power BI will validate your formula and automatically add the new measure to the **Apocolypse Sales** table in the **Data pane.**

1. Notice that a **calculator icon** (🧮) appears next to your new measure — this indicates it’s a DAX measure rather than a column.
2. The name “Count of Sales” is just the label for your measure — you can rename it at any time to make it clearer.

**A screenshot of a computer

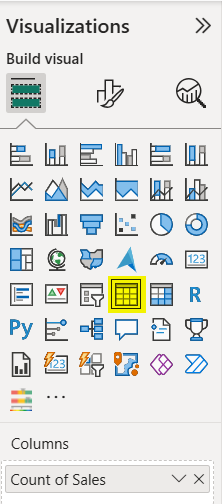
AI-generated content may be incorrect.**

Lesson: Wrangling

**Step 3: Visualizing the DAX Measure in a Table**

1. **Select the Table Visual**

* In the Visualizations pane, click on the Table icon (highlighted in the image).
* This will add a blank table visual to your report canvas.

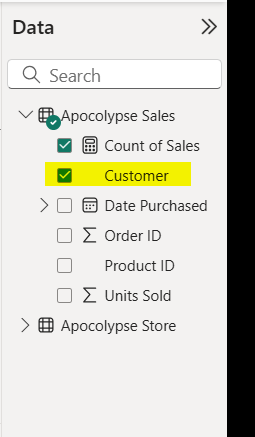
****

1. **Add Your Measure**

* Select the Count of Sales measure from the Apocolypse Sales table into the Values section of the visual.
* You’ll now see the total number of sales (in this case, 74) displayed in the table.

1. **Add a Dimension (Customer)**

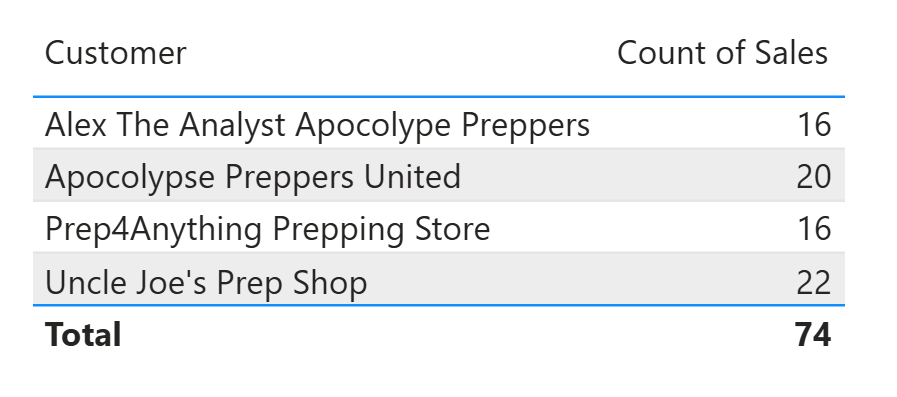
* In the Data pane, check the box next to Customer under Apocolypse Sales.

****

The visual will now display each customer alongside their corresponding count of sales.

1. **Analyze the Results**

* You can now see which customer made the most purchases.
* For example, in this case, Uncle Joe’s Prep Shop had the highest number of sales (22).

****

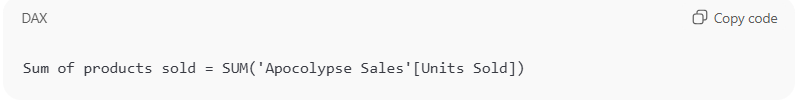
💡 Tip:

Table visuals are great for summarizing and comparing DAX measures across different categories (like customers, products, or regions).

**Step 4: Creating a SUM Measure for Total Products Sold**

1. **Create a New Measure**

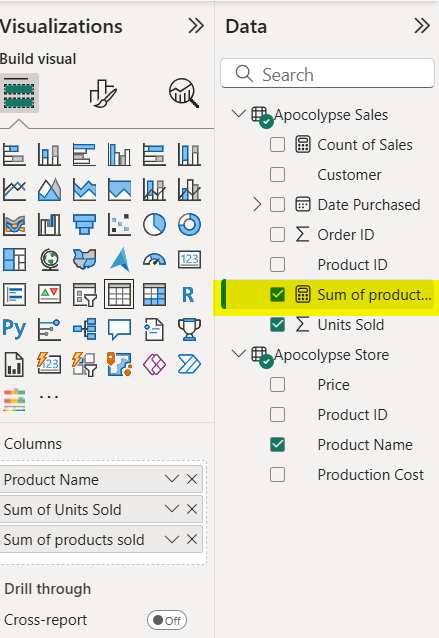
* Go to the Apocolypse Sales table again.
* Right-click and select New measure.
* In the formula bar, type the following DAX expression:



* Press **Enter** to create the measure.

1. **Add a Matrix Visual**

* In the Visualizations pane, select the Matrix icon.
* Select Product Name from the Apocolypse Store table into the Rows field.
* Then Select both Sum of Units Sold and Sum of products sold into the Values field.

****

1. **Analyze Your Matrix**

* The matrix now displays each product alongside its Sum of Units Sold and Sum of products sold.
* You can see the total at the bottom (in this example, 613 units in total).

**A screenshot of a graph

AI-generated content may be incorrect.**

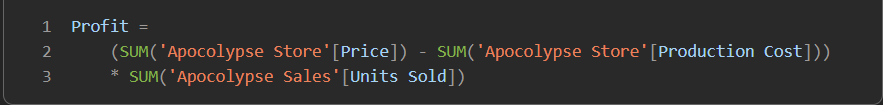
💡 Tip: Even though both columns show the same total here, this exercise demonstrates how to create and compare multiple measures — a core skill in DAX reporting.

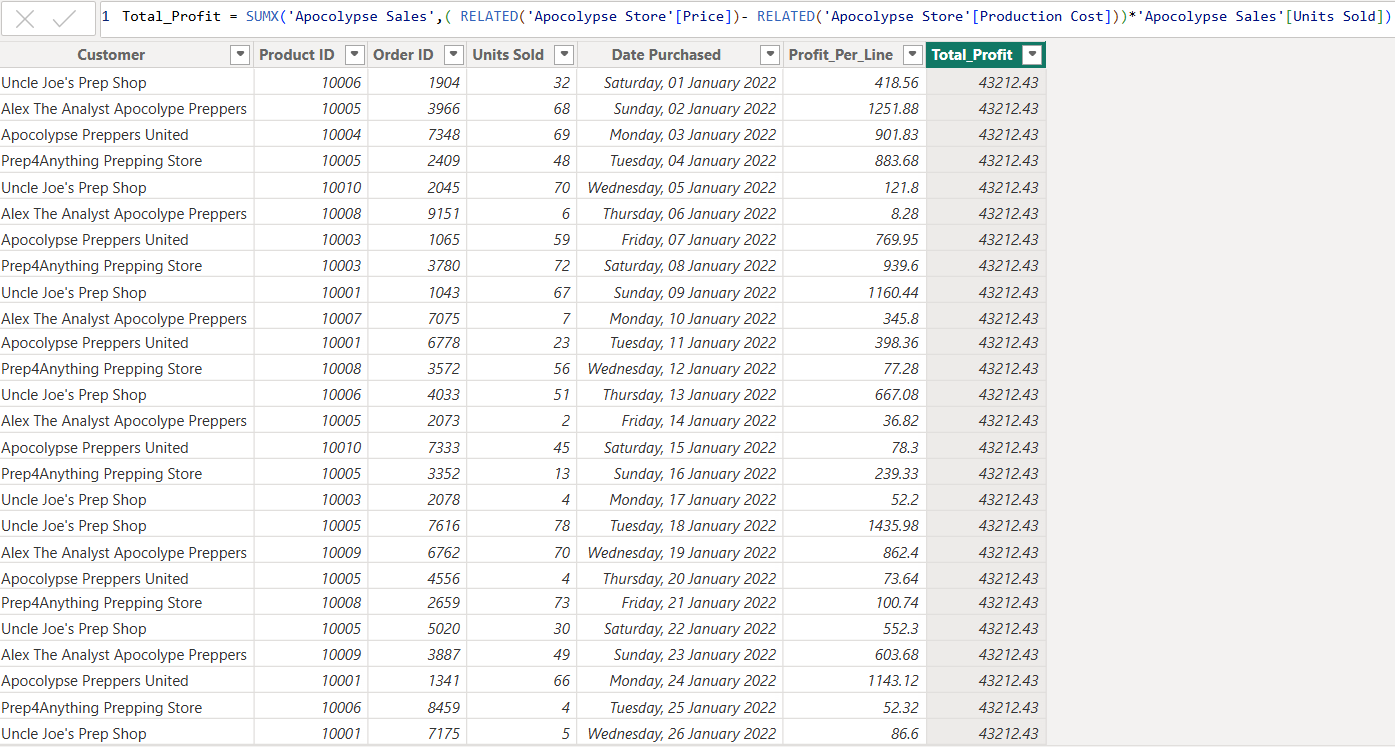
**We are now looking at the difference between Sum and Sumx**

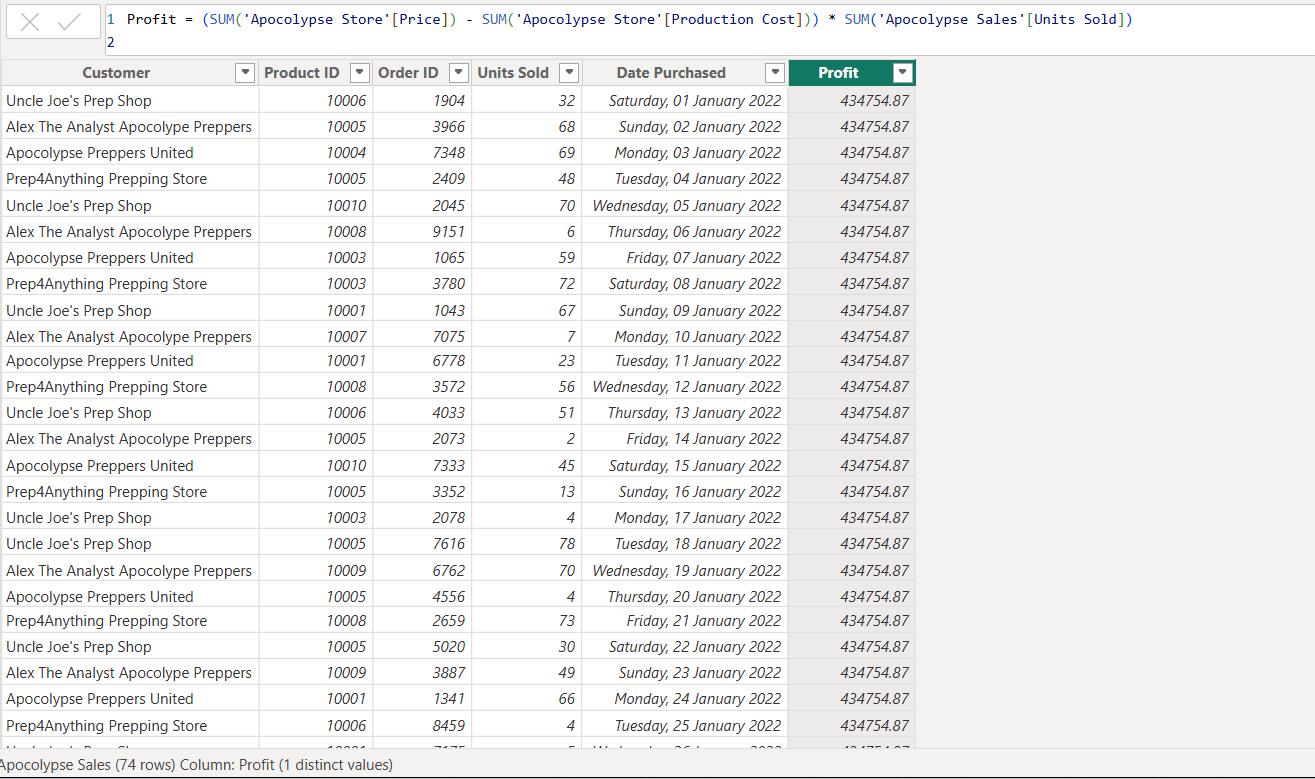
**Step 5: Profit per line (column) + Total Profit (SUMX measure)**

**A) Create a row-level profit column (in Apocolypse Sales)**

1. Go to **Data view** → select **Apocolypse Sales** → **New column**.
2. Name it **Profit** and use:





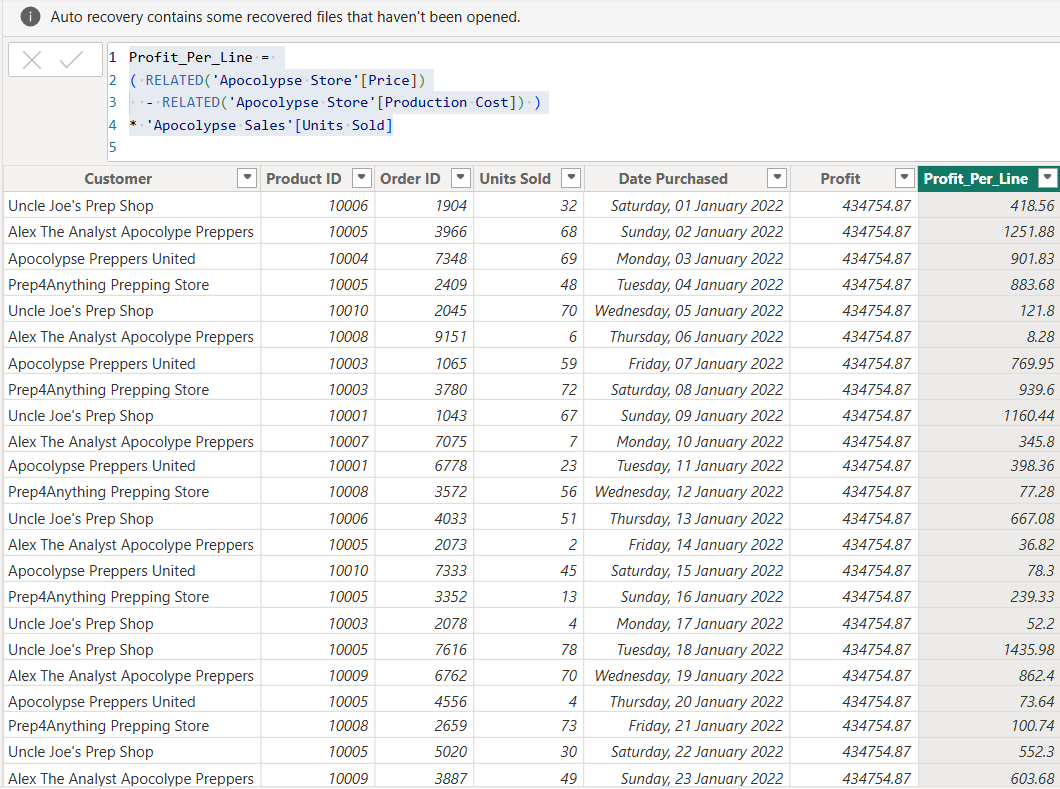


**B) Create a SUMX measure for total profit (filters aware)**

1. A screen shot of a computer

   AI-generated content may be incorrect.In **Apocolypse Sales**, **New measure** → name **Total Profit**:

* SUMX iterates the Sales table, calculating row profit in the current filter context and summing it.



🧮 **Difference Between SUM and SUMX in Power BI**

| **Function** | **What It Does** | **When to Use** | **Example** |
| --- | --- | --- | --- |
| **SUM** | Adds up **a single column** of numbers. It doesn’t evaluate any expressions per row — it just totals one field. | When you need a simple column total. | SUM('Apocolypse Sales'[Units Sold]) — totals all units sold. |
| **SUMX** | Works **row by row** across a table (like Excel’s =SUMPRODUCT) — it **evaluates an expression for each row**, then sums the results. | When you need to multiply or calculate something *before* summing (e.g., Price × Quantity, Profit per product). | SUMX('Apocolypse Sales', 'Apocolypse Sales'[Price] \* 'Apocolypse Sales'[Units Sold]) — calculates total sales amount. |

A yellow square with black corners

AI-generated content may be incorrect.LESSON SUMMARY

You should now be able to:

* SUM is **fast and simple**, but it only works on a **single column** — it doesn’t perform row-by-row calculations.
* SUMX is **more flexible**, allowing you to **evaluate an expression for each row** before summing

Unit 6

Lesson 1

## Creating Visualizations in Power BI

LESSON OBJECTIVES

After completing this lesson, you will be able to:

* Understand what data visualizations are and why they’re essential in Power BI



### Introduction to Visualizations

Visualizations are the heart of Power BI — they turn raw data into meaningful insights that can be easily understood. Instead of scanning through rows of numbers, visuals help you see patterns, trends, and relationships at a glance.

With Power BI, you can create a wide variety of visuals such as bar charts, pie charts, tables, maps, and KPIs to tell your data story. Each visualization helps answer a different type of question — for example:

* Which products are performing best?
* How are sales trending over time?
* Which customers contribute the most revenue?

Creating effective visuals is not just about displaying data — it’s about communicating insights clearly and interactively. Power BI allows you to customize visuals, apply filters, and connect multiple visuals together to create a dynamic, story-driven dashboard.

**Creating a Stacked Bar Chart in Power BI**

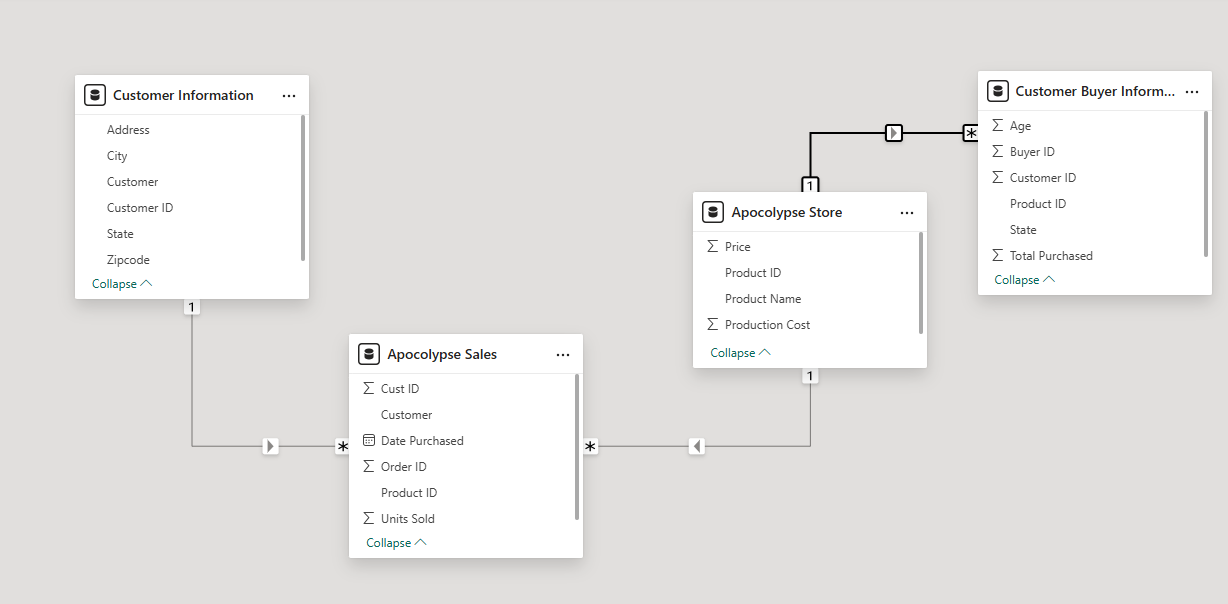
A stacked bar chart is one of the most popular visuals in Power BI. It allows you to compare values across categories and see how each part contributes to the total — all in a single view.

In this example, we’ll visualize the **total Units Sold** for each **Product Name** using a stacked bar chart.

🧭 **Step-by-Step Guide**

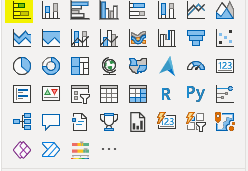
1. Open the Dataset
2. Open your Power BI file titled “**Apocalypse Food Prep – Visualizations Tutorial**”.
3. Tick All Columns
4. Make sure all columns from the dataset are selected so that Power BI recognizes all available fields.
5. Go to Model View
6. Select the Model View (relationship icon on the left sidebar).
7. Verify that all your tables are connected

You should see a relationship between Product ID in Apocalypse Store and Product ID Purchased in Customer Buyer Information.

****

**Building the Visualization**

* + 1. **Select a Stacked Bar Chart**
    2. In the **Visualizations pane**, click on **the Stacked Bar Chart icon.**

****

* + 1. Select Product Name (from Apocalypse Store) to the Y-axis.
    2. Select Units Sold (from Apocalypse Sales) to the X-axis.
    3. You now have a simple bar chart showing the Sum of Units Sold by Product Name.

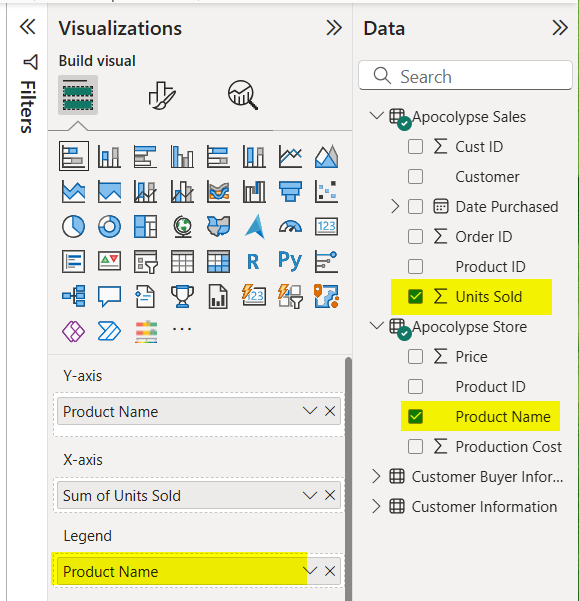
A graph with blue lines

AI-generated content may be incorrect.

**Add Color and Detail**

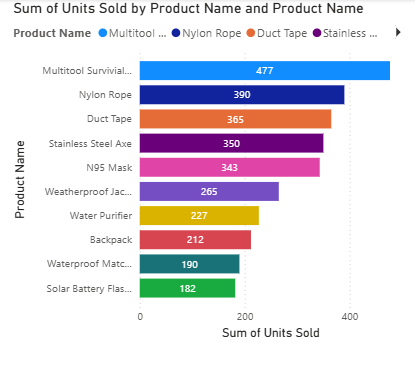
**Add a Legend**

1. To make the chart multi-colored, drag Product Name again to the **Legend section**.



1. Each bar now represents a product with its own color.
2. **Format the Chart**
3. Click the paint roller icon in the Visualizations pane to open the Format Your Visual options.
   * + 1. A yellow rectangle with a brush and a black text

          AI-generated content may be incorrect.
4. Scroll down and turn Data Labels → On.
5. Adjust label position, color, and font size for better readability.



**Section 2: Creating a 100% Stacked Column Chart**

A 100% stacked column chart allows you to compare the relative percentage contribution of each category to the total instead of showing raw numbers. It’s a great way to understand proportions and distribution within your dataset.

In this example, we’ll visualize the percentage of total units sold by each customer.

1. **Select the Chart Type**

From the Visualizations pane, choose the 100% Stacked Column Chart icon.

1. **Add Data Fields**
   * Under Customer Information, select Customer.
   * Under Apocalypse Sales, select Units Sold.

Power BI will automatically plot the Sum of Units Sold for each customer as a percentage of the total.

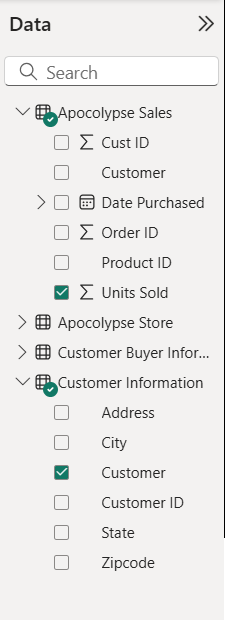
1. **Customize the Chart**

* Add a clear title, e.g., “Sum of Units Sold by Customer”.
* Adjust the colors and labels by selecting the paint roller icon to open the Format Visual pane.
* Turn Data Labels → On to display the percentage values directly on the bars.

**Result**

You’ll now see a visual that shows how much each customer contributes to total sales, expressed as a percentage.

This type of chart helps you quickly identify which customers make up the largest share of your total sales — providing insights into customer performance and contribution.

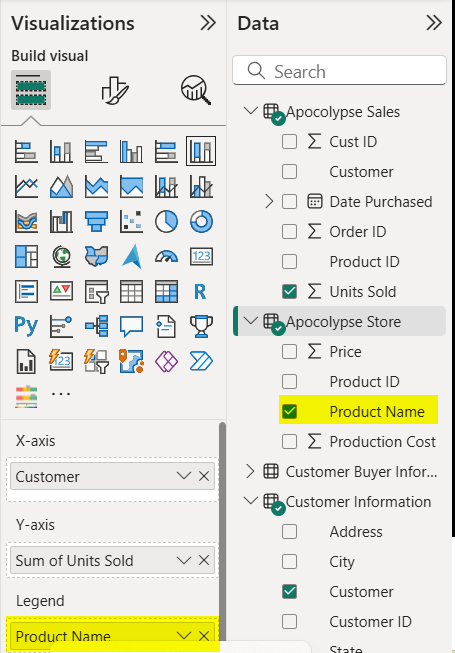
 A screenshot of a graph

AI-generated content may be incorrect.

1. **Add the Product Breakdown**

To see which products make up each customer’s total, drag **Product Name (from Apocalypse Store**) into the Legend area.

Each color in the bar now represents a different product — showing how much of each customer’s purchases come from that specific product.

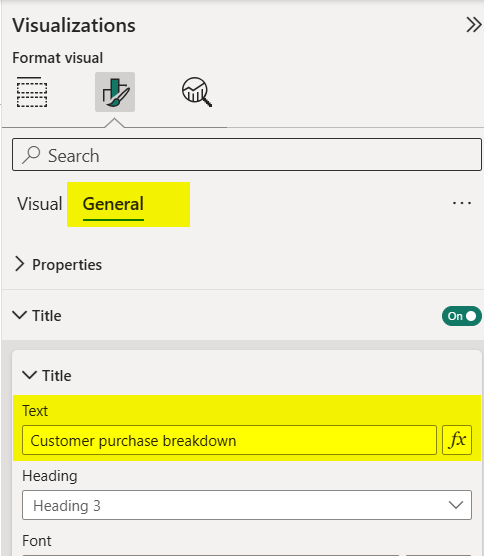


1. **Format Your Chart**

* Add Data Labels
* Click the chart.
* Go to the Format Visual tab (paint roller icon).
* Turn Data Labels → On.
* This displays percentage values for each product segment within the total.

1. **Edit the Chart Title**

* Under Visual → General → Title, type:
* Customer Purchase Breakdown
* Adjust Axis Titles (Optional)
* Under X-axis, rename the title to: Customer

A screenshot of a graph

AI-generated content may be incorrect.

📈 **Creating a Line Chart**

A Line Chart is perfect for showing trends over time — such as sales, growth, or performance across days, months, or years. It helps you identify patterns, seasonality, and changes in data behavior over a timeline.

In this example, we’ll visualize how product sales fluctuate over time.

**🧭 Step-by-Step Guide**

1. **Select the Line Chart Icon**

From the Visualizations pane, choose the Line Chart visual.

1. **Add Data Fields**

From Apocalypse Sales, select:

* Date Purchased → drag to the X-axis
* Units Sold → drag to the Y-axis
* Adjust the X-Axis Hierarchy: Remove the Date Purchased and Year.

A screenshot of a computer

AI-generated content may be incorrect.A screenshot of a computer

AI-generated content may be incorrect.

1. **Add Product Names to the Legend**

From Apocalypse Store, drag Product Name to the Legend field.

Each product will now have its own color line, making it easier to track trends per product.

A screenshot of a computer

AI-generated content may be incorrect.

A graph of colored lines

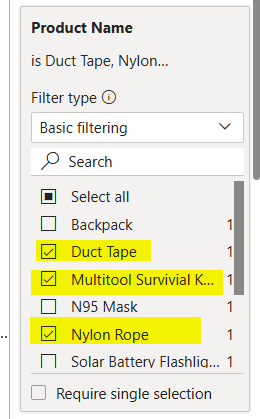
AI-generated content may be incorrect.

1. **Filter Specific Products**

Filter to Focus on Key Products

In the Filters Pane, under Product Name, select only:

* Nylon Rope
* Duct Tape
* Multitool Survival Knife

This helps you focus on the top-selling or most interesting products.

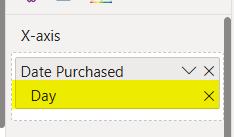
1. **Zoom in on More Detail**

Switch to Daily View (Optional)

Go back to the X-axis and expand Date Purchased.

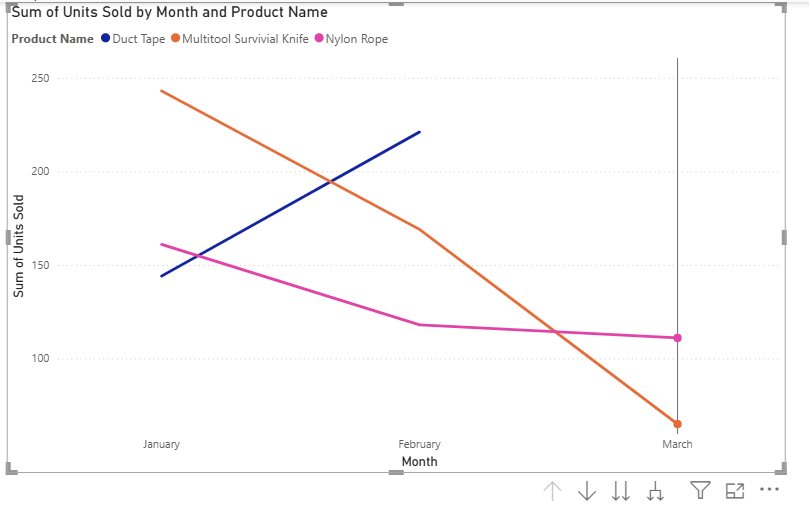
Choose Day instead of Month.

This gives a more granular view — letting you analyze sales spikes or dips on specific days.

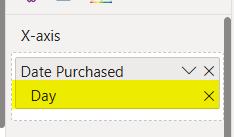


A graph with lines and numbers

AI-generated content may be incorrect.



We can even filter the day to get even more information.



A graph with lines and numbers

AI-generated content may be incorrect.

📊 **Creating a Line and Clustered Column Chart**

The Line and Clustered Column Chart is perfect when you want to compare two different measures that share a common category — for example, product price versus production cost.

It helps you analyze profit margins or see how much it costs to produce each product compared to its selling price.

🧭 **Step-by-Step Guide**

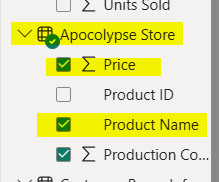
1. **Select the Chart Type**

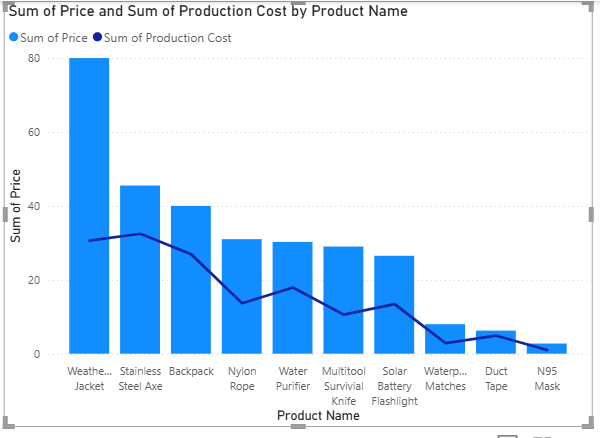
From the Visualizations pane, choose the Line and Clustered Column Chart icon.

1. **Add Data Fields**

* X-axis: Select Product Name (from Apocalypse Store)
* Column Y-axis: Sum of Price
* Line Y-axis: Sum of Production Cost

A screenshot of a computer

AI-generated content may be incorrect.



**Interpret the Chart**

* The columns (blue bars) show the Sum of Price for each product.
* The line (dark blue) represents the Sum of Production Cost.
* This allows you to easily see how much it costs to make each item versus how much it sells for.

**Creating a Scatter Plot**

A Scatter Plot helps you visualize the relationship between two numerical values.

It’s particularly useful for identifying correlations, clusters, and outliers — points that stand out from the rest of your data.

In this example, we’ll explore how price and production cost relate for different products.

**🧭 Step-by-Step Guide**

1. Select the Chart Type

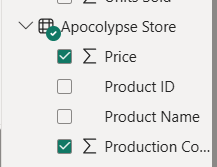
From the Visualizations pane, select the Scatter Chart icon.

1. **Add Data Fields**

From Apocalypse Store:

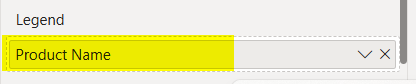
* Select Price → into the Y-axis
* Select Production Cost → into the X-axis
* A screenshot of a list

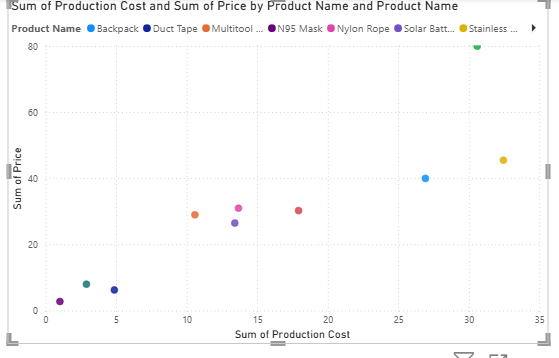
  AI-generated content may be incorrect.Select Product Name → into the Values area



1. **Add a Legend**

Drag Product Name again into the Legend field.



 **Scatter Plot**

**Creating a Donut / Pie Chart**

The Donut Chart (or Pie Chart) is ideal for showing how different categories contribute to a total.

In this case, we’ll analyze total purchases by state to see which regions contribute the most to overall sales.

**Step-by-Step Guide**

1. **Select the Chart Type**

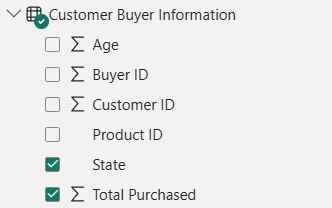
From the Visualizations pane, choose either:

* The Pie Chart icon 🥧, or
* The Donut Chart icon 🍩 (for a more modern and clear design).

1. **Add Data Fields**

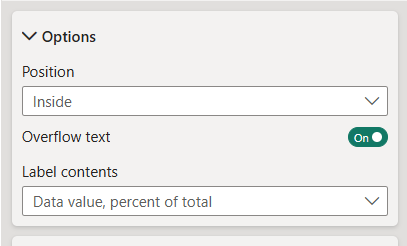
From Customer Buyer Information:

* Select Total Purchased
* Select State

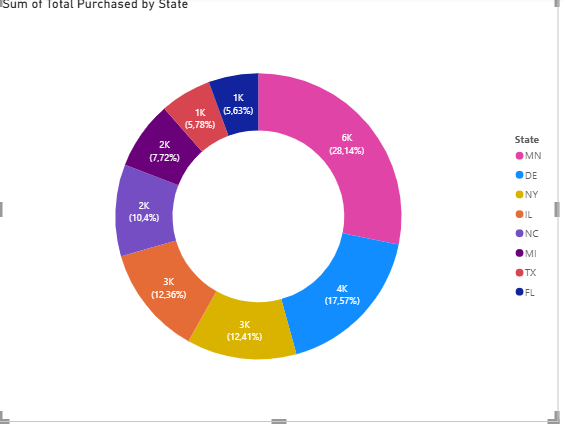


**Customize Your Chart**

* Format the Data Labels
* Go to the Format Visual panel (paint roller icon).
* And change position to inside



.



**Creating a Card Visualization**

A Card visual is used to display single value summaries such as totals, averages, minimums, or maximums.

It’s perfect for highlighting KPIs like total revenue, number of customers, or minimum purchase value — giving quick, clear insights.

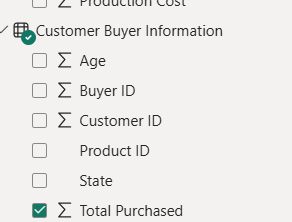
**Step-by-Step Guide**

1. Select the Chart Type

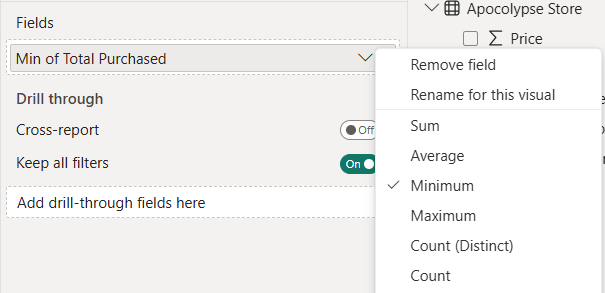
From the Visualizations pane, select the Card icon 🧮.

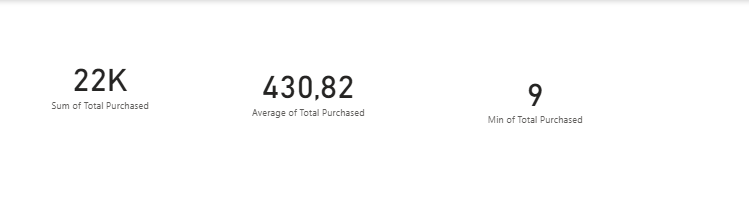
1. **Add Your Data Field**

From Customer Buyer Information, select Total Purchased into the Fields area.

You’ll immediately see the Sum of Total Purchased displayed as a large number.

Then you can drill down





## Learning Assessment

1. PowerBi Models always contains data

*Determine whether this statement is true or false.*

True

 False

1. What are types of models used in PowerBI

*Choose the correct answers.*

* 1. Import Model
  2. SQL Server
  3. Microsoft Fabric

 D Local Model

1. Which type of visualization would you use to show trends over time?

*Determine whether this statement is true or false.*

True

 False

1. Which type of visualization would you use to show trends over time?

*Choose the correct answers.*

* 1. Pie Chart
  2. Line Chart
  3. Bar Chart
  4. Table