

Power Query Basics



power query in excel



Power Query in Excel is a data connection technology that enables you to discover, connect, combine, and refine data across a wide variety of sources. It is a powerful tool for data transformation and preparation. Here's a brief overview of how to use Power Query:

Accessing Power Query

1. **Excel 2016 and later:** Go to the `Data` tab and find the `Get & Transform Data` group.
2. **Excel 2010 and 2013:** You might need to install the Power Query add-in. Once installed, you'll find it under the `Power Query` tab.

Basic Steps in Power Query

1. **Load Data:** Click on `Get Data` to import data from various sources (Excel files, databases, web pages, etc.).
2. **Transform Data:** Use the Power Query Editor to clean and transform your data. This includes:
 - **Filtering Rows:** Remove unnecessary rows.
 - **Removing Columns:** Remove columns that are not needed.
 - **Changing Data Types:** Ensure data types are correct.
 - **Splitting Columns:** Split columns by delimiter or fixed width.
 - **Merging Queries:** Combine multiple queries (like SQL joins).
 - **Pivot/Unpivot Columns:** Transform your data structure.
3. **Load Data to Excel:** Once your data is prepared, you can load it back into Excel as a table, PivotTable, or chart.

Example: Importing and Transforming Data

1. **Import Data:** Click on `Data` > `Get Data` > `From File` > `From Workbook`, and select your Excel file.
2. **Open Power Query Editor:** The data will load, and the Power Query Editor will open.
3. **Transformations:**
 - **Remove Unnecessary Columns:** Right-click on the column headers and select `Remove`.
 - **Filter Rows:** Click the drop-down arrow in the column header and apply your filter.
 - **Change Data Types:** Click the icon in the column header and select the correct data type.
 - **Split Columns:** Right-click on the column, select `Split Column`, and choose your splitting option.
4. **Close & Load:** Click `Close & Load` to bring the data back into Excel.

Advanced Features

- **Custom Columns:** Create new columns using custom formulas.
- **Conditional Columns:** Create columns based on conditions.

- **Group By:** Aggregate data by grouping it.
- **Merge Queries:** Combine data from different queries.

Power Query is especially useful for handling large datasets and repetitive tasks, making data preparation more efficient and less error-prone.



cleaning and transformation



Cleaning and transforming data in Power Query involves various steps to ensure the data is accurate, consistent, and ready for analysis. Here's a detailed guide on how to clean and transform data using Power Query in Excel:

Step-by-Step Guide to Data Cleaning and Transformation

1. Load Data into Power Query Editor:

- Go to the **Data** tab in Excel.
- Click on **Get Data** and select your data source (e.g., Excel file, CSV, database).
- Load the data into the Power Query Editor.

2. Remove Unnecessary Columns:

- Identify columns that are not needed for analysis.
- Right-click on the column header and select **Remove** or use the **Choose Columns** option in the **Home** tab.

3. Filter Rows:

- Apply filters to remove unwanted rows.
- Click the drop-down arrow in the column header and select the filter criteria.

4. Change Data Types:

- Ensure each column has the correct data type (e.g., text, number, date).
- Click on the data type icon in the column header and select the appropriate data type.

5. Remove Duplicates:

- Eliminate duplicate rows to ensure data uniqueness.
- Go to the **Home** tab and click on **Remove Rows** > **Remove Duplicates**.

6. Handle Missing Data:

- Replace missing values or remove rows with missing data.
- Use the **Transform** tab and select **Replace Values** or **Remove Rows** > **Remove Blank Rows**.

7. Trim and Clean Text:

- Remove extra spaces and clean text data.
- Select the column, go to the **Transform** tab, and choose **Format** > **Trim** and **Clean**.

8. Split Columns:

- Split columns based on delimiters or fixed width.
- Right-click on the column, select **Split Column**, and choose the splitting option.

9. Merge Columns:

- Combine multiple columns into one.
- Select the columns to merge, right-click, and choose **Merge Columns**.

10. Rename Columns:

- Ensure column names are meaningful and consistent.
- Double-click the column header and type the new name.

11. Apply Conditional Columns:

- Create new columns based on conditions.
- Go to the `Add Column` tab and select `Conditional Column`.

12. Group By:

- Aggregate data by grouping it based on one or more columns.
- Go to the `Transform` tab and click on `Group By`.

13. Pivot and Unpivot Columns:

- Transform the structure of your data.
- Use the `Transform` tab to select `Pivot Column` or `Unpivot Columns`.

14. Custom Columns:

- Create new columns using custom formulas.
- Go to the `Add Column` tab and select `Custom Column`.

15. Sort Data:

- Sort data in ascending or descending order.
- Click the drop-down arrow in the column header and select the sort option.

16. Combine Queries:

- Merge or append queries to combine data from different sources.
- Go to the `Home` tab and select `Merge Queries` or `Append Queries`.

Example Workflow

Let's consider an example of cleaning and transforming a dataset:

1. **Load Data:** Import a CSV file with sales data.
2. **Remove Columns:** Remove columns like "Address" and "Phone Number" that are not needed.
3. **Filter Rows:** Filter out rows where the "Sales Amount" is zero.
4. **Change Data Types:** Ensure the "Date" column is of type Date and the "Sales Amount" column is of type Decimal Number.
5. **Remove Duplicates:** Remove duplicate rows based on the "Order ID".
6. **Handle Missing Data:** Replace missing values in the "Customer Name" column with "Unknown".
7. **Trim and Clean:** Trim and clean the "Customer Name" column.
8. **Split Column:** Split the "Product Category" column which contains values separated by commas.
9. **Merge Columns:** Merge the "First Name" and "Last Name" columns into a single "Full Name" column.
10. **Rename Columns:** Rename "Order ID" to "Order Number" and "Sales Amount" to "Total Sales".
11. **Group By:** Group the data by "Product Category" and aggregate the "Total Sales".
12. **Sort Data:** Sort the grouped data by "Total Sales" in descending order.

Closing and Loading the Data

After performing all the necessary transformations:

1. Click `Close & Load` in the `Home` tab.
2. Choose to load the data into a new worksheet, existing worksheet, or as a connection.

This will bring your cleaned and transformed data back into Excel, ready for analysis.



dealing with text tools in power query



Power Query offers several tools to handle and transform text data. These tools can be found in the `Transform` and `Add Column` tabs in the Power Query Editor. Here's a guide on how to use these text tools effectively:

Common Text Transformation Tools

1. Trim

- Removes leading and trailing spaces from text.
- Go to the `Transform` tab, select the text column, and click on `Format` > `Trim`.

2. Clean

- Removes non-printable characters from text.
- Go to the `Transform` tab, select the text column, and click on `Format` > `Clean`.

3. Uppercase, Lowercase, and Capitalize Each Word

- Change the text to uppercase, lowercase, or capitalize each word.
- Go to the `Transform` tab, select the text column, and click on `Format`, then choose the appropriate option (Uppercase, Lowercase, or Capitalize Each Word).

4. Replace Values

- Replace specific text values with another text.
- Right-click the text column, select `Replace Values`, enter the value to replace and the new value, and click `OK`.

5. Split Column

- Split text into multiple columns based on a delimiter or a fixed width.
- Right-click the text column, select `Split Column`, and choose either `By Delimiter` or `By Number of Characters`.
- For `By Delimiter`, specify the delimiter (e.g., comma, space) and choose whether to split at each occurrence or at a specific number of times.
- For `By Number of Characters`, specify the number of characters and whether to split once or repeatedly.

6. Merge Columns

- Combine multiple columns into a single column.
- Select the columns to merge, go to the `Transform` tab, and click on `Merge Columns`.
- Specify the separator to use (e.g., space, comma) and provide a name for the new column.

7. Extract

- Extract specific parts of the text, such as the first few characters, last few characters, or text between delimiters.
- Go to the `Add Column` tab, select the text column, and click on `Extract`, then choose the appropriate option (`First Characters`, `Last Characters`, `Range of Characters`).

8. Text Length

- Add a column that shows the length of the text in each row.
- Go to the `Add Column` tab, select the text column, and click on `Statistics` > `Text Length`.

9. Substring

- Create a new column by extracting a substring from the text.
- Go to the `Add Column` tab, select the text column, and click on `Custom Column`.
- Use the `Text.Middle` function to specify the starting position and the length of the substring. For example: `= Text.Middle([ColumnName], startPosition, length)`.

Advanced Text Functions

For more advanced transformations, you can use custom columns and M code. Here are some examples of M functions for text manipulation:

1. Text.Contains

- Checks if a text contains a specific substring.
- Example: `= Text.Contains([ColumnName], "substring")`

2. Text.StartsWith and Text.EndsWith

- Checks if a text starts or ends with a specific substring.
- Example: `= Text.StartsWith([ColumnName], "prefix")`
- Example: `= Text.EndsWith([ColumnName], "suffix")`

3. Text.Replace

- Replaces all occurrences of a substring with another substring.
- Example: `= Text.Replace([ColumnName], "oldValue", "newValue")`

4. Text.Upper, Text.Lower, and Text.Proper

- Converts text to uppercase, lowercase, or proper case.
- Example: `= Text.Upper([ColumnName])`
- Example: `= Text.Lower([ColumnName])`
- Example: `= Text.Proper([ColumnName])`

Example Workflow

Let's consider a scenario where you need to clean and transform a column with full names (e.g., "John Doe"):

1. **Load Data:** Import the data into Power Query.
2. **Trim:** Remove leading and trailing spaces.
 - Select the `Full Name` column, go to `Transform` > `Format` > `Trim``.
3. **Clean:** Remove non-printable characters.
 - Select the `Full Name` column, go to `Transform` > `Format` > `Clean``.
4. **Capitalize Each Word:** Ensure the names are in proper case.
 - Select the `Full Name` column, go to `Transform` > `Format` > `Capitalize Each Word``.
5. **Split Column:** Split the `Full Name` column into `First Name` and `Last Name`.
 - Select the `Full Name` column, right-click, choose `Split Column` > `By Delimiter``.
 - Choose space as the delimiter and split at each occurrence.

By using these text tools, you can efficiently clean and transform your text data in Power Query.



number tools



Power Query in Excel offers a variety of tools to manipulate and transform numeric data. These tools can be found in the `Transform`` and `Add Column`` tabs within the Power Query Editor. Here's a comprehensive guide on how to use these number tools:

Common Number Transformation Tools

1. Change Data Type

- Ensure that the column is recognized as a number.

- Select the column, go to the `Transform` tab, click on `Data Type`, and select the appropriate number type (e.g., Decimal Number, Whole Number, Currency).

2. Replace Values

- Replace specific numeric values with another value.
- Right-click the column, select `Replace Values`, enter the value to replace and the new value, and click `OK`.

3. Round Numbers

- Round numbers to a specified number of decimal places.
- Select the column, go to the `Transform` tab, click on `Round`, and choose the rounding option (Round, Round Up, Round Down).
- Specify the number of decimal places.

4. Statistics

- Perform basic statistical operations like Sum, Average, Min, Max, Count, Standard Deviation, etc.
- Select the column, go to the `Transform` tab, and click on `Statistics`. Choose the desired operation.

5. Standardize Numbers

- Normalize your data to have a mean of 0 and a standard deviation of 1.
- Select the column, go to the `Transform` tab, click on `Standardize`, and specify the parameters if needed.

6. Add Column from Examples

- Automatically generate new columns based on examples you provide.
- Go to the `Add Column` tab, click on `Column from Examples`, and provide examples in the new column based on your data. Power Query will infer the transformation logic.

7. Add Custom Column

- Create new columns using custom formulas.
- Go to the `Add Column` tab, click on `Custom Column`, and use M language to define your formula (e.g., `= [Column1] + [Column2]`).

8. Mathematical Operations

- Perform basic arithmetic operations.
- Go to the `Add Column` tab, click on `Standard`, and choose from options like Add, Subtract, Multiply, Divide, Integer-Divide, Modulo, etc.

Advanced Number Functions

For more advanced numeric transformations, you can use custom columns and M code. Here are some examples of M functions for number manipulation:

1. Number.Abs

- Returns the absolute value of a number.
- Example: `= Number.Abs([ColumnName])`

2. Number.Power

- Returns a number raised to the specified power.
- Example: `= Number.Power([ColumnName], 2)` (for squaring the number)

3. Number.Sqrt

- Returns the square root of a number.
- Example: `= Number.Sqrt([ColumnName])`

4. Number.Round, Number.RoundUp, Number.RoundDown

- Rounds a number to the nearest integer or specified number of decimal places.
- Example: `= Number.Round([ColumnName], 2)`

- Example: `= Number.RoundUp([ColumnName], 2)`
- Example: `= Number.RoundDown([ColumnName], 2)`

5. Number.Mod

- Returns the remainder of a number divided by another number.
- Example: `= Number.Mod([ColumnName], divisor)`

6. Number.Sign

- Returns the sign of a number (-1, 0, 1).
- Example: `= Number.Sign([ColumnName])`

Example Workflow

Let's consider a scenario where you need to clean and transform a column with sales data:

1. **Load Data:** Import the data into Power Query.
2. **Change Data Type:** Ensure the `Sales Amount` column is of type Decimal Number.
 - Select the `Sales Amount` column, go to `Transform` > `Data Type` > `Decimal Number`.
3. **Replace Values:** Replace negative sales amounts with 0.
 - Select the `Sales Amount` column, right-click, and choose `Replace Values`.
 - Enter `-` in the `Value to Find` field and `0` in the `Replace With` field, then click `OK`.
4. **Round Numbers:** Round the sales amounts to 2 decimal places.
 - Select the `Sales Amount` column, go to `Transform` > `Round` > `Round`.
 - Enter `2` for the number of decimal places and click `OK`.
5. **Add Custom Column:** Calculate 10% tax on the sales amounts and create a new column for it.
 - Go to `Add Column` > `Custom Column`.
 - Name the new column `Tax Amount` and use the formula `= [Sales Amount] * 0.1`.
6. **Add Another Custom Column:** Add the tax amount to the sales amount to get the total amount.
 - Go to `Add Column` > `Custom Column`.
 - Name the new column `Total Amount` and use the formula `= [Sales Amount] + [Tax Amount]`.

Closing and Loading the Data

After performing all the necessary transformations:

1. Click `Close & Load` in the `Home` tab.
2. Choose to load the data into a new worksheet, existing worksheet, or as a connection.

This will bring your cleaned and transformed numeric data back into Excel, ready for analysis.



date and time tools



Power Query provides robust tools for handling date and time data, allowing you to perform a wide range of transformations. Here's a detailed guide on using date and time tools in Power Query:

Common Date and Time Transformation Tools

1. Change Data Type

- Ensure that the column is recognized as a Date, Time, or DateTime.
- Select the column, go to the `Transform` tab, click on `Data Type`, and choose `Date`, `Time`, or `DateTime`.

2. Extract Date Components

- Extract parts of the date like year, month, day, etc.
- Select the date column, go to the `Add Column` tab, and click on `Date`. You can then select options like `Year`, `Month`, `Day`, `Start of Year`, `End of Year`, etc.

3. Extract Time Components

- Extract parts of the time like hour, minute, second, etc.
- Select the time column, go to the `Add Column` tab, and click on `Time`. You can then select options like `Hour`, `Minute`, `Second`.

4. Calculate Age

- Calculate the age based on a date column.
- Select the date column, go to the `Add Column` tab, and click on `Date` > `Age`.

5. Date Arithmetic

- Perform arithmetic operations like adding or subtracting days, months, or years.
- Use custom columns and M code to add/subtract date values.
- Example: `= Date.AddDays([DateColumn], 10)` adds 10 days to each date in the column.

6. Difference Between Dates

- Calculate the difference between two date columns.
- Go to the `Add Column` tab, click on `Custom Column`, and use the formula `= Duration.Days([EndDate] - [StartDate])`.

Advanced Date and Time Functions

For more advanced transformations, you can use custom columns and M code. Here are some examples of M functions for date and time manipulation:

1. Date.From

- Converts a value to a date.
- Example: `= Date.From([ColumnName])`

2. Date.Year, Date.Month, Date.Day

- Extract the year, month, or day from a date.
- Example: `= Date.Year([DateColumn])`
- Example: `= Date.Month([DateColumn])`
- Example: `= Date.Day([DateColumn])`

3. Date.AddDays, Date.AddMonths, Date.AddYears

- Adds a specified number of days, months, or years to a date.
- Example: `= Date.AddDays([DateColumn], 10)`
- Example: `= Date.AddMonths([DateColumn], 1)`
- Example: `= Date.AddYears([DateColumn], 5)`

4. Date.StartOfDay, Date.EndOfDay

- Returns the start or end of the day for a date.
- Example: `= Date.StartOfDay([DateColumn])`
- Example: `= Date.EndOfDay([DateColumn])`

5. Date.DayOfWeek

- Returns the day of the week as a number (0 = Sunday, 1 = Monday, etc.).
- Example: `= Date.DayOfWeek([DateColumn])`

6. DateTime.FixedLocalNow

- Returns the current date and time in the local time zone.

- Example: `= DateTime.FixedLocalNow()`

7. Time.From

- Converts a value to a time.
- Example: `= Time.From([ColumnName])`

8. Time.Hour, Time.Minute, Time.Second

- Extract the hour, minute, or second from a time.
- Example: `= Time.Hour([TimeColumn])`
- Example: `= Time.Minute([TimeColumn])`
- Example: `= Time.Second([TimeColumn])`

Example Workflow

Let's consider a scenario where you need to transform a dataset with a `DateTime` column:

1. **Load Data:** Import the data into Power Query.
2. **Change Data Type:** Ensure the `DateTime` column is recognized as `DateTime`.
 - Select the `DateTime` column, go to `Transform` > `Data Type` > `DateTime`.
3. **Extract Date Components:**
 - Extract the year, month, and day.
 - Select the `DateTime` column, go to `Add Column` > `Date` > `Year` to add a new column with the year.
 - Repeat for `Month` and `Day`.
4. **Extract Time Components:**
 - Extract the hour, minute, and second.
 - Select the `DateTime` column, go to `Add Column` > `Time` > `Hour` to add a new column with the hour.
 - Repeat for `Minute` and `Second`.
5. **Calculate Age:**
 - Calculate the age from the `DateTime` column.
 - Select the `DateTime` column, go to `Add Column` > `Date` > `Age`.
6. **Date Arithmetic:**
 - Add 7 days to the `DateTime` column.
 - Go to `Add Column` > `Custom Column`.
 - Name the new column `DatePlus7` and use the formula `= Date.AddDays([DateTime], 7)`.
7. **Difference Between Dates:**
 - Calculate the difference in days between `EndDate` and `StartDate` columns.
 - Go to `Add Column` > `Custom Column`.
 - Name the new column `DaysDifference` and use the formula `= Duration.Days([EndDate] - [StartDate])`.

Closing and Loading the Data

After performing all the necessary transformations:

1. Click `Close & Load` in the `Home` tab.
2. Choose to load the data into a new worksheet, existing worksheet, or as a connection.

This will bring your cleaned and transformed date and time data back into Excel, ready for analysis.



conditional columns in power query



Creating conditional columns in Power Query allows you to add a new column based on values in existing columns, similar to using IF statements in Excel. Here's how to create conditional columns in Power Query:

Step-by-Step Guide to Creating Conditional Columns

1. Load Data into Power Query Editor:

- Open your Excel workbook.
- Go to the `Data` tab, click on `Get Data`, select your data source, and load the data into the Power Query Editor.

2. Open Add Conditional Column Dialog:

- In the Power Query Editor, go to the `Add Column` tab.
- Click on `Conditional Column`.

3. Define the Conditional Column:

- In the `Add Conditional Column` dialog box, specify the name for your new column in the `New column name` field.
- Use the dropdown menus to set your conditions:
 - **Column Name:** Select the column to base your condition on.
 - **Operator:** Choose the comparison operator (e.g., equals, does not equal, is greater than, is less than, etc.).
 - **Value:** Enter the value to compare against.
 - **Output:** Specify the value to display in the new column if the condition is met.

4. Add More Conditions (Optional):

- Click on `Add Clause` to add more conditions.
- Repeat the steps to set the column, operator, value, and output for each additional condition.

5. Specify Default Output:

- If none of the conditions are met, you can specify a default output value.
- Enter the default value in the `Else` field.

6. Click OK:

- After defining all your conditions and default output, click `OK` to create the conditional column.

Example Workflow

Suppose you have a dataset with student scores and you want to create a new column called "Grade" based on the score:

1. **Load Data:** Import the student scores data into Power Query.
2. **Open Add Conditional Column Dialog:** Go to `Add Column` > `Conditional Column`.
3. **Define Conditions:**
 - Name the new column `Grade`.
 - Set the first condition: If `Score` is greater than or equal to 90, then `A`.
 - Click `Add Clause` to add more conditions:
 - If `Score` is greater than or equal to 80, then `B`.

- If `Score` is greater than or equal to 70, then `C`.
- If `Score` is greater than or equal to 60, then `D`.
- For the default output, specify `F` if none of the conditions are met.

Creating Conditional Columns Using Custom M Code

For more complex conditions, you can use the `Custom Column` feature and write M code:

1. Add Custom Column:

- Go to `Add Column` > `Custom Column`.
- Name the new column and enter a formula using M code. For example:

```
m
= if [Score] >= 90 then "A"
  else if [Score] >= 80 then "B"
  else if [Score] >= 70 then "C"
  else if [Score] >= 60 then "D"
  else "F"
```

Examples of Conditional Columns

1. Categorize Sales Data:

- Load your sales data into Power Query.
- Add a conditional column to categorize sales into `High`, `Medium`, and `Low` based on the `Sales Amount`.

```
m
= if [Sales Amount] > 10000 then "High"
  else if [Sales Amount] > 5000 then "Medium"
  else "Low"
```

2. Flag Late Orders:

- Load your order data with `Order Date` and `Delivery Date`.
- Add a conditional column to flag orders as `Late` if the `Delivery Date` is after the `Due Date`.

```
m
= if [Delivery Date] > [Due Date] then "Late"
  else "On Time"
```

Tips for Creating Conditional Columns

- **Test Your Conditions:** Ensure that your conditions are correctly ordered and none are inadvertently skipped.
- **Use Else Clause:** Always specify an `Else` clause to handle cases where none of the conditions are met.

- **Combine Multiple Conditions:** You can combine multiple conditions using logical operators (AND, OR) in custom M code for more complex scenarios.

Closing and Loading the Data

After creating the conditional column:

1. Click `Close & Load` in the `Home` tab.
2. Choose to load the data into a new worksheet, existing worksheet, or as a connection.

This will bring your data, including the new conditional column, back into Excel, ready for analysis.



combine multiple files



Combining multiple files in Power Query allows you to consolidate data from several sources into a single table. This is particularly useful when dealing with data stored across multiple files or sheets. Here's a detailed guide on how to combine multiple files in Power Query:

Combining Files from a Folder

1. **Place Files in a Folder:**
 - Ensure all the files you want to combine are in the same folder.
 - The files should have the same structure (same column headers and data types).
2. **Load Files into Power Query:**
 - Open Excel and go to the `Data` tab.
 - Click on `Get Data` > `From File` > `From Folder`.
 - Browse to the folder containing your files and click `OK`.
3. **Combine Files:**
 - In the `Folder` dialog box, click `Combine`.
 - Power Query will automatically detect the structure of the files.
 - A preview of one of the files will be displayed. Click `OK`.
4. **Edit Query in Power Query Editor:**
 - The combined data will be loaded into the Power Query Editor.
 - You can now make any necessary transformations to clean and prepare your data.

Example Workflow

Let's walk through an example of combining multiple Excel files stored in a folder.

Step-by-Step Example

1. **Prepare Your Files:**
 - Ensure all Excel files to be combined are placed in the same folder.
 - Ensure each file has the same structure (e.g., `SalesData_Jan.xlsx`, `SalesData_Feb.xlsx`, etc.).
2. **Load Files into Power Query:**
 - Open Excel, go to the `Data` tab, click on `Get Data` > `From File` > `From Folder`.

- Browse to the folder containing your files and click `OK`.

3. Combine Files:

- In the `Folder` dialog box, click `Combine`.
- In the `Combine Files` dialog box, Power Query will show a preview of the data.
- Click `OK`.

4. Edit Query in Power Query Editor:

- The combined data will be loaded into the Power Query Editor.
- Make necessary transformations to clean and prepare your data.
- Click `Close & Load` to load the combined data into Excel.

Combining Files from Different Data Sources

Power Query can combine data from different sources as well. Here's how to do it:

1. Load Data from Multiple Sources:

- Load data from the first source into Power Query.
- Repeat the process to load data from the second source into Power Query.

2. Combine Queries:

- In the Power Query Editor, go to the `Home` tab and click on `Append Queries` to combine the data vertically (similar to stacking data).
- Click `Merge Queries` to combine the data horizontally (similar to joining data based on a key column).

Example of Combining Data from Different Sources

Step-by-Step Example

1. Load Data from the First Source:

- Go to the `Data` tab, click on `Get Data`, and select the first data source (e.g., Excel, CSV, etc.).
- Load the data into Power Query.

2. Load Data from the Second Source:

- Repeat the process to load data from the second source into Power Query.

3. Combine Queries:

- In the Power Query Editor, go to the `Home` tab.
- Click `Append Queries` to combine the data vertically (if the data has the same structure).
- Click `Merge Queries` to combine the data horizontally (if the data shares a common key column).

4. Edit Combined Query:

- Make necessary transformations to clean and prepare your data.
- Click `Close & Load` to load the combined data into Excel.

Tips for Combining Files

- **Ensure Consistency:** Make sure that all files to be combined have a consistent structure (same columns and data types).
- **Preview Data:** Use the preview feature in Power Query to ensure the combined data looks correct before loading it into Excel.
- **Transform Data:** Apply necessary transformations in the Power Query Editor to clean and standardize your data before combining it.

- **Use Parameters:** If combining files from different folders or sources, consider using parameters to make the process more dynamic and manageable.

By following these steps, you can efficiently combine multiple files in Power Query, allowing for more streamlined data analysis and reporting.

ChatGPT can make mistakes. Check important info.