



# **POWER QUERY EDITOR**

---

## ***Complete Transformation Guide***

Prepared by: Navitha

# POWER QUERY EDITOR

## COMPLETE TRANSFORMATION GUIDE

Prepared By: Navitha

---

### What is Power Query Editor?

Power Query Editor is a **data transformation and data preparation tool** used in **Power BI** and **Excel**.

It allows us to do:

- Connect to multiple data sources.
- Clean and shape data.
- Transform data without writing SQL.
- Create repeatable ETL steps.

All actions are recorded as Applied Steps using M language.'

### Power Query Editor-Interface Overview

The interface is divided into five main sections:

1. Ribbon Tab (Top Section)
  - The Ribbon contains all the commands used to transform and manage data.
  - It works similarly to the Power BI main ribbon but is focused only on data preparation.

#### Common Ribbon Tabs and What They Do:

- ◆ **Home Tab** (This is the most used tab in Power Query)
- ◆ **Transform Tab** (This tab focuses on modifying data inside columns)
- ◆ **Add Column Tab** (This tab is used when you want to derive new data)
- ◆ **View Tab** (Helpful for data understanding and debugging)

2. Queries Pane (Left Side)
  - ❖ The Queries Pane shows all queries (tables) in the current Power BI file.

#### What It Does:

1. Lists all loaded and referenced queries.
2. Helps manage multiple tables at once.
3. Allows:
  - a. Renaming queries
  - b. Duplicating queries
  - c. Deleting queries
  - d. Disabling load

3. Data Preview (Center Section)

- ❖ The Data Preview area shows a sample view of your data.

#### What It Does:

1. Displays rows and columns of selected query.
2. Shows how data looks after each transformation.
3. Allows:
  - a. Filtering
  - b. Sorting
  - c. Selecting columns

## POWER QUERY EDITOR

- d. Right-click transformations
4. M Formula Bar / Power Query Formula Bar
  - ❖ Power Query uses a language called M (Power Query Formula Language).
  - ❖ The Formula Bar shows the M expression for the selected step.
5. Query Settings Pane (Right Side)
  - ❖ This pane controls how the selected query is built.

### Sections Inside Query Settings:

- ◆ **Properties** (Name- Rename the query)
- ◆ **Applied Steps** (Shows step-by-step transformations applied to the query)

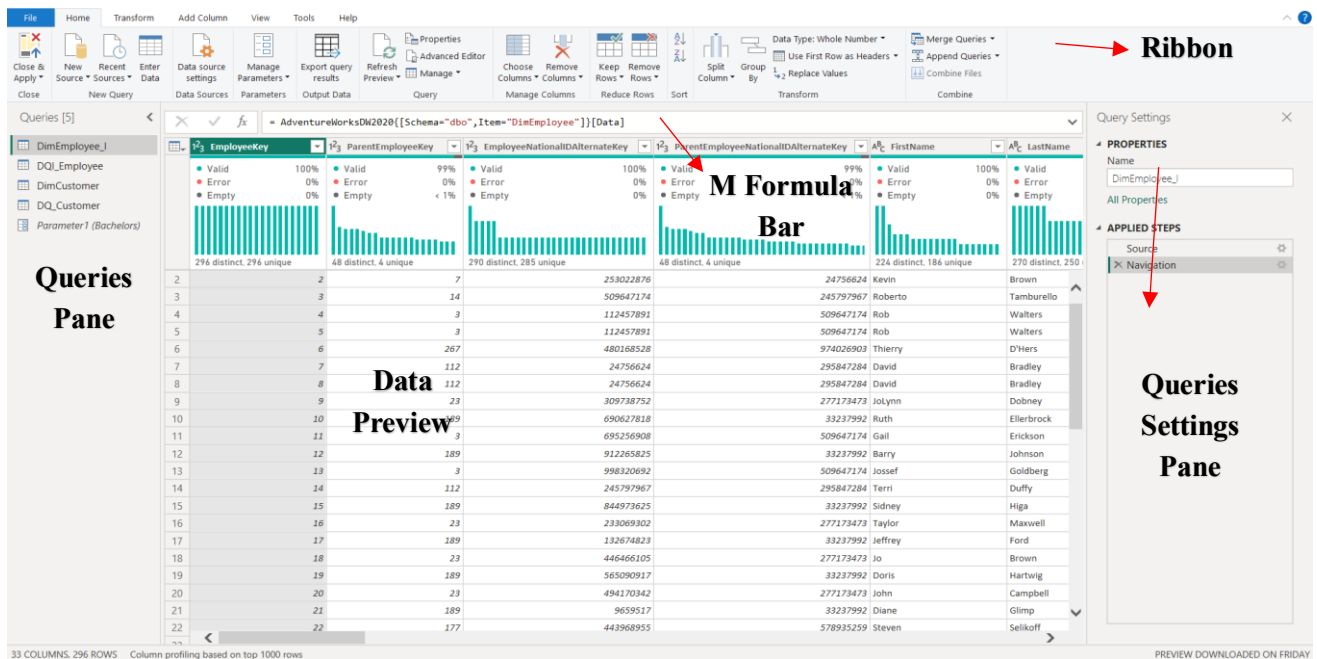


Fig: Power Query Editor

## HOME TAB

### 1. Home Tab- Detailed Explanation

The **Home tab** is mainly used for **data loading, basic transformations, and query management.**

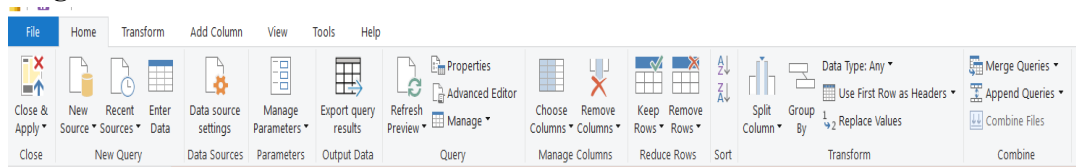


Fig 1: Home Tab

#### 1.1 Close & Apply

Used to **finalize transformations.**

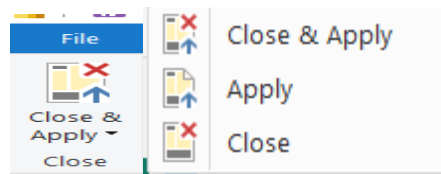


Fig 1.1: Close & Apply

##### 1.1.1 Close & Apply

- ❖ Saves all changes
- ❖ Loads transformed data into Power BI
- ❖ Closes Power Query Editor

##### 1.1.2 Apply

- ❖ Applies changes
- ❖ Keep Power Query Editor open

##### 1.1.3 Close

- ❖ Closes Power Query Editor
- ❖ Does not apply changes

#### 1.2 New Query

Used to create new query

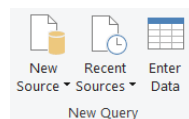


Fig 1.2: New Query

##### 1.2.1 New Source

- ❖ Connects to new data sources
- ❖ Example: Excel, SQL Server, CSV, Web

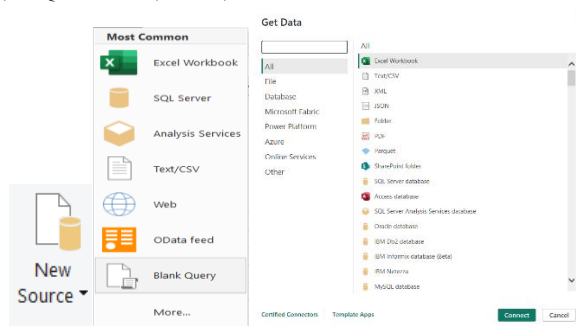


Fig 1.2.1: New Source

##### 1.2.2 Recent Sources

## POWER QUERY EDITOR

- ❖ Shows previously used data sources
- ❖ Saves connection time

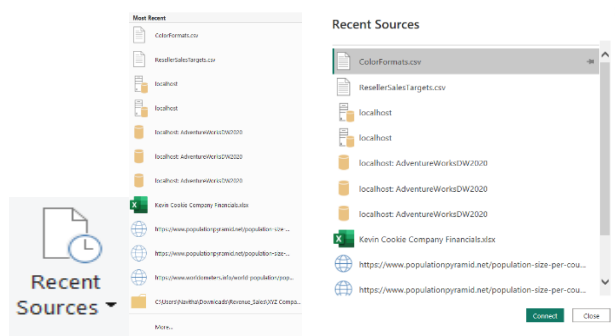


Fig 1.2.2: Recent Sources

### 1.2.3 Enter Data

- ❖ Manually type data
- ❖ Creates a table inside Power BI

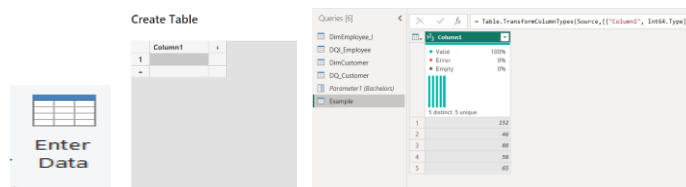


Fig 1.2.3: Enter Data

### 1.3 Data Sources

**Data Sources** in Power Query Editor represent the **origin of the data** used in a query.

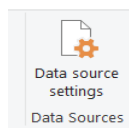


Fig 1.3: Data Sources

### 1.3.1 Data Source Settings

**Data Source Settings** is used to **view, manage, edit, and control** how Power Query connects to different data sources.

### a. Manage Credentials

### Purpose:

Used to **store, update, or change login credentials** for a data source.

**Credentials include:**

- ❖ Windows Authentication
- ❖ Database Authentication
- ❖ Microsoft Account
- ❖ Basic Authentication
- ❖ Anonymous

### b. Change Privacy Levels

### Purpose:

Controls how data from different sources can be **combined safely**.

### Privacy Levels:

- ❖ **Public** – Data can be shared freely
- ❖ **Organizational** – Internal company data
- ❖ **Private** – Highly sensitive data

### c. Edit or Clear Permissions

# POWER QUERY EDITOR

## Purpose:

Allows you to:

- ❖ Edit existing permissions
- ❖ Clear stored credentials
- ❖ Remove unused or incorrect data source access

## Use cases:

- ❖ Fix authentication issues
- ❖ Switch between users
- ❖ Remove old connections

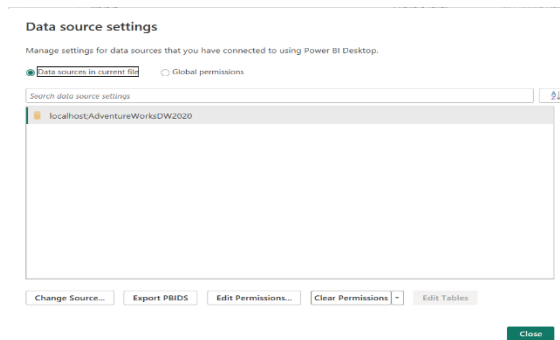


Fig 1.3.1: Data Source Settings

## 1.4 Parameters

Used to create **dynamic queries**.

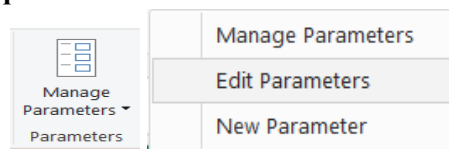


Fig 1.4: Parameters

### 1.4.1 Manage Parameters

- ❖ View all parameters.

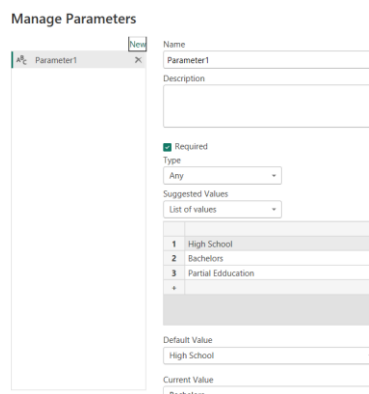


Fig 1.4.1: Manage Parameters

### 1.4.2 Edit Parameters

- ❖ Modify existing parameter values.

#### Edit Parameters



Fig 1.4.2: Edit Parameters

## 1.4.3 New Parameter

- ❖ Create new parameter
- ❖ Used for filters, file paths, dates, etc.

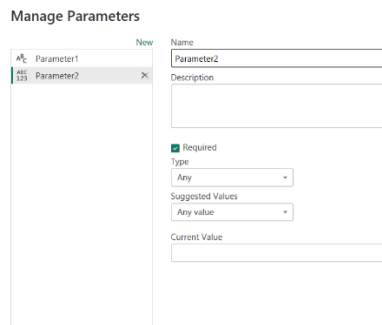


Fig 1.4.3: New parameter

## 1.5 Output Data

- ❖ Exports query output.
- ❖ Used for checking data externally.



Fig 1.5: Export Query Results

## 1.6 Query

- ❖ It is a group of options used to manage and control queries.
- ❖ Used to control how queries behave and execute.
- ❖ Mainly used for query maintenance.

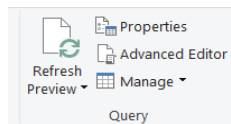


Fig 1.6: Query

### 1.6.1 Refresh Preview

- ❖ Reloads data preview.
- ❖ Does not apply changes.
- ❖ Used to see latest data.

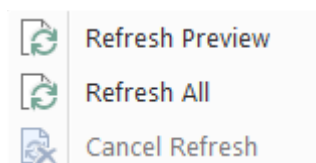


Fig 1.6.1: Refresh Preview

### 1.6.2 Refresh all

- ❖ Refreshes all queries.
- ❖ Updates data from source.
- ❖ Applies latest transformations.

### 1.6.3 Properties

- ❖ Allows renaming the query.
- ❖ Add description.
- ❖ Enable or disable query load.

#### 1.6.4 Advanced Editor

- ❖ Shows full M code of query.
- ❖ Used for advanced transformations.
- ❖ Allows manual editing of logic.

#### 1.7 Manage Columns

- ❖ Provides options to select or remove columns.
- ❖ Helps organize table structure.
- ❖ Used during initial data cleaning.

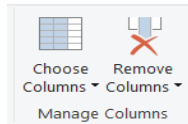


Fig 1.7: Manage Columns

##### 1.7.1 Choose Columns

- ❖ Select required columns only.
- ❖ Removes unselected columns.
- ❖ Improve data focus.

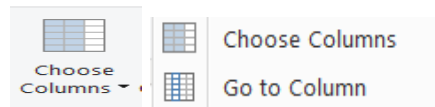


Fig 1.7.1: Choose Columns

##### 1.7.2 Go to Column

- ❖ Jumps directly to a selected column.
- ❖ Saves time in wide tables.
- ❖ Improves navigation.

##### 1.7.3 Remove Column

- ❖ Deletes selected columns.
- ❖ Reduces unnecessary data.
- ❖ Creates an applied step.

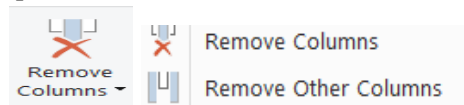


Fig 1.7.3: Remove Column

##### 1.7.4 Remove Other Columns

- ❖ Keeps selected columns.
- ❖ Removes all remaining columns.
- ❖ Useful for quick clean up.

#### 1.8 Reduce Rows

- ❖ Contains row-level filtering options.
- ❖ Used to clean data. Common in pre-processing stage.

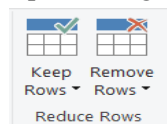


Fig 1.8: Reduce Rows

##### 1.8.1 Keep Rows

- ❖ Keeps selected rows.
- ❖ Removes all others.
- ❖ Used for focused analysis.



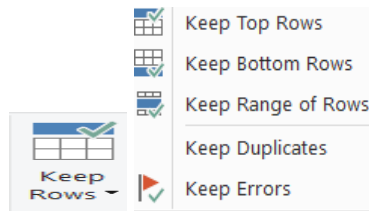


Fig 1.8.1: Keep Rows

#### 1.8.1.1 Keep Top Rows

- Keeps first N rows.
- Used for testing.
- Quick preview.

#### 1.8.1.2 Keep Bottom Rows

- Keeps last N rows.
- Useful for recent data.
- Maintains row order.

#### 1.8.1.3 Keep Range of Rows

- Keeps rows in a range.
- Allows start and end selection.
- Flexible filtering.

#### 1.8.1.4 Keep Duplicates

- Keeps only duplicate rows.
- Removes unique rows.
- Useful for data checks.

#### 1.8.1.5 Keep Errors

- Keeps rows with errors.
- Used for error analysis.
- Helps debugging.

### 1.8.2 Remove Rows

- ❖ Removes unwanted rows.
- ❖ Cleans data.
- ❖ Reduces dataset size.

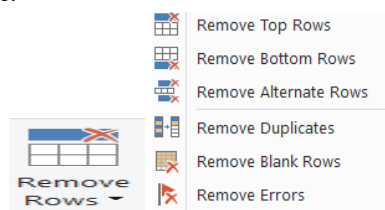


Fig 1.8.2: Remove Rows

#### 1.8.2.1 Remove Top Rows

- Deletes first N rows.
- Used to remove headers or junk.
- Common in Excel imports.

#### 1.8.2.2 Remove Bottom Rows

- Deleted last N rows.
- Removes totals or notes.
- Improves accuracy.

#### 1.8.2.3 Remove Duplicates

- Deletes duplicate rows.

- Keeps unique values.
- Improves data integrity.

### 1.8.2.4 Remove Blank

- Removes empty rows.
- Cleans dataset.
- Prevents errors.

### 1.8.2.5 Remove Errors

- Deletes error rows.
- Improves refresh stability.
- Cleaner output.

### 1.8.2.6 Remove Alternative Rows

- Removes every nth row.
- Used for patterned data.
- Advanced clean up.

## 1.9 Sort

- ❖ Contains sorting options.
- ❖ Sorts based on column values.
- ❖ Creates an applied step.

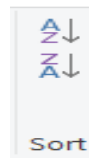


Fig 1.9: Sort

## 1.10 Transform

- ❖ Contains common transformation tools.
- ❖ Used after initial cleaning.
- ❖ Prepares data for analysis.

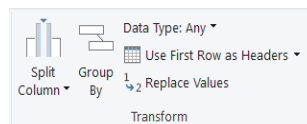


Fig 1.10: Transform

### 1.10.1 Split Columns

- ❖ Splits one column into multiple columns.
- ❖ Uses delimiter or position.
- ❖ Useful for names, codes.

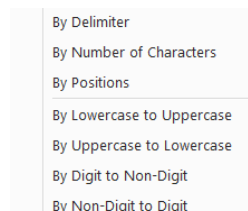


Fig 1.10.1: Split Columns

### 1.10.2 Group By

- ❖ Groups rows based on a column.
- ❖ Performs aggregations.
- ❖ Used for summaries.

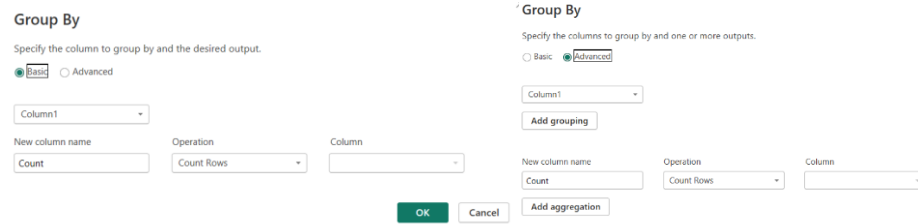


Fig 1.10.2: Group By

## 1.10.3 Data Type

- ❖ Changes column data type.
- ❖ Ensures correct calculations and Prevents errors.

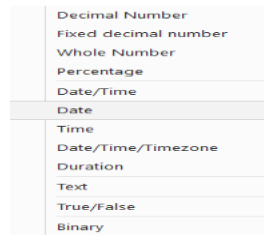


Fig 1.10.3: Data Type

## 1.10.4 Use First Row as Header

- ❖ Converts first row into column names.
- ❖ Improves table structure.
- ❖ Common for Excel imports.

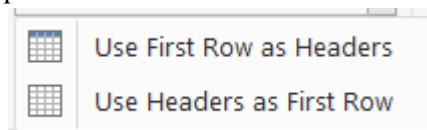


Fig 1.10.4: Use First Row as Header

## 1.10.5 Replace Values

- ❖ Replaces specific values.
- ❖ Fixes incorrect data.
- ❖ Standardizes data.

## 1.11 Combine

- ❖ Contains merge and append options.
- ❖ Combines queries logically.
- ❖ Creates new combined queries.

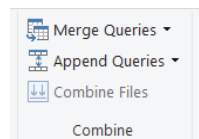


Fig 1.11: Combine

### 1.11.1 Merge Columns

- ❖ Combines columns into one.
- ❖ Uses delimiter.
- ❖ Useful for full names.

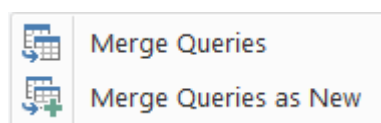


Fig 1.11.1: Merge Columns

### 1.11.2 Append Columns

## POWER QUERY EDITOR

- ❖ Adds rows from another table.
- ❖ Structure must be similar.
- ❖ Used for monthly or yearly data.

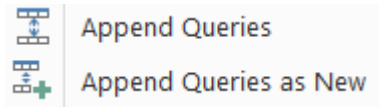


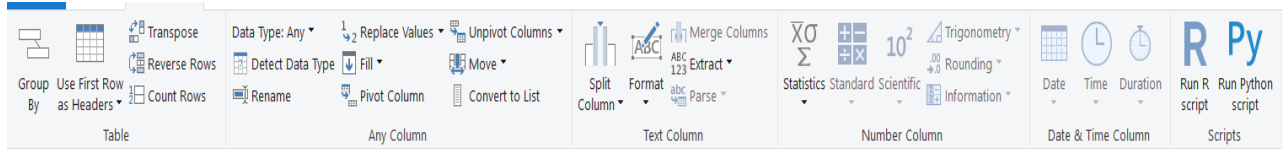
Fig 1.11.2: Append Columns

## TRANSFORM TAB

### 2. Transform Tab

Used to modify existing data in columns and tables.

- ❖ Changes data values and structure.
- ❖ Works on selected columns or tables.
- ❖ Does not create new columns (only modifies existing ones).



#### 2.1 Table

Operations applied to the entire table.

- ❖ Changes table structure.
- ❖ Affects rows and columns together.
- ❖ Used for layout-level transformations.

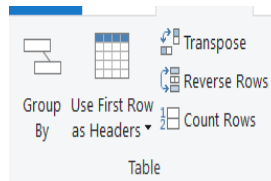


Fig 2.1: Table

##### 2.1.1 Transpose

- ❖ Converts rows into columns.
- ❖ Converts columns into rows.
- ❖ Useful when data layout is incorrect.

##### 2.1.2 Count Rows

- ❖ Counts total number of rows.
- ❖ Creates a single value output.
- ❖ Useful for data validation.

##### 2.1.3 Reverse Rows

- ❖ Reverses row order.
- ❖ Last row becomes first.
- ❖ Useful for sorting adjustments.

#### 2.2 Any Column

Operations that apply to any data type column.

- ❖ Works on text, number, and date columns.
- ❖ Used for common column operations.
- ❖ Flexible and widely used.

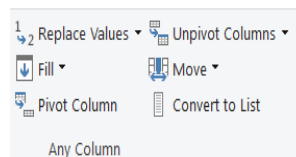


Fig 2.2: Any Column

##### 2.2.1 Detect Data Type

- ❖ Automatically detects column data types.
- ❖ Converts text to number or dates.

- ❖ Reduces manual data type errors.

### 2.2.2 Rename

- ❖ Changes column name.
- ❖ Improves readability.
- ❖ Does not affect data values.

### 2.2.3 Replace Errors

- ❖ Replaces error values.
- ❖ Prevents refresh failures.
- ❖ Improve data quality.

### 2.2.4 Fill

Used to fill missing values.

- ❖ Copies nearby values.
- ❖ Used for hierarchical data.
- ❖ Prevents blanks.



Fig 2.2.4: Fill

#### 2.2.4.1 Fill Down

- Fills blank cells downwards.
- Uses value from above row.
- Common in grouped data.

#### 2.2.4.2 Fill Up

- Fills blank cells upward.
- Uses values from below row.
- Useful for reverse-filled data.

### 2.2.5 Pivot Columns

- ❖ Converts unique values into columns.
- ❖ Aggregates data.
- ❖ Used for summarization.

### 2.2.6 Unpivot Columns

- ❖ Converts columns into rows.
- ❖ Normalizes wide tables.
- ❖ Makes data analysis-friendly.

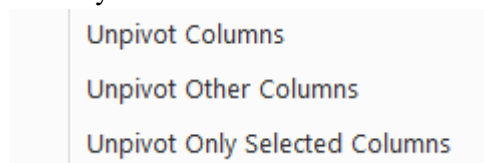


Fig 2.2.6: Unpivot Columns

#### 2.2.6.1 Unpivot other columns

- Keeps selected columns fixed.
- Unpivots remaining columns.
- Commonly used option.

#### 2.2.6.2 Unpivot columns

- Unpivots selected columns.

- Converts column headers into values.
- Simplifies table structure.

### 2.2.6.3 Unpivot only selected columns

- Works only on chosen columns.
- Full user control.
- Prevents unwanted changes.

### 2.2.7 Move

Changes column position.

- ❖ Reorders columns.
- ❖ Improves readability.
- ❖ No data change.

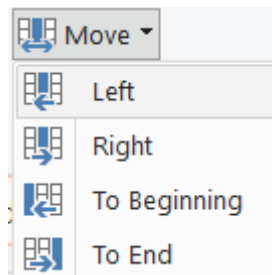


Fig 2.2.7: Move

#### 2.2.7.1 Left

- Moves column to the left.
- One position at a time.
- Used for rearranging.

#### 2.2.7.2 Right

- Moves column to the right.
- Simple repositioning.
- Layout adjustment.

#### 2.2.7.3 To Beginning

- Moves column to first position.
- Improves column priority.
- Useful for key columns.

#### 2.2.7.4 To End

- Moves column to last position.
- Keeps main column visible.
- Layout organization.

## 2.3 Text Column

Used for text-based transformations.

- ❖ Applies only to text data.
- ❖ Used for cleaning strings.
- ❖ Improves consistency.

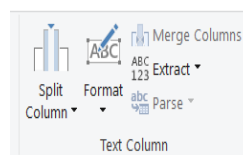


Fig 2.3: Text Column

### 2.3.1 Format

- ❖ Uppercase, lowercase, capitalize.

## POWER QUERY EDITOR

- ❖ Trim and clean spaces.
- ❖ Improves text uniformity.

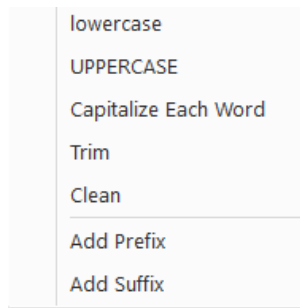


Fig 2.3.1: Format

### 2.3.2 Extract

- ❖ Extracts part of text.
- ❖ Example: First, last characters.
- ❖ Useful for codes and IDs

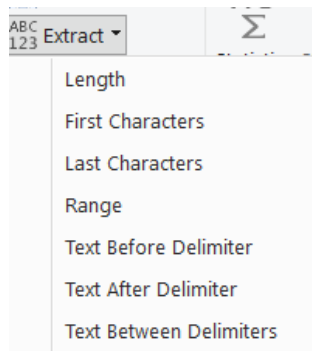


Fig 2.3.2: Extract

### 2.3.3 Parse

- ❖ Converts text into structured data.
- ❖ Example: data from text.
- ❖ Improves data usability.

## 2.4 Number Column

- ❖ Used for numeric calculations.
- ❖ Applies only to number columns
- ❖ Performs mathematical operations
- ❖ Useful for calculations

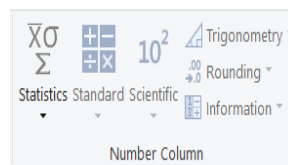


Fig 2.4: Number Column

### 2.4.1 Statistics

- ❖ Sum, average, min, max
- ❖ Used for analysis
- ❖ Quick aggregation



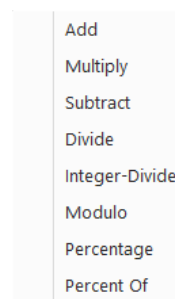


Sum
Minimum
Maximum
Median
Average
Standard Deviation
Count Values
Count Distinct Values

Fig 2.4.1: Statistics

### 2.4.2 Standard

- ❖ Add, subtract, multiply, divide
- ❖ Used for value adjustments
- ❖ Simple math operations

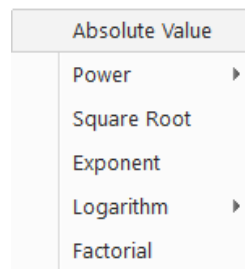


Add
Multiply
Subtract
Divide
Integer-Divide
Modulo
Percentage
Percent Of

Fig 2.4.2: Standard

### 2.4.3 Scientific

- ❖ Logarithm and power functions
- ❖ Used for advanced calculations
- ❖ Mostly technical use

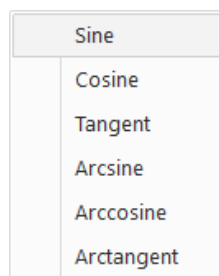


Absolute Value	
Power	▸
Square Root	
Exponent	
Logarithm	▸
Factorial	

Fig 2.4.3: Scientific

### 2.4.4 Trigonometry

- ❖ Sin, cos, tan functions
- ❖ Used in scientific data
- ❖ Less common in business data



Sine	
Cosine	
Tangent	
Arcsine	
Arccosine	
Arctangent	

Fig 2.4.4: Trigonometry

#### 2.4.5 Rounding

- ❖ Round up or down
- ❖ Fix decimal places
- ❖ Improves data presentation



Fig 2.4.5: Rounding

#### 2.4.6 Information

- ❖ Checks numeric properties
- ❖ Identifies errors or nulls
- ❖ Useful for validation

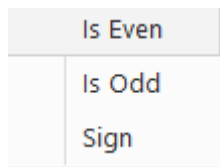


Fig 2.4.6: Information

### 2.5 Date & Time Column

Used for date and time operations.

- ❖ Extracts date parts
- ❖ Improves time-based analysis
- ❖ Works only on date/time data

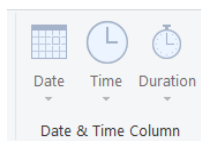


Fig 2.5: Date & Time Column

#### 2.5.1 Date

- ❖ Extract year, month, day
- ❖ Used for time analysis
- ❖ Improves reporting

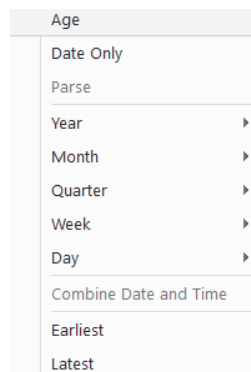


Fig 2.5.1: Date

#### 2.5.2 Time

- ❖ Extract hour, minute, second
- ❖ Used for time tracking
- ❖ Useful in logs and events

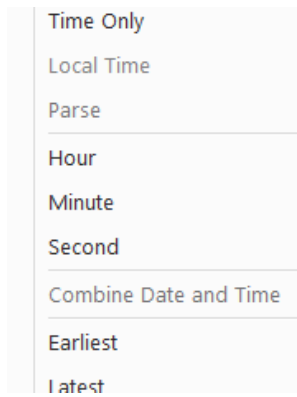


Fig 2.5.2: Time

### 2.6 Scripts

Used for advanced scripting.

- ❖ Supports external languages
- ❖ Used for complex logic
- ❖ Advanced feature



Fig 2.6: Scripts

#### 2.6.1 Run R script

- ❖ Executes R code
- ❖ Used for statistical analysis
- ❖ Requires R installation

#### 2.6.2 Run Python Script

- ❖ Executes Python code
- ❖ Used for advanced data processing
- ❖ Requires Python installation

## ADD COLUMN TAB

### 3. Add Column

Used to create new columns without changing original data.

- ❖ Helps derive new information
- ❖ Original columns remain unchanged
- ❖ Each action creates a new applied step

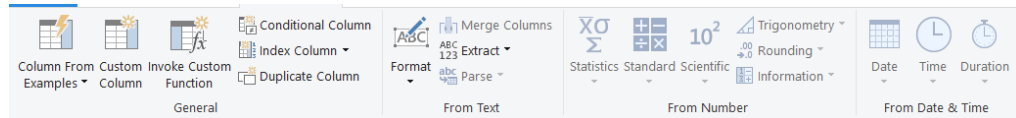


Fig 3: Add column

#### 3.1 General

Contains common options to add columns.

- ❖ Used for basic column creation
- ❖ Supports automatic and manual methods
- ❖ Useful for calculated columns

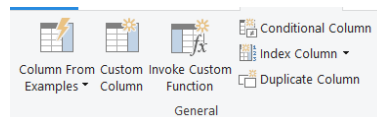


Fig 3.1: General

##### 3.1.1 Column From Examples

Creates new columns by observing examples.

- ❖ No coding required
- ❖ Auto-detects patterns
- ❖ Very beginner-friendly

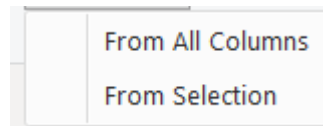


Fig 3.1.1: Column From Examples

##### 3.1.1.1 From selection

- Uses selected column(s) only
- User provides sample output
- Power Query generates logic

##### 3.1.1.2 From all columns

- Uses data from all columns
- Identifies complex patterns
- Suitable for advanced extraction

##### 3.1.2 Custom Column

Creates a column using formulas.

- ❖ Uses Power Query (M) expressions
- ❖ Allows advanced logic
- ❖ Highly flexible option

## POWER QUERY EDITOR

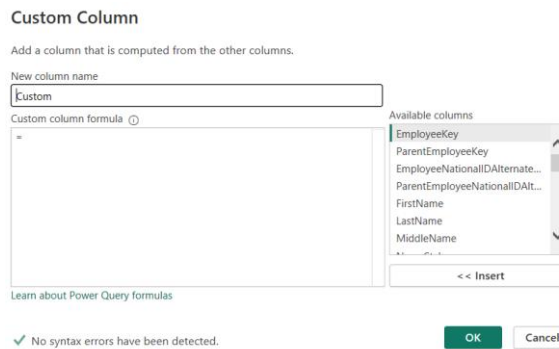


Fig 3.1.2: Custom Column

### 3.1.3 Invoke Custom Column

Applies a custom function to data.

- ❖ Reuses existing functions
- ❖ Used for repetitive logic
- ❖ Improves query efficiency

## Invoke Custom Function

Invoke a custom function defined in this file for each row.

New column name

Function query

Fig 3.1.3: Invoke Custom Column

### 3.1.4 Conditional Column

Creates columns using conditions.

- ❖ Based on IF–ELSE logic
- ❖ Easy to configure
- ❖ No coding required

### 3.1.5 Index Column

Adds sequential numbers to rows.

- ❖ Used for row numbering
- ❖ Helps in sorting and referencing
- ❖ Automatically increments values

From 0
From 1
Custom...

Fig 3.1.5: Index Column

#### 3.1.5.1 From 0

- Index starts from 0
- Common in programming logic
- Useful for technical processing

#### 3.1.5.2 From 1

- Index starts from 1

- Easy for users to read
- Common in reports

### 3.1.5.3 Custom

- User-defined start value
- Flexible indexing
- Used for special cases

### 3.1.6 Duplicate Column

Creates a copy of a column.

- ❖ Original column remains safe
- ❖ Useful before transformations
- ❖ Prevents data loss

## VIEW TAB

### 4. View Tab

Used to control layout and data analysis view.

- ❖ Improves data understanding
- ❖ Helps in debugging
- ❖ No data changes occur

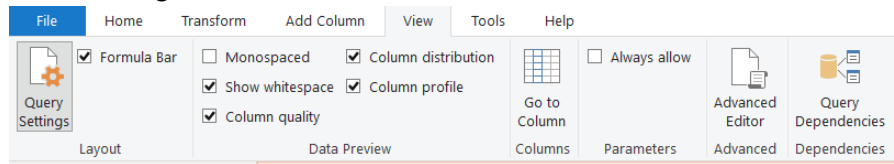


Fig 4: View Tab

#### 4.1 Layout

Controls display elements.

- ❖ Manages screen layout
- ❖ Shows or hides panes
- ❖ Improves workspace clarity

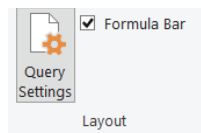


Fig 4.1: Layout

##### 4.1.1 Query settings

- ❖ Shows applied steps
- ❖ Helps manage transformations
- ❖ Can be hidden or shown

##### 4.1.2 Formula bar

- ❖ Displays M code for steps
- ❖ Allows editing formulas
- ❖ Useful for advanced users

#### 4.2 Data Preview

Controls data analysis visuals.

- ❖ Helps understand data quality
- ❖ Displays column statistics
- ❖ Used for inspection

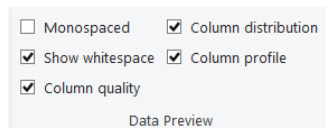


Fig 4.2: Data Preview

##### 4.2.1 Monospaced

- ❖ Uses fixed-width font
- ❖ Improves data alignment
- ❖ Useful for text inspection

##### 4.2.2 Show Whitespace

- ❖ Displays spaces and tabs
- ❖ Helps detect hidden characters
- ❖ Useful for text cleaning

##### 4.2.3 Column Quality

- ❖ Shows valid and error values
- ❖ Uses visual indicators
- ❖ Helps detect issues quickly

#### 4.2.4 Column distribution

- ❖ Shows value frequency
- ❖ Highlights unique values
- ❖ Helps detect outliers

#### 4.2.5 Column Profile

- ❖ Shows min, max, average
- ❖ Displays data statistics
- ❖ Useful for analysis

### 4.3 Columns

Column navigation tools.

- ❖ Helps locate columns quickly
- ❖ Useful in wide tables
- ❖ Improves navigation

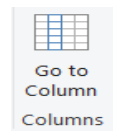


Fig 4.3: Columns

#### 4.3.1 Go to column

- ❖ Jumps to selected column
- ❖ Saves scrolling time
- ❖ Useful for large datasets

### 4.4 Parameters

Controls parameter behavior.

- ❖ Used in dynamic queries
- ❖ Affects filtering and logic
- ❖ Helps automation



Fig 4.4: Parameters

#### 4.4.1 Always allow

- ❖ Always allows parameter usage
- ❖ Prevents execution issues
- ❖ Useful in scheduled refresh

### 4.5 Advanced

Advanced configuration tools.

- ❖ Used by experienced users
- ❖ Allows deeper control
- ❖ Supports complex logic

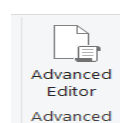


Fig 4.5: Advanced

#### 4.5.1 Advanced Editor



- ❖ Shows full M code
- ❖ Allows manual coding
- ❖ Used for complex transformations

### 4.6 Dependencies

Shows query relationships.

- ❖ Displays query flow
- ❖ Helps understand dependencies
- ❖ Useful for debugging

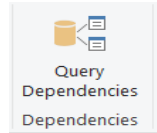


Fig 4.6: Dependencies

#### 4.6.1 Query Dependencies

- ❖ Visual map of queries
- ❖ Shows data flow
- ❖ Helps impact analysis

## **TOOLS TAB**

### **5. Tools Tab**

Used for performance analysis.

- ❖ Monitors query execution
- ❖ Helps optimize performance
- ❖ Mainly for advanced users

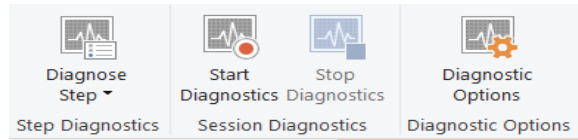


Fig 5: Tools Tab

#### **5.1 Step Diagnostics**

- ❖ Analyzes individual steps
- ❖ Measures execution time
- ❖ Identifies slow steps

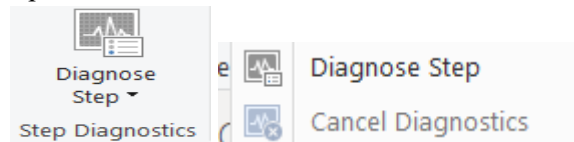


Fig 5.1: Step Diagnostics

#### **5.2 Session Diagnostics**

- ❖ Tracks entire session
- ❖ Records query activity
- ❖ Used for troubleshooting

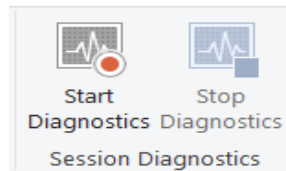


Fig 5.2: Session Diagnostics

##### **5.2.1 Start Diagnostics**

- ❖ Begins performance tracking
- ❖ Captures execution details
- ❖ Used before refresh

##### **5.2.2 Stop Diagnostics**

- ❖ Stops tracking
- ❖ Saves diagnostic data
- ❖ Used after analysis

#### **5.3 Diagnostics options**

- ❖ Configures diagnostics settings
- ❖ Controls data capture
- ❖ Improves analysis accuracy

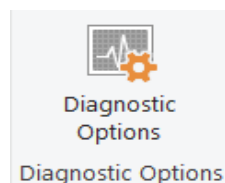


Fig 5.3: Diagnostics options

## HELP TAB

### 6. Help Tab

Provides learning and support.

- ❖ Helps users understand Power Query
- ❖ Provides official resources
- ❖ Supports skill improvement

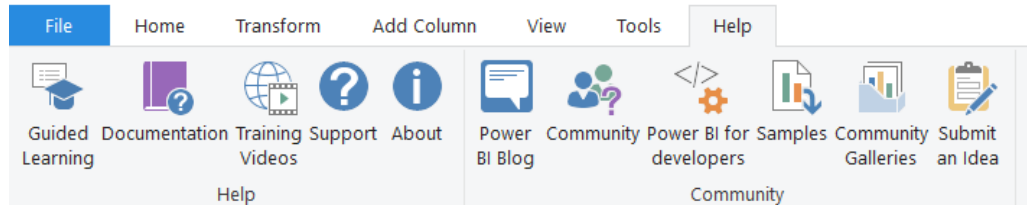


Fig 6: Help Tab

#### 6.1 Help

Main help resources.

- ❖ Learning materials
- ❖ Official documentation
- ❖ Product information

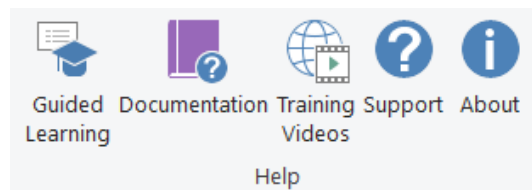


Fig 6.1: Help Tab

##### 6.1.1 Guided Learning

- ❖ Step-by-step tutorials
- ❖ Beginner-friendly
- ❖ Covers basics

##### 6.1.2 Documentation

- ❖ Official Microsoft docs
- ❖ Detailed explanations
- ❖ Reference material

##### 6.1.3 Training Videos

- ❖ Video-based learning
- ❖ Practical demonstrations
- ❖ Easy to follow

##### 6.1.4 Support

- ❖ Technical help options
- ❖ Issue resolution
- ❖ Microsoft support access

##### 6.1.5 About

- ❖ Version information
- ❖ Product details
- ❖ License information

#### 6.2 Community

Community-driven resources.

- ❖ User discussions
- ❖ Shared solutions
- ❖ Knowledge exchange

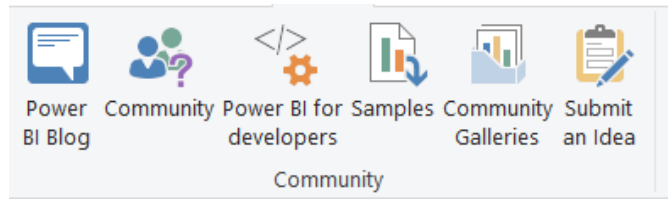


Fig 6.2: Community

### 6.2.1 Power BI Blog

- ❖ Product updates
- ❖ New features
- ❖ Announcements

### 6.2.2 Community

- ❖ Forums for discussion
- ❖ Ask and answer questions
- ❖ Peer support

### 6.2.3 Power BI for developers

- ❖ Developer tools
- ❖ APIs and embedding
- ❖ Advanced integrations

### 6.2.4 Samples

- ❖ Sample datasets
- ❖ Example reports
- ❖ Learning resources

### 6.2.5 Community Galleries

- ❖ Shared dashboards
- ❖ Design inspiration
- ❖ Best practices

### 6.2.6 Submit an idea

- ❖ Suggest new features
- ❖ Provide feedback
- ❖ Participate in product improvement