

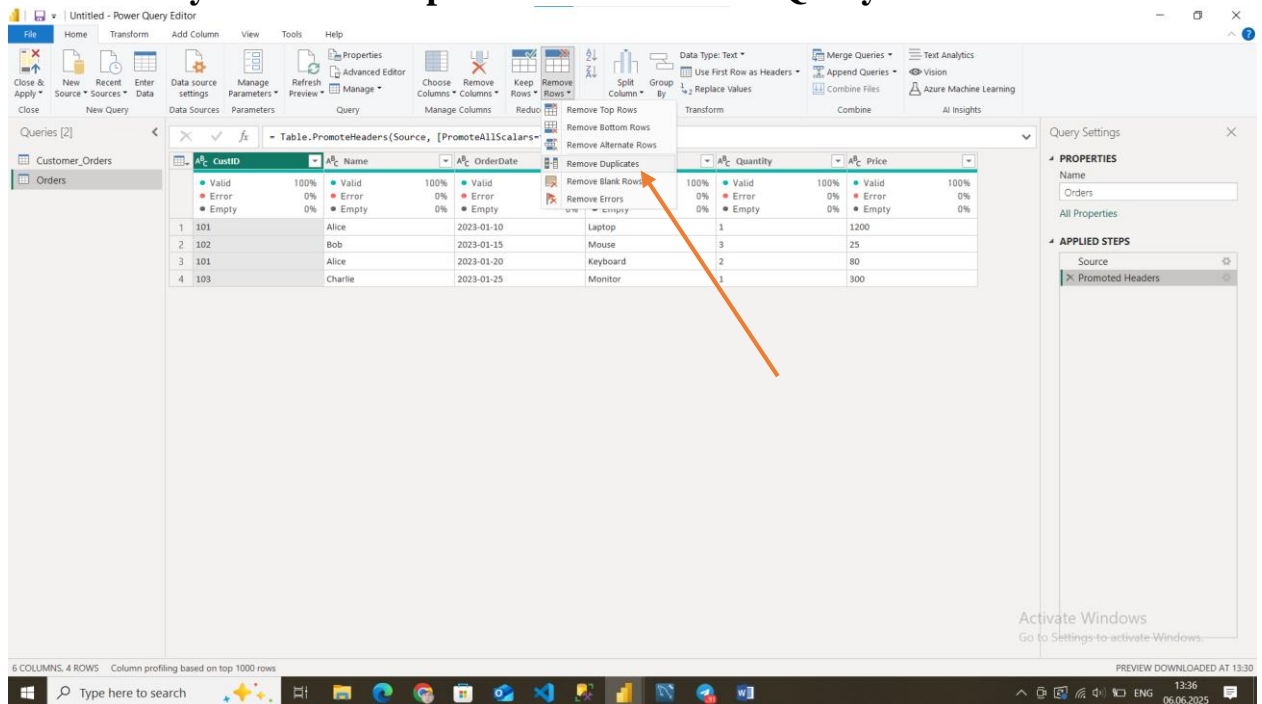
1. What is the purpose of the "Applied Steps" pane in Power Query?

Answer:

The "**Applied Steps**" pane tracks every transformation you make to your data in order. It allows you to:

- Review each step of your query.
- Edit, reorder, or delete specific steps.
- Debug errors by seeing where something went wrong.
- Ensure transformations are performed in the correct sequence.

2. How do you remove duplicate rows in Power Query?



Go to the **Home** tab.

- Select the columns you want to check for duplicates (or no selection to consider entire rows).
- Click **Remove Rows** → **Remove Duplicates**.

3. What does the Filter icon do in Power Query?

Answer:

The **Filter** icon (a little funnel) next to a column header lets you:

- Select or deselect specific values.
- Apply number/date/text filters.
- Keep or exclude nulls or blanks.

4. How would you rename a column from CustID to CustomerID?

The first screenshot shows the Power Query Editor with a table named 'Table.PromoteHeaders(Source, [PromoteAllScalars=true])'. The table has columns: CustID, OrderDate, Product, Quantity, and Price. A right-click context menu is open over the 'CustID' column header, with the 'Rename' option selected. The second screenshot shows the same table after the column has been renamed to 'CustomerID'. The table now has columns: CustomerID, Name, OrderDate, Product, Quantity, and Price. The data rows are as follows:

	CustomerID	Name	OrderDate	Product	Quantity	Price
1	101	Alice	2023-01-10	Laptop	1	1200
2	102	Bob	2023-01-15	Mouse	3	25
3	101	Alice	2023-01-20	Keyboard	2	80
4	103	Charlie	2023-01-25	Monitor	1	300

- Right-click the CustID column header.
- Choose **Rename**.
- Type CustomerID and press Enter.

5. What happens if you click "Close & Apply" in Power Query?

- The query editor closes.
- All applied transformations are saved.
- The modified data is loaded into the Power BI data model

6. Remove all rows where Quantity is less than 2.

The first screenshot shows the Power Query Editor with a table named 'Table.TransformColumnTypes(*Renamed Columns*,{{"Quantity", Int64.Type}})'. The table has columns: CustomerID, Name, OrderDate, Product, Quantity, and Price. The 'Quantity' column is selected, and the 'Number Filters' menu is open, showing the 'Greater Than Or Equal To...' option.

The second screenshot shows the 'Filter Rows' dialog box. The 'Basic' tab is selected, and the condition is set to 'is greater than or equal to' with the value '2' entered in the input field. The 'And' radio button is selected.

Table data from the first screenshot:

CustomerID	Name	OrderDate	Product	Quantity	Price
101	Alice	2023-01-10		1200	
102	Bob	2023-01-15		25	
101	Alice	2023-01-20		80	
103	Charlie	2023-01-25			

- Click the drop-down next to **Quantity**.
- Choose **Number Filters** → **Greater Than or Equal To** → enter 2.
- Click **OK**.

7. Split the OrderDate column into separate Year, Month, and Day columns.

- Select OrderDate.
- Go to **Transform** → **Split Column** → **By delimiter** → **OK**

The screenshot shows the Power Query Editor interface. The main data table has the following columns: CustomerID, Name, Product, Quantity, and Price. The data rows are as follows:

CustomerID	Name	Product	Quantity	Price
101	Alice	Laptop	1	1200
102	Bob	Mouse	3	25
101	Alice	Keyboard	2	80
103	Charlie	Monitor	1	300

The 'Split Column by Delimiter' dialog box is open, showing the 'Split at' section with the following options:

- ☐ Left-most delimiter
- ☐ Right-most delimiter
- ☒ Each occurrence of the delimiter

The 'Quote Character' is set to '"'. The 'Split using special characters' checkbox is unchecked.

8. Replace all Mouse entries in the Product column with Computer Mouse.
Answer:

- Select Product column.
- Go to **Transform** tab → **Replace Values**.
- Find: Mouse
- Replace with: Computer Mouse
- Click **OK**.

The screenshot shows the Power Query Editor interface. The main area displays a table with 6 columns and 4 rows of data. The columns are CustomerID, Name, OrderDate, Product, Quantity, and Price. The data is as follows:

	CustomerID	Name	OrderDate	Product	Quantity	Price
1	101	Alice	10.01.2023	Laptop	1	1200
2	102	Bob	15.01.2023	Mouse	3	25
3	101	Alice	20.01.2023	Keyboard	2	80
4	103	Charlie	25.01.2023	Monitor	1	300

A 'Replace Values' dialog box is open, showing the 'Value To Find' as 'Mouse' and the 'Replace With' as 'Computer Mouse'. The dialog also includes an 'Advanced options' link and 'OK' and 'Cancel' buttons.

The right sidebar shows the 'Query Settings' pane with the 'Properties' tab selected, displaying the name 'Orders' and the 'Applied Steps' list.

9. Sort the table by OrderDate (newest first).

Answer:

- Click the drop-down next to **OrderDate**.
- Choose **Sort Descending** (↓).

The top screenshot shows the Power Query Editor with a table containing 4 rows and 6 columns: CustomerID, Name, OrderDate, Product, Quantity, and Price. A context menu is open over the 'OrderDate' column, showing options like 'Sort Ascending', 'Sort Descending', 'Clear Sort', 'Clear Filter', 'Remove Empty', and 'Date Filters'. The 'Sort Descending' option is selected.

The bottom screenshot shows the same table after sorting by 'OrderDate' in descending order. The data is now sorted by date, with the latest date (25.01.2023) at the top.

CustomerID	Name	OrderDate	Product	Quantity	Price
103	Charlie	25.01.2023	Monitor	1	300
101	Alice	20.01.2023	Keyboard	2	80
102	Bob	15.01.2023	Computer Mouse	3	25
101	Alice	10.01.2023	Laptop	1	1200

10. How would you handle null values in the Price column?

Answer:

Options:

- Or, remove rows with nulls:
 - **Home → Remove Rows → Remove Blank Rows.**
 - Or use filter to exclude null values.

The top screenshot shows the Power Query Editor with the following M-code in the formula bar:

```
Table.ReplaceValue(#"Changed Type", "Mouse", "Computer Mouse", ReplaceValue.Kind.Replace, "Product")
```

The bottom screenshot shows the Power Query Editor with the following M-code in the formula bar:

```
Table.SelectRows(#"Replaced Value", each not List.IsEmpty(List.RemoveMatchingItems(Record.FieldValues(_), {"", null})))
```

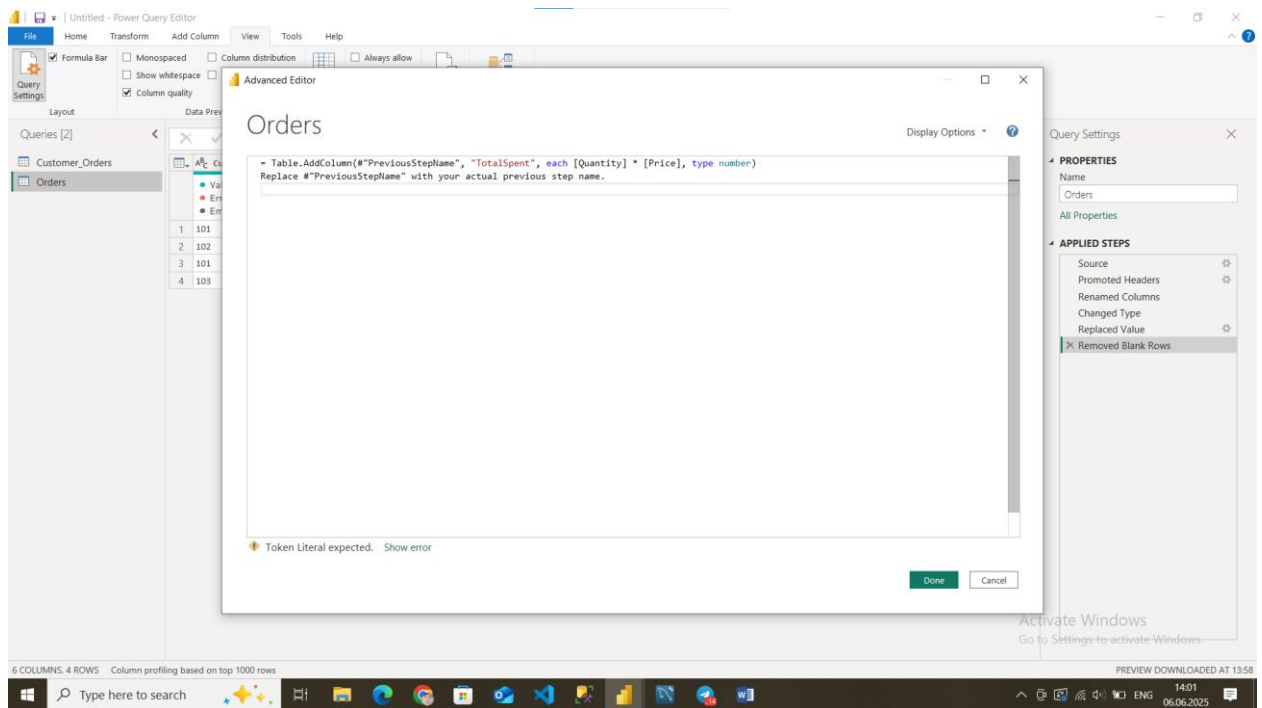
Both screenshots show a table with 6 columns and 4 rows of data. The table data is as follows:

CustomerID	Name	OrderDate	Product	Quantity	Price
101	Alice	10.01.2023	Laptop	1	1200
102	Bob	15.01.2023	Computer Mouse	3	25
101	Alice	20.01.2023	Keyboard	2	80
103	Charlie	25.01.2023	Monitor	1	300

11. Write custom M-code to add a column calculating TotalSpent = Quantity * Price.

= Table.AddColumn(#"PreviousStepName", "TotalSpent", each [Quantity] * [Price], type number)

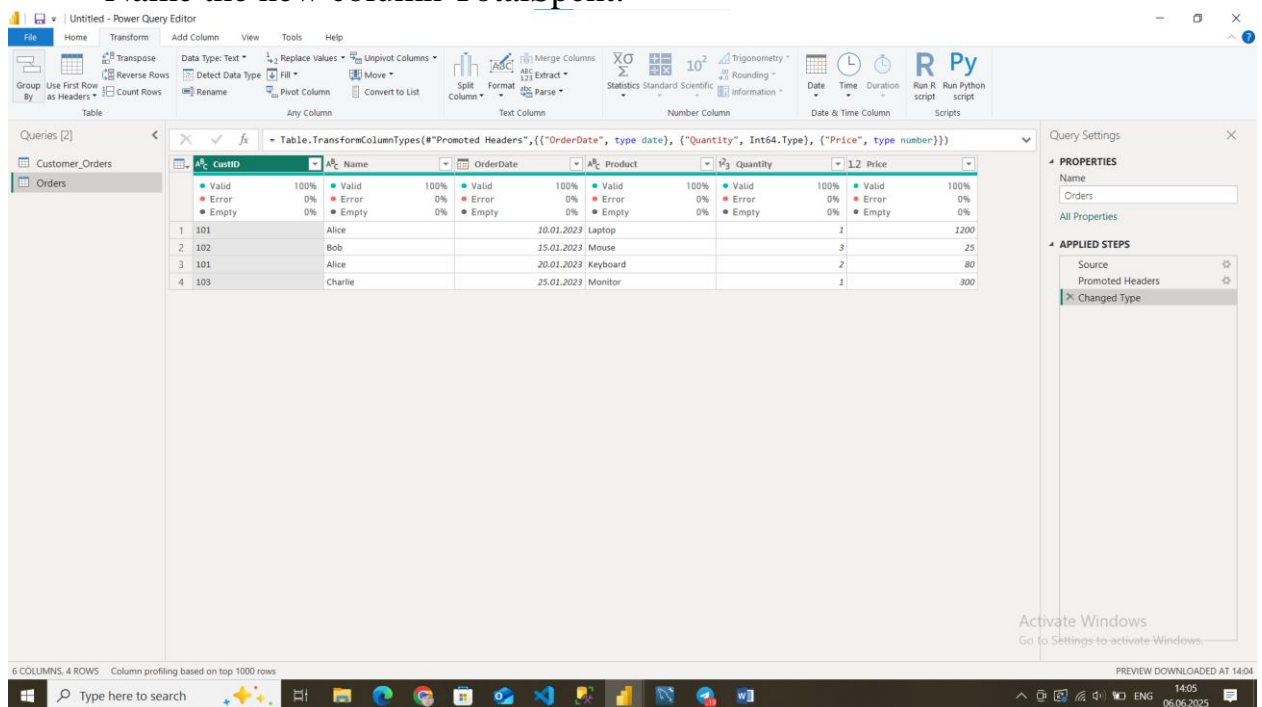
Replace #"PreviousStepName" with your actual previous step name.

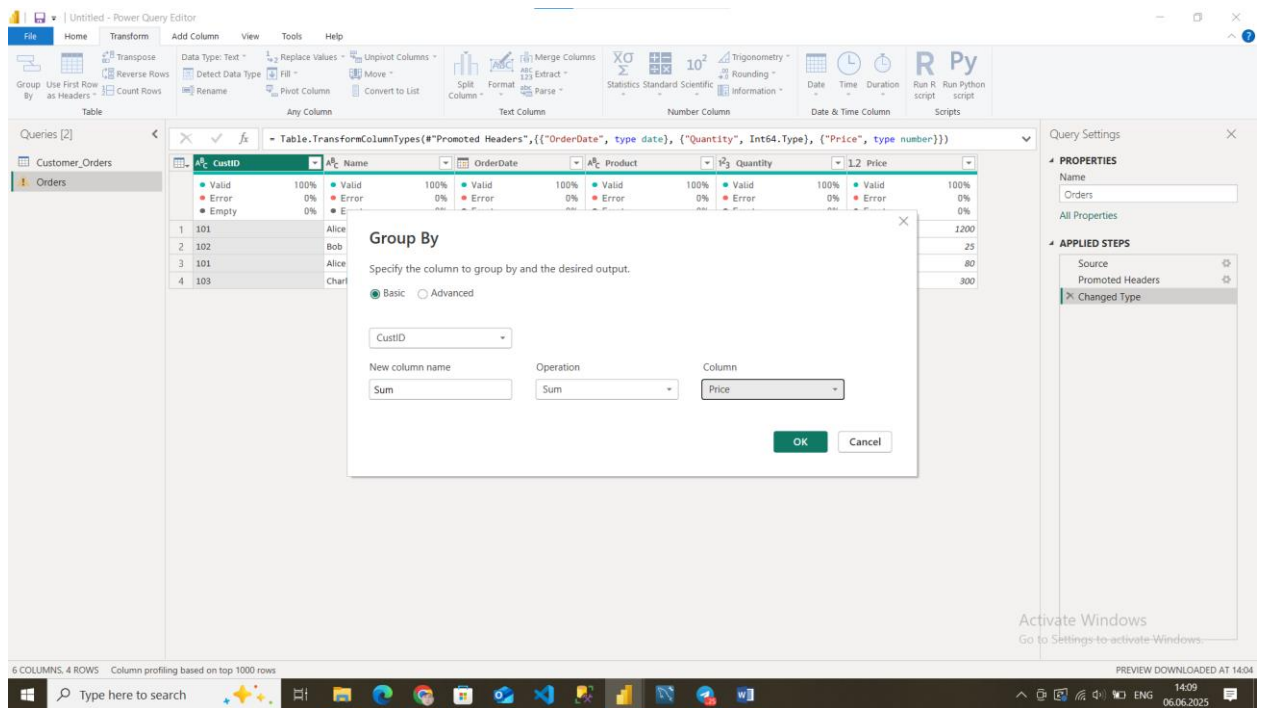


12. Group the table by CustID to show total spending per customer.

Answer:

- Select CustID column.
- **Transform** → **Group By**.
- Group By: CustID.
- Operation: **Sum** on TotalSpent (or $\text{Quantity} \times \text{Price}$ if not yet calculated).
- Name the new column TotalSpent.





13. Fix inconsistent date formats (e.g., 01/10/2023 vs. 2023-01-10) in OrderDate.

Answer:

- Select OrderDate.
- **Transform** → **Data Type** → **Date**.

Power Query automatically parses various date formats into a consistent internal date type.

The top screenshot shows the Power Query Editor with the following data table:

	CustID	Name	OrderDate	Product	Quantity	Price
1	101	Alice		top	1	1200
2	102	Bob		use	3	25
3	101	Alice		board	2	80
4	103	Charlie		itor	1	300

The bottom screenshot shows the same data table with the following changes:

	CustID	Name	OrderDate	Product	Quantity	Price
1	101	Alice	10.01.2023	Laptop	1	1200
2	102	Bob	15.01.2023	Mouse	3	25
3	101	Alice	20.01.2023	Keyboard	2	80
4	103	Charlie	25.01.2023	Monitor	1	300

14. Create a conditional column: Label orders as "High Value" if Price > 100.

Answer:

- **Add Column → Conditional Column.**
- **Column Name: OrderLabel.**
- **If Price is greater than 100, then "High Value", else "Regular".**

The top screenshot shows the Power Query Editor with a table containing 6 columns and 4 rows. The columns are: CusID, Name, OrderDate, Product, Quantity, and Price. The data is as follows:

	CusID	Name	OrderDate	Product	Quantity	Price
1	101	Alice	10.01.2023	Laptop	1	1200
2	102	Bob	15.01.2023	Mouse	3	25
3	101	Alice	20.01.2023	Keyboard	2	80
4	103	Charlie	25.01.2023	Monitor	1	300

The bottom screenshot shows the 'Add Conditional Column' dialog box. The condition is: 'If Price is greater than 100, then High Value, else Regular'.

15. Optimize the query to reduce refresh time (e.g., remove unused columns early).

Answer:

- At the start of your query steps:
 - **Home** → **Choose Columns** → keep only necessary columns.
- Avoid redundant steps.
- Merge queries where possible.
- Limit the number of transformations.
- Prefer filtering data early in the query.