Lesson 3

Topic: Data Transformation with Power Query (Part 1)

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

The **"Applied Steps" pane** shows a chronological list of all the transformations you've applied to your data in Power Query. Each step (e.g., removing columns, filtering rows, renaming headers) is recorded so that Power Query can **replay** them each time the query runs. You can also modify, delete, or reorder steps from this pane.

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

To **remove duplicate rows**:

- Select the columns you want to check for duplicates.
- Go to the **Home** tab.
- Click **Remove Rows → Remove Duplicates**.
  Power Query will keep the first occurrence and remove all subsequent duplicates based on the selected columns.

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

The **Filter icon** (next to column headers) lets you filter rows based on values in that column. You can:

- Select/deselect specific values.
- Apply number, text, or date filters (e.g., "greater than", "contains", "before").
  Filtering helps reduce or refine the data set according to your criteria.

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

To **rename a column**:

- Right-click the column header **"CustID"** and choose **Rename**.
- Type **"CustomerID"** and press Enter.
  Alternatively, go to the **Transform** tab and click **Rename**.

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

Clicking **"Close & Apply"**:

- **Closes the Power Query Editor**.
- **Applies all changes** (transformation steps) you've made to the data.
- **Loads the transformed data** into the Power BI data model, making it available for reporting, analysis, and visualization.

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

- **Select the `OrderDate` column**.
- Go to the **"Add Column"** tab in the Power Query ribbon.
- Use these options one by one:
  - **Date → Year → Year**
    → This creates a new column with the **year** of the `OrderDate`.
  - **Date → Month → Month**
    → This adds a column with the **month number** (1–12).
  - **Date → Day → Day**
    → This creates a column with the **day** of the month.

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

- In Power Query, select the **`Product`** column.
- Go to the **"Transform"** tab.
- Click **"Replace Values"**.
- In the dialog box:
  - **Value to Find**: `Mouse`
  - **Replace With**: `Computer Mouse`
- Click **OK**.

9. Sort the table by OrderDate (newest first)

Steps:

1. Select the `OrderDate` column.
2. On the **Home** tab → Click **Sort Descending (↓)**.

This will sort your data with the **newest date at the top**.

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

Options:

- **Replace nulls with a default value** (e.g., 0):
  - Select `Price` → **Transform** tab → **Replace Values** → Replace `null` with `0`.
- **Remove rows with null prices**:
  - Select `Price` → **Home** tab → **Remove Rows → Remove Blank Rows**.
- **Fill values** from above or below if context allows:
  - Use **Transform → Fill → Down/Up**.

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

M Code: `= Table.AddColumn(PreviousStepName, "TotalSpent", each [Quantity] * [Price], type number)`
Replace `PreviousStepName` with the actual name of the step before this.
UI alternative:

- Go to **Add Column** tab → **Custom Column**.
- Formula: `[Quantity] * [Price]`
- Name it: `TotalSpent`.

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

Steps:

1. Select the `CustID` column.
2. Go to **Home → Group By**.
3. In the dialog:
    - Group by: `CustID`
    - New column name: `TotalSpent`
    - Operation: **Sum**
    - Column: `TotalSpent` (must be calculated first as in step 11)

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

Steps:

1. Ensure the `OrderDate` column is of **Data Type: Date**.
    - Select `OrderDate` → **Transform → Data Type → Date**.
    - Power Query will parse most formats automatically.

If stored as Text, convert:

```
= Table.TransformColumnTypes(Source, {{"OrderDate", type date}})
```

If Power Query can't interpret, use:

```
= Table.TransformColumns(Source, {{"OrderDate", each Date.FromText(_), type date}})
```

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

Steps:

1. Go to **Add Column → Conditional Column**.
2. Name: `ValueLabel`
3. Condition:
    - If `Price` > 100 → then `"High Value"`
    - Else → `"Regular"`

M-code equivalent:

```
= Table.AddColumn(Source, "ValueLabel", each if [Price] > 100 then "High Value" else "Regular", type text)
```

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

Best Practices:

- **Remove unnecessary columns** early in the query:
    - Select columns you need → Right-click → **Remove Other Columns**
- **Filter rows** early to reduce data volume.
- **Avoid unnecessary steps** (like splitting, renaming multiple times).
- **Disable loading for intermediate queries** if not used in final report:
    - Right-click query → **Enable Load (uncheck)**.
- Use **"Table.Buffer"** if working with referenced tables to cache results.

Example: `= Table.SelectColumns(Source, {"CustID", "OrderDate", "Price", "Quantity`