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

The **"Applied Steps"** pane tracks every transformation you apply to your data.

- It's a **visual history** of your changes (e.g., filtering, renaming, splitting).
- You can **edit, reorder, or delete steps** to change your query behavior.

---

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

- Select the column(s) you want to evaluate for duplicates.
- Click **Home** > **Remove Rows** > **Remove Duplicates**.

---

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

- The **Filter icon** (dropdown next to column headers) lets you:
  - Filter specific values
  - Set number/text/date ranges
  - Keep/remove nulls

---

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

- Right-click the **CustID** column → select **Rename** → type **CustomerID**
  OR use M code:

```
Table.RenameColumns(Source, {{"CustID", "CustomerID"}})
```

---

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

- Power Query **closes**, and all applied steps are:
  - **Saved**
  - **Loaded into Power BI's data model**
  - Used for visualizations and DAX calculations

---

6. Remove all rows where Quantity is less than 2

- Use a filter:
  - Click filter icon on `Quantity` → select **Number Filters > Greater Than** → enter `1`
    M code:

```
Table.SelectRows(Source, each [Quantity] >= 2)
```

---

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

- Select `OrderDate` column → go to **Add Column > Date > Year / Month / Day**
  M code example:

```
Table.AddColumn(Source, "Year", each Date.Year([OrderDate]), Int64.Type)
```

(Repeat for `Month`, `Day`)

---

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

- Right-click `Product` column → **Replace Values**
- Replace **"Mouse"** with **"Computer Mouse"**
  M code:

```
Table.ReplaceValue(Source, "Mouse", "Computer Mouse", Replacer.ReplaceText,
{"Product"})
```

---

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

- Click column header of `OrderDate` → select **Sort Descending**
  M code:

```
Table.Sort(Source, {{"OrderDate", Order.Descending}})
```

---

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

- Replace nulls: Right-click `Price` → **Replace Values** → Replace `null` with `0` or a placeholder
  M code:

```
Table.ReplaceValue(Source, null, 0, Replacer.ReplaceValue, {"Price"})
```

Or filter them out:

```
Table.SelectRows(Source, each [Price] <> null)
```

---

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

```
Table.AddColumn(Source, "TotalSpent", each [Quantity] * [Price], type number)
```

---

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

- Select `CustID` → click **Group By**
- Group by: **CustID**

- Operation: **Sum** on `TotalSpent` column
  M code:

```
Table.Group(Source, {"CustID"}, {{"TotalSpent", each List.Sum([TotalSpent]),
type number}})
```

---

13. Fix inconsistent date formats in `OrderDate`

- Ensure column is of **Date type**:
  - Select `OrderDate` → **Transform** > **Data Type** > **Date**
- Power Query auto-converts text if format is recognized
  If needed, use:

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

---

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

- Go to **Add Column** > **Conditional Column**
- If `Price` > 100 then “High Value” else “Regular”
  M code:

```
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 practice: Remove unnecessary columns **at the start of the query**
  Do this with:

```
Table.SelectColumns(Source, {"OrderDate", "Product", "Quantity", "Price"})
```

This minimizes memory use and speeds up load/refresh time.