1. What is the difference between "Merge" and "Append" in Power Query?

- **Merge** = **Join** two tables **horizontally** by matching key columns (e.g., `CustID`)
- **Append** = **Stack** tables **vertically** (e.g., combine rows from multiple months)

---

2. How do you split a "Full Name" column into "First Name" and "Last Name"?

- Select the `Full Name` column → **Split Column > By Delimiter**
  - Choose **space** as the delimiter
  - Choose "Split at first occurrence"

M code:

```
Table.SplitColumn(Source, "Full Name", Splitter.SplitTextByDelimiter(" ", QuoteStyle.Csv), {"First Name", "Last Name"})
```

---

3. What is "Pivot Columns" used for?

Pivot Columns transforms row values into **column headers**.
For example:
| Product | Month | Sales | → Pivot on "Month"
Becomes → | Product | Jan | Feb | Mar |

---

4. How do you undo a step in Power Query?

- In the **"Applied Steps"** pane, click the **X** next to the step you want to remove.

---

5. What is the purpose of "Reference" vs. "Duplicate" in queries?

- **Duplicate**: Creates a copy with the same steps (independent).
- **Reference**: Creates a new query that **depends** on the original (linked source).

Use **Reference** when you want multiple outputs from the same cleaned dataset.

---

6. Merge Orders.csv and Customers.xlsx on CustID (inner join)

Steps:

- Import both files
- Select **Home > Merge Queries**
- Match on `CustID` from both tables
- Choose **Join Kind: Inner**

M code:

```
Table.NestedJoin(Orders, {"CustID"}, Customers, {"CustID"}, "Customers",
JoinKind.Inner)
```

Then expand the `Customers` column.

---

7. Pivot the `Product` column to show total `Quantity` per product

- Select `Product` column
- Click **Transform > Pivot Column**
- Use `Quantity` as the values column
- Aggregate using **Sum**

---

8. Append two tables with identical columns (e.g., Orders_Jan.csv + Orders_Feb.csv)

- Load both tables
- Use **Home > Append Queries > Append as New**
- Choose the two tables

M code:

```
Table.Combine({Orders_Jan, Orders_Feb})
```

---

9. Use "Fill Down" to replace nulls in the Email column with the previous value

- Select `Email` column → **Transform > Fill > Down**

M code:

```
Table.FillDown(Source, {"Email"})
```

---

10. Extract the domain (e.g., "example.com") from the Email column

Use **Transform > Extract > Text After Delimiter** with `@`

M code:

```
Table.AddColumn(Source, "Domain", each Text.AfterDelimiter([Email], "@"),
type text)
```

---

11. Write M-code to merge queries dynamically based on a parameter (e.g., `JoinType = "Inner"`)

Assume parameter `JoinType` = `"Inner"`:

```
Table.NestedJoin(Orders, {"CustID"}, Customers, {"CustID"}, "Customers",
JoinKind.FromText(JoinType))
```

Note: `JoinKind.FromText()` is not built-in. Instead, use conditional logic:

```
joinType = if JoinType = "Inner" then JoinKind.Inner else JoinKind.LeftOuter,
Table.NestedJoin(Orders, {"CustID"}, Customers, {"CustID"}, "Customers",
joinType)
```

---

12. Unpivot a table with "Jan_Sales", "Feb_Sales" into "Month" and "Sales"

- Select sales columns (e.g., Jan_Sales, Feb_Sales)
- **Transform > Unpivot Columns**

Result:

Month	Sales
Jan_Sales	100
Feb_Sales	200

Use **Replace Values** or **Transform > Extract** to clean month names.

---

13. Handle errors in a custom column (e.g., division by zero) using `try...otherwise`

M code:

```
Table.AddColumn(Source, "SafeDivide", each try [Revenue] / [Units] otherwise
0)
```

---

14. Create a function in Power Query to clean phone numbers (e.g., remove dashes)

1. Create Blank Query → Advanced Editor:

```
(phone as text) =>
let
    cleaned = Text.Select(phone, {"0".."9"})
in
    cleaned
```

Name it `CleanPhone`.

2. Use in column:

```
Table.AddColumn(Source, "CleanedPhone", each CleanPhone([Phone]))
```

---

15. Optimize a query with 10+ steps—identify bottlenecks and simplify

Best Practices:

- **Remove unnecessary columns early**
- **Avoid unnecessary sorts or grouping**
- **Reduce column transformations**
- **Use native SQL queries** for database sources
- **Combine transformations** where possible
- **Disable query previews** while editing (Performance > Fast Data Load)

Example optimization:

```
// Combine select and rename in one step
Table.RenameColumns(Table.SelectColumns(Source, {"ID", "Amount"}), {{"ID",
"CustomerID"}})
```