---

1. What is a primary key in a table?

A **primary key** is a column (or set of columns) that uniquely identifies each row in a table. It **must not contain duplicates or nulls**.
Example: `CustomerID` in the **Customers** table.

---

2. Name the two types of table relationships in Power BI.

- **One-to-many (1:*):** One record in the first table relates to many in the second (most common).
- **Many-to-many (*:*):** Records in both tables can have multiple matches (requires special handling).

---

3. How do you create a relationship between two tables in Power BI?

- Go to **Model view** → Drag and drop the matching field (e.g., `CustomerID`) from one table to another.
- Or use **Manage Relationships** → New → Select the tables and matching columns.

---

4. What is a "star schema"?

A **star schema** is a data model with:

- A central **fact table** (e.g., Sales),
- Surrounded by **dimension tables** (e.g., Products, Customers, Dates),
- Linked via **one-to-many** relationships from dimensions to the fact.
  It looks like a star when visualized.

---

5. Which table is typically the fact table in a sales dataset?

Sales is the **fact table**—it stores transactional data like `OrderID`, `Quantity`, and `ProductID`.

---

6. Link Sales.csv to Customers.csv using CustomerID (one-to-many):

- Ensure `CustomerID` in Customers has **unique values**.
- In **Model view**, create a relationship:
    - Customers[CustomerID] → Sales[CustomerID]
    - Cardinality: **One-to-many**, Cross filter: **Single**.

---

7. Why is ProductID in Sales.csv a foreign key?

Because `ProductID` in Sales refers to products listed in the **Products** table.
It links to the **primary key** in Products, so it's called a **foreign key** in Sales.

---

8. Fix a relationship error where ProductID has mismatched data types:

- Go to **Power Query Editor**,
- Select `ProductID` in both tables,
- Set both to the **same data type** (e.g., Whole Number or Text),
- Apply changes and retry the relationship.

---

9. Explain why a star schema improves performance:

- Simplifies relationships (no many-to-many or loops),
- Reduces ambiguity,
- Improves DAX performance,
- Easier to understand and maintain.

---

10. Add a new column `TotalSales` in Sales (Quantity * Price from Products):

First, create a relationship between `Sales[ProductID]` and `Products[ProductID]`.
Then create a calculated column in **Sales**:

```
TotalSales = Sales[Quantity] * RELATED(Products[Price])
```

---

11. Optimize a model with circular relationships—how would you resolve it?

- **Avoid circular paths** by:
    - Removing one relationship,
    - Using **DAX functions** like `LOOKUPVALUE()` instead of direct joins,
    - Creating a **bridge table** to break the loop.
      Power BI doesn't allow circular relationships, so you must redesign.

---

12. Create a role-playing dimension for OrderDate and ShipDate:

- Duplicate the Date table:
    - `Date_Orders` → relate to Sales[OrderDate]
    - `Date_Shipped` → relate to Sales[ShipDate]
- Use `USERELATIONSHIP()` in DAX to switch between them:

Shipped Sales = CALCULATE([TotalSales], USERELATIONSHIP(Sales[ShipDate], Date_Shipped[Date]))

---

13. Handle a many-to-many relationship between Customers and Products:

- Create a **bridge table** (e.g., `CustomerProduct`) with `CustomerID` and `ProductID`
- Create **two one-to-many relationships**:
    - Customers → CustomerProduct
    - Products → CustomerProduct
      Then, use this bridge table for filtering and analysis.

---

14. Use bidirectional filtering sparingly—when is it appropriate?

- Use only when **filtering needs to flow both ways**, such as:
    - For **dynamic row-level security (RLS)** scenarios
    - When using **bridge tables** to resolve many-to-many relationships
- Avoid it when unnecessary—it **slows performance** and may create ambiguous paths.

---

15. Write DAX to enforce referential integrity if a CustomerID is deleted:

Create a **measure** that checks if a customer exists in Customers:

Customer Exists =

```
IF (
    ISBLANK(RELATED(Customers[CustomerID])),
    "Missing Customer",
    "Valid Customer"
)
```