1. What is a primary key in a table?

A **primary key** in a table is a column (or a combination of columns) whose values **uniquely identify each row** in that table. No two rows can have the same primary key value. A primary key can't contain `NULL` values. A table can only have one primary key, but that key can consist of multiple columns

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

One to many – when One value in a column of the first table relates to multiple rows in another table. Example: One *Customer* can have many *Orders*.

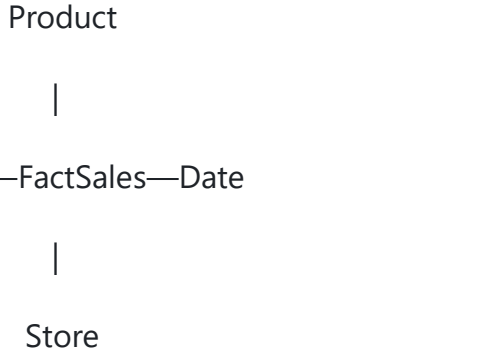
Many-to-Many – when both tables can have multiple matching values for each other.Example: Students and Courses — one student can enroll in many courses, and one course can have many students.

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

After loading data to the power bi, there is an icon on the left side called Model view. There, we can see how the tables are connected to each other. If you want to create relationship, you should take necessary column name and move to other table with the same column name value. For example, connect customer id from customers table with customer id from orders table. In the new window, you can select columns from both tables and choose necessary relationship type. Press ok.

4. What is a "star schema"?

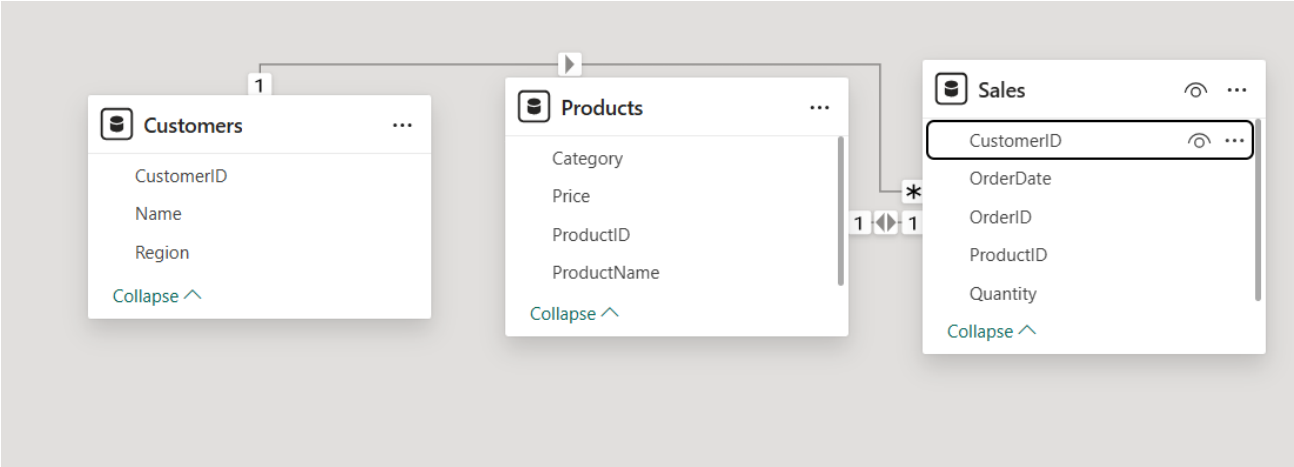
This is shema when there is a one central fact table and it is surrounded with descriptive tables which are only one and are not devided to several parts.



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

In a sales dataset, the Sales (or Transactions) table is typically the fact table. Dimensions like **Products**, **Customers**, and **Dates** describe the facts, but the Sales table stores the actual measurable events.

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



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

Because ProductID in Products table is a primary key and it has relationship with sales table which is descriptive table, where the product id is foreign key.

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

Go to Data view.

Select each table and column (ProductID).

In the ribbon, under Column tools → Data type, set both to the same type (usually Whole Number or Text, depending on your data).

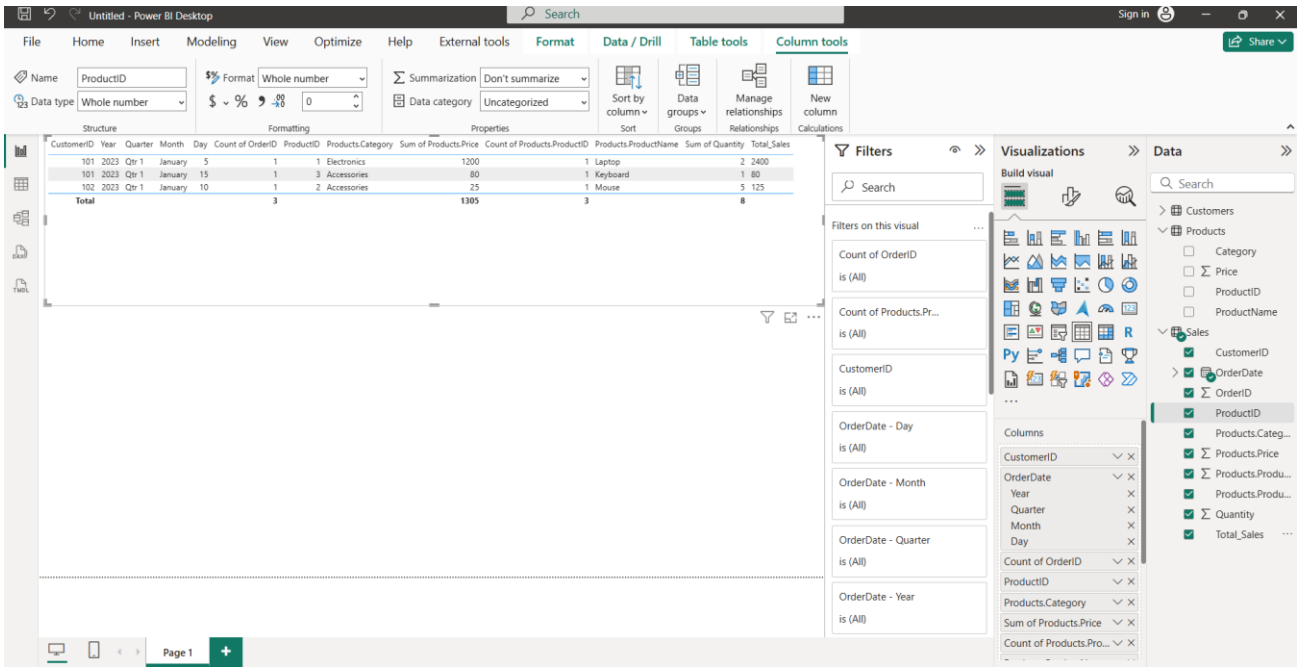
9. Explain why a star schema improves performance.

A star schema improves performance mainly because it is simpler and more efficient for analytical queries compared to more complex models like snowflake schemas.

In a star schema, fact tables link directly to dimension tables. Queries require fewer joins, so they run faster. Fact tables contain measures (e.g., Sales, Quantity) that are easily aggregated without traversing multiple intermediate tables. The model's structure is predictable, so indexes and storage optimizations are more effective.

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

Firstly, with the helping of merge function, sales and products are joined in power query. In add column header, custom column is selected and there this formula is written Total_sales = Quantity * Price from Products. Here is output:



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

In Power BI, **circular relationships** happen when tables are linked in a loop (A → B → C → A).
This causes ambiguity in filter propagation and can slow down or break calculations.

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

In Model View, right-click your Date table → Duplicate (or copy query in Power Query and rename it).

Rename them:

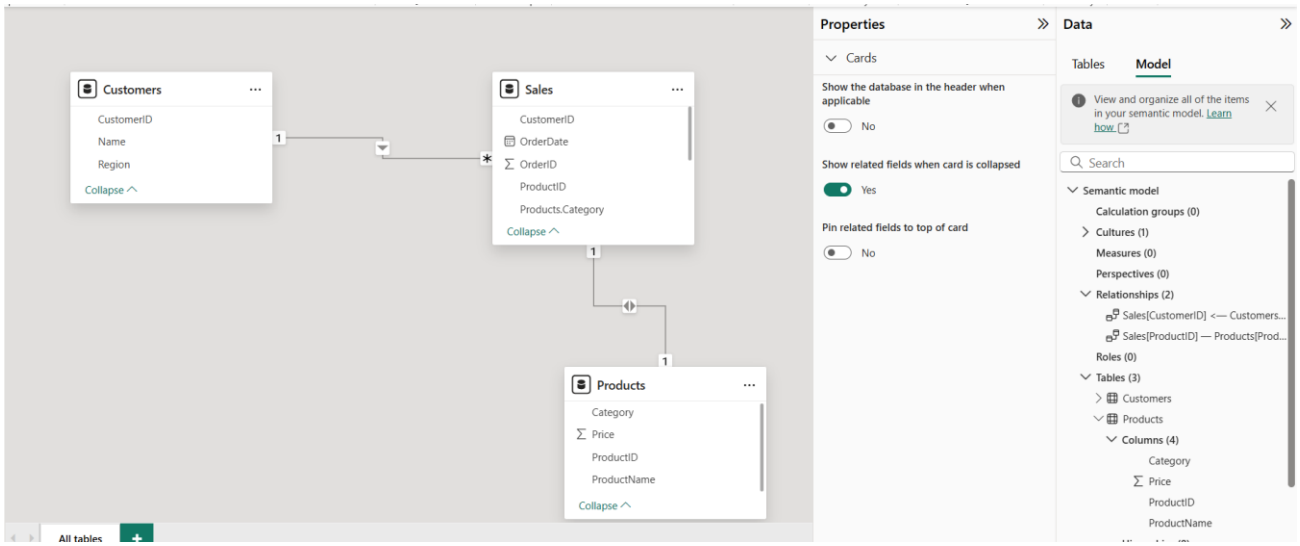
Order Date

Ship Date

. Create separate relationships

- Connect:
    - Order Date[Date] → Sales[OrderDate]
    - Ship Date[Date] → Sales[ShipDate]
- Only one can be *active* at a time.
    - Make the **Order Date** relationship active.
    - Keep the **Ship Date** relationship inactive.

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



Create the relationships

Customers → FactSales (one-to-many on CustomerID)

Products → FactSales (one-to-many on ProductID)

This resolves the many-to-many by making both sides "one-to-many" to the bridge table.

Avoid direct many-to-many

Bad: Customers ↔ Products (many-to-many) — performance and filtering issues

Good: Customers → Orders ← Products (via bridge)

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

In Power BI, **bidirectional filtering** should be used sparingly because it can:

- Increase model complexity
- Cause ambiguous filter paths (leading to unexpected results)
- Reduce performance on large datasets

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

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