1. Primary Key in Each Table:
Sales: OrderID (each order is unique)

Products: ProductID (each product is unique)

Customer: CustomerID (each customer is unique)

2. Types of Relationships:
Customer[CustomerID] → Sales[CustomerID]: One-to-many

Products[ProductID] → Sales[ProductID]: One-to-many

3. Create Relationship (e.g., Customers → Sales):
In Power BI Model view:

Drag CustomerID from Customer table to CustomerID in Sales table.

Repeat for ProductID from Products to Sales.

4. Star Schema:
Your diagram is a classic star schema:

Fact table: Sales

Dimension tables: Products, Customer

Sales is at the center, with direct links to Products and Customer.

5. Fact Table:
Sales is the fact table (contains transaction records).

6. Link Sales.csv to Customers.csv using CustomerID (one-to-many):
Customer table (CustomerID is unique): One

Sales table (CustomerID can repeat): Many

Link: Customers[CustomerID] → Sales[CustomerID]

7. Why is ProductID in Sales.csv a foreign key?
Because each value in Sales[ProductID] points to a unique product in Products[ProductID].

8. Fix a Relationship Error (ProductID mismatched types):
If ProductID is text in one table and number in another:

In Power Query, set both columns to the same type (e.g., Whole Number).

9. Star Schema Performance:
Keeps relationships simple and direct.

Fast aggregation, filtering, and DAX calculations.

Reduces ambiguity.

10. Add a new column TotalSales in Sales (Quantity * Price from Products):
DAX Calculated Column (in Sales table):

dax
Копировать
Редактировать

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

Sample Results:

OrderID	Quantity	ProductID	Price (from Products)	TotalSales
1001	2	1	1200	2400
1002	5	2	25	125
1003	1	3	80	80

11. Optimize Circular Relationships:
With only one path between any two tables, you avoid circular relationships.

If you ever get one (e.g., between Sales, Products, and Customer), remove or redesign a relationship, or use DAX for indirect relationships.

12. Role-Playing Dimension Example:
Suppose Sales had two date columns: OrderDate and ShipDate.

Duplicate your Dates table.

Relate Sales[OrderDate] to Dates1[Date], Sales[ShipDate] to Dates2[Date].

Use USERELATIONSHIP in DAX to activate as needed.

13. Many-to-Many Relationship (Customers & Products):
If a Customer can buy many Products and a Product can be bought by many Customers, Sales is the bridge table (it has both CustomerID and ProductID).

14. Bidirectional Filtering:
Only use when you need filters to flow in both directions, e.g., for complex reporting, but avoid for simple star schema.

15. DAX to Enforce Referential Integrity (e.g., CustomerID deleted):
Highlight "orphaned" sales rows (with a CustomerID not in Customer table):

dax
Копировать
Редактировать

```
IsOrphanedCustomer =
ISBLANK(RELATED(Customer[CustomerID]))
```

Or to count orphaned rows:

dax
Копировать
Редактировать

```
OrphanedSales =
CALCULATE(COUNTROWS(Sales), ISBLANK(RELATED(Customer[CustomerID])))
```