**Power BI Data Modeling – Hands-on Classroom Exercise**

**Learning Objectives:**

By the end of this lab, students will be able to:

* Load and connect multiple tables into Power BI
* Create relationships between tables (1:many, many:many)
* Work with active and inactive relationships
* Understand cross-filter direction
* Build basic charts and use custom DAX measures

**Exercise Overview**

Tables to Use (as per provided data model):

* Sales
* Customer
* Reseller
* Product
* SalesTerritory
* Date
* SalesOrder (used in many-to-many example)

**Part 1: Load the Data**

1. Open Power BI Desktop.
2. Load all 7 tables from Excel/CSV or dataset.
3. Check that field names match the diagram.

**Part 2: Create Relationships (Star Schema)**

Create the following relationships in Model View:

| **From (Fact Table)** | **To (Dimension Table)** | **Column (Fact)** | **Column (Dim)** | **Cardinality** | **Active?** |
| --- | --- | --- | --- | --- | --- |
| Sales | Customer | CustomerKey | CustomerKey | Many-to-One | Active |
| Sales | Product | ProductKey | ProductKey | Many-to-One | Active |
| Sales | Reseller | ResellerKey | ResellerKey | Many-to-One | Active |
| Sales | SalesTerritory | SalesTerritoryKey | SalesTerritoryKey | Many-to-One | Active |
| Sales | Date | OrderDateKey | DateKey | Many-to-One | Active |
| Sales | Date | DueDateKey | DateKey | Many-to-One | Inactive |

**Part 3: Active vs Inactive Relationship Example**

1. Use the inactive relationship between Sales[DueDateKey] and Date[DateKey].
2. Create two DAX measures:

Sales by Order Date :=

CALCULATE(SUM(Sales[Sales Amount]))

Sales by Due Date :=

CALCULATE(

SUM(Sales[Sales Amount]),

USERELATIONSHIP(Sales[DueDateKey], Date[DateKey])

)

1. Create a line chart:
   * X-axis: Date[Full Date]
   * Values: Sales by Order Date, Sales by Due Date

**Part 4: Many-to-Many Relationship Example**

1. Create relationship between:
   * Sales[SalesOrderLineKey] ⇄ SalesOrder[SalesOrderLineKey]
2. Accept the many-to-many warning.
3. Add a table visual:
   * Rows: SalesOrder[Sales Order Line]
   * Values: Sales[Sales Amount]

**Part 5: Cross Filter Direction**

1. In Model View, click on any relationship line.
2. In the properties pane, observe **Cross filter direction**:
   * Default: **Single** (filter flows from dimension to fact)
   * Change to: **Both** to allow bi-directional filtering
3. Test effect by:
   * Creating a slicer using Product[Category]
   * Adding a matrix with SalesTerritory[Region] and Sales[Sales Amount]
   * Compare behavior with **single** vs **both** direction

**Key Note:** Use bidirectional filtering only when necessary to avoid ambiguous filter paths.

**Part 6: Build Visualizations**

1. **Bar Chart**
   * Axis: Product[Category]
   * Value: Sales[Sales Amount]
2. **Map or Bar Chart**
   * Axis: Reseller[Country-Region]
   * Value: Sales[Sales Amount]
3. **Line Chart**
   * Axis: Date[Full Date]
   * Values: Sales by Order Date
4. **Top 5 Customers**
   * Axis: Customer[Customer]
   * Value: Sales[Sales Amount]
   * Apply Top N filter: Top 5 by Sales

**Bonus: DAX Measures**

Create the following measures:

Total Sales := SUM(Sales[Sales Amount])

Total Orders := DISTINCTCOUNT(Sales[SalesOrderLineKey])

Use them in KPI cards or matrix visual.

**Summary**

| **Concept** | **Learned** |
| --- | --- |
| Star schema modeling | ✅ |
| Active vs Inactive | ✅ |
| Many-to-many | ✅ |
| Cross filter direction | ✅ |
| DAX Measures | ✅ |
| Relationships in visuals | ✅ |

**Student Deliverables**

* PBIX file with:
  + Correct model relationships
  + All visuals built
  + DAX measures created

**Instructor Notes:**

* Encourage experimenting with model view
* Discuss the impact of relationship types
* Optional: Include quiz or reflective question at the end