🔹 Week 6: DAX Measures, CALCULATE, Conditional Logic & Slicers [Power BI]

🔸 Objective

The objective of Week 6 was to learn **DAX (Data Analysis Expressions)** for creating **reusable measures**, apply **business logic using CALCULATE and conditional statements**, and build **interactive reports using slicers**. This week focused on **thinking about the question first**, then writing DAX to answer it, instead of relying only on drag-and-drop summaries.

🔸 Tools & Technologies Used

- Power BI Desktop
- DAX (Data Analysis Expressions)
- Data Model (Star Schema with correct relationships from Week 5)
- FEMA & Netflix datasets

🔸 Why DAX Over Drag & Drop

Drag-and-drop gives quick totals, but DAX provides:

- Reusable calculations across visuals
- Context-aware results (works with filters & slicers)
- Business logic inside measures
- Consistent metrics across the report

Guiding question used before creating a measure:

"Which question am I answering with this measure or visual, and why is this calculation required?"

🔸 Measures in Power BI

🔹 Implicit Measures

- Auto-created by Power BI
- Quick but not reusable
- Limited control

🔹 Explicit Measures

- User-created using DAX
- Reusable and flexible
- Recommended for professional reports

🔸 Core DAX Measures (Basic)

```
Total Amount = SUM(FactTable[Amount])

Average Amount = AVERAGE(FactTable[Amount])

Total Records = COUNT(FactTable[RecordID])
```

Purpose: Create consistent totals, averages, and counts used across visuals.

🔸 CALCULATE (Context Modifier) – Syntax & Use

Purpose: Change filter context inside a measure to answer specific questions.

```
Total Major Disasters Amount =
CALCULATE(
    SUM(FactTable[Amount]),
    FactTable[DisasterType] = "Major Disaster"
)
```

What it does:

- Calculates total amount **only for Major Disasters**
- Applies business logic inside the measure (not just via slicers)

🔸 Conditional Logic (IF) – Syntax & Use

Purpose: Classify or segment results based on business rules.

```
Impact Category =
IF(
    [Total Amount] > 1000000,
    "High Impact",
    "Low Impact"
)
```

What it does:

- Dynamically labels results based on a condition
- Useful for KPIs and conditional reporting

🔸 Composite Measure (Measure Using Measures)

```
Amount Per Record =
```

DIVIDE([Total Amount], [Total Records])

Purpose:

- Builds advanced metrics using existing measures
- Improves reusability and consistency

🔸 Slicer (Brief)

What is a Slicer?
A slicer is an interactive filter that allows users to filter visuals by fields such as **State, Year, Category**.

Why use slicers?

- Improves user interaction
- Enables dynamic analysis
- Makes dashboards user-friendly

Slicer vs DAX Filters (Key Idea):

- Slicers control **what the user filters**
- DAX controls **how calculations behave**
- Both are used together for powerful reports

🔸 Key Points (Week 6)

- Defined the business question before writing any DAX
- Understood why DAX is more powerful than drag-and-drop
- Created explicit measures for reuse
- Used CALCULATE to apply business logic
- Used IF for conditional classification
- Built composite measures from base measures
- Used slicers for interactive filtering
- Ensured measures work correctly with the star schema model