Lesson 16

Topic: Advanced Charting Techniques
Prerequisites: Download Card_data.csv file and disable Auto DateTime options by Options in Current File and create new DATE table by using Calendarauto and Format functions.

1.Total Card Limit by Card Brand and Card Type (Stacked Column Chart)

- X-axis: card_brand (e.g., Visa, Mastercard, Amex)
- Y-axis: Sum of card_limit
- Legend: card_type (e.g., Credit, Debit, Prepaid)
- Tooltip: client_id count (i.e., number of clients using this brand/type combo)

To visualize the **total card limit by card brand and card type**, I created a **stacked column chart**. On the X-axis, I used the field card_brand (e.g., Visa, Mastercard, Amex), while on the Y-axis I plotted the **sum of card_limit**. Each column is broken down by card_type (such as Credit, Debit, or Prepaid) to show the contribution of each type within a brand.

To enrich the user interaction, I added a tooltip that displays the **number of clients** using each card brand and type combination. For this, I created a DAX measure:

Client Count = DISTINCTCOUNT('Cards'[client_id])

2.Drill Down into Monthly Trends (Drill Down Feature)

- Create a drill-down chart for analyzing card issues by time:
  - Hierarchy: Year → Month (from account_opened_date)
  - Values: Count of card_number
  - Chart type: Stacked Column Chart
  - Enable drill down/up to move between years and months.

To analyze trends in card issues over time, I built a **drill-down stacked column chart** based on the `account_opened_date`. The chart uses a **date hierarchy** with two levels: **Year → Month**. This allows viewers to explore high-level annual trends and then drill into monthly patterns for deeper analysis.

The **value field** used in the chart is the **count of `card_number`**, which represents how many cards were issued during each time period.

3.Top 10 Clients by Total Card Limit (Bar Chart)

- Axis: client_id

- Value: Total card_limit
- Filter: Top 10 clients
- Sort: Descending by total card limit
- Use a bar chart to clearly show top clients.

  To highlight the most valuable clients in terms of credit exposure, I created a **bar chart** that displays the **Top 10 clients by total card limit**. The X-axis (or category) represents client_id, and the Y-axis shows the **sum of card_limit** associated with each client.

4.Client Drill-through Details

- Create a drill-through page showing all card details (e.g., card_type, card_brand, limit) for a selected client_id when a user clicks from any chart.
- Add filters like card_type or expire_dates (Year).

  To allow for deeper client-specific analysis, I created a **drill-through page** that displays detailed card information for any selected client_id. This drill-through feature enables users to right-click a client in any visual and jump to a dedicated page showing all card-level data for that individual.

5.Heatmap-style Matrix of Expiry Trends

- Rows: card_brand
- Columns: Year of expire_dates
- Values: Count of cards expiring
- Format as a matrix with conditional formatting to highlight peaks (red for more expiries).

  To visualize upcoming card expiries across different brands and years, I created a **heatmap-style matrix**. This visual helps identify expiration peaks using color-coded formatting.

6.Create a dynamic slicer which will give options to select or to enter number and bar chart or column chart need to show only Top N (N should be selected by user in dynamic slicer) Card Brands according to their card limit. (column chart)

- X axis: Card Brand
- Y axis: Sum of Card Limit

  To provide flexible control over how many top card brands to display, I created a **dynamic Top N slicer** combined with a **column chart** that updates based on the user's input.

7.Download sales.csv file and disable Auto DateTime options by Options in Current File and create new DATE table by using Calendarauto and Format functions.

- Calculate Average number of days between Sales date and previous sales date for each customer
  (ex:

last sales date 05.05.2025,
prev sales date 05.01.2025,
sales before prev sales date 04.21.2025
Answer: 7 (4 + 10) / 2)

I created a new **date table** using the CALENDARAUTO() function combined with FORMAT() for readable columns:

```
Date =

ADDCOLUMNS(

    CALENDARAUTO(),

    "Year", YEAR([Date]),

    "Month", FORMAT([Date], "MMMM"),

    "MonthNum", MONTH([Date]),

    "Day", DAY([Date]),

    "Quarter", "Q" & FORMAT([Date], "Q")

)
```