Part 1: Card Data (Card_data.csv)

1. DATE Table

```
DateTable =
ADDCOLUMNS (
    CALENDARAUTO(),
    "Year", YEAR([Date]),
    "Month", FORMAT([Date], "MMM"),
    "YearMonth", FORMAT([Date], "YYYY-MM")
)
```

👉 Disable *Auto Date/Time* in **File → Options → Current File → Data Load**.

---

2. Total Card Limit by Card Brand & Card Type

- Chart: **Stacked Column**
- X-axis → `card_brand`
- Y-axis → `SUM(card_limit)`
- Legend → `card_type`
- Tooltip → `DISTINCTCOUNT(client_id)`

Measure for tooltip:

```
Client Count = DISTINCTCOUNT(Card_data[client_id])
```

---

3. Drill-down into Monthly Trends

1. Link `account_opened_date` → `DateTable[Date]`.
2. Chart hierarchy: **Year → Month**.
3. Value = count of cards:

```
Cards Issued = COUNTROWS(Card_data)
```

4. Chart type: Stacked Column → enable drill-down.

---

4. Top 10 Clients by Card Limit

Measure:

```
Total Card Limit = SUM(Card_data[card_limit])
```

Visual filter:

- Add **Top N filter** on `client_id` → Top 10 by `Total Card Limit`.
- Chart: Bar, sort descending.

---

5. Client Drill-through Page

1. Create new page → add fields: `client_id`, `card_type`, `card_brand`, `card_limit`, `expire_dates`.
2. Set page as **Drill-through** on `client_id`.
3. Add slicers for `card_type`, `Year(expire_dates)`.

---

6. Expiry Trends Heatmap

- Matrix visual:
  - Rows → `card_brand`
  - Columns → `Year(expire_dates)`
  - Values → `COUNT(card_number)`
- Add **conditional formatting** (color scale: green → yellow → red).

---

7. Dynamic Top N Card Brands by Limit

Create parameter table:

```
TopN = GENERATESERIES(1, 20, 1)
```

Measure:

```
Top N Card Brands =
VAR N = SELECTEDVALUE(TopN[Value], 5)
RETURN
IF (
    RANKX(ALL(Card_data[card_brand]), [Total Card Limit]) <= N,
    [Total Card Limit]
)
```

Chart:

- Axis → `card_brand`
- Y → `Top N Card Brands`

Add slicer from **TopN[Value]**.

📊 Part 2: Sales Data (Sales.csv)

1. DATE Table

```
SalesDateTable =
ADDCOLUMNS (
    CALENDARAUTO(),
```

```
    "Year", YEAR([Date]),
    "Month", FORMAT([Date], "MMM"),
    "YearMonth", FORMAT([Date], "YYYY-MM")
)
```

---

2. Average Days Between Sales per Customer

We need lagged dates.

```
Prev Sales Date =
CALCULATE (
    MAX(Sales[sales_date]),
    FILTER (
        Sales,
        Sales[customer_id] = EARLIER(Sales[customer_id]) &&
        Sales[sales_date] < EARLIER(Sales[sales_date])
    )
)
```

Days between:

```
Days Between =
DATEDIFF([Prev Sales Date], Sales[sales_date], DAY)
```

Average per customer:

```
Avg Days Between Sales =
AVERAGEX (
    VALUES(Sales[customer_id]),
    AVERAGE(Sales[Days Between])
)
```

☑ Example you gave (05.05 → 05.01 → 04.21):
Differences = 4 + 10 = 14
Average = 14 / 2 = **7** ☑