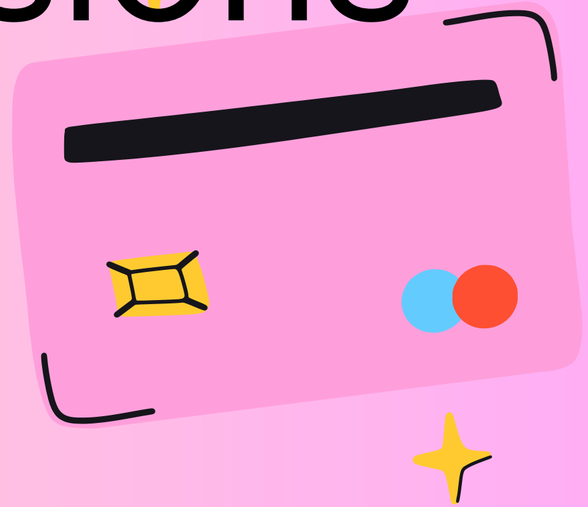
Financial Analysis Using DAX expressions

Credit Card Analysis



By: Gadifele Mabale

Role: Financial Data Analyst

Task: Analyzing credit card usage and financial metrics for a banking institution using Power BI and DAX functions.

Objective: Calculate financial metrics including:

- Running totals
- Moving averages
- Growth rates
- KPIs for customer behavior assessment
- Credit utilization analysis
- Delinquency risk evaluation
-

Next Step: Write DAX formulas for the following financial metrics and KPIs.

1. running total of credit card transactions

```
1 Running_Total = CALCULATE(
2     SUM(credit_card[Total_Trans_Amt]),
3     FILTER(ALL('Calendar'[Date]),
4     'Calendar'[Date] <= MAX('Calendar'[Date])
5     )
6 )
```

Month	Running_Total
January	$4,322,186
February	$7,861,761
March	$11,250,588
April	$15,425,316
May	$18,852,229
June	$22,385,889
July	$26,932,847
August	$30,382,715
September	$33,835,589
October	$37,886,498
November	$41,291,918
December	$44,522,013
Total	**$44,522,013**

2. 4-week moving average of the creditLimit for each client.

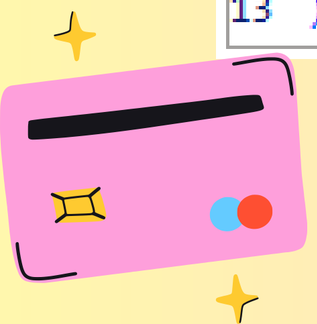
```
4_Week_Moving_Average =
AVERAGEX(
    DATESINPERIOD('Calendar'[Date],MAX('Calendar'[Date]), -28, DAY),
    CALCULATE(AVERAGE(credit_card[Credit_Limit])
    )
```

3. mom% growth on transaction amount.

```
MoM_Growth =
VAR current_month = CALCULATE(SUM(credit_card[Total_Trans_Amt])
)
VAR previous_month = CALCULATE(SUM(credit_card[Total_Trans_Amt]),
DATEADD('Calendar'[Date], -1,MONTH)
)
RETURN
DIVIDE(current_month - previous_month,
previous_month,
0
)
```

wow% growth on transaction amount.

```
1  WoW =
2  VAR current_week =
3  CALCULATE(SUM(credit_card[Total_Trans_Amt])
4  )
5  VAR previous_week =
6  CALCULATE(SUM(credit_card[Total_Trans_Amt]),
7  DATEADD('Calendar'[Date], -7,DAY)
8  )
9  RETURN
10 DIVIDE(current_week - previous_week,
11 previous_week,
12 0
13 )
```



4. Customer Acquisition Cost (CAC) as a Ratio of Transaction Amount.

```
1  Customer_Acquisition_Cost =
2  DIVIDE(SUM(credit_card[Customer_Acq_Cost]),
3  SUM(credit_card[Total_Trans_Amt]),0
4  )
```

5. Yearly average of avg_utilization_ratio for all clients

```
Yearly_Average_Ratio = AVERAGE(credit_card[Avg_Utilization_Ratio])
```

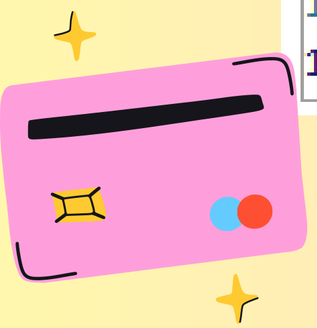
6. Percentage of Interest_Earned compared to Total_Revolving_Bal for each client.

```
1 %_InterestEarned =
2 DIVIDE(SUM(credit_card[Interest_Earned]),
3 SUM(credit_card[Total_Revolving_Bal]),
4 0
5 )
```

7. Top 5 Clients by Total Transaction Amount.

```
Top_5_Clients =
TOPN(
    5,
    ADDCOLUMNS(
        SUMMARIZE(
            Credit_Card,
            Credit_Card[Client_Num]
        ),
        "Total_Transactions", CALCULATE(SUM(Credit_Card[Total_Trans_Amt]))
    ),
    [Total_Transactions],
    DESC
)

```



8. Clients whose Avg_Utilization_Ratio exceeds 80%

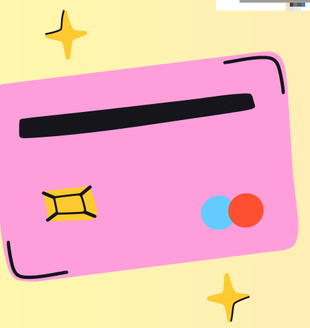
```
High_Utilization =
IF(
    AVERAGE(credit_card[Avg_Utilization_Ratio]) > 0.8,
    "High Utilization",
    "Low Utilization"
)
```

9. Customer Churn Indicator: Create a KPI that flags clients who have not made any transactions (Total_Trans_Amt = 0) in the last 6 months.

```
trans_in_last_6_months =
VAR Last6Months =
    CALCULATE(
        SUM(credit_card[Total_Trans_Amt]),
        DATESINPERIOD(Calendar[Date], MAX(Calendar[Date]), -6, MONTH)
    )
RETURN
    IF(ISBLANK(Last6Months) || Last6Months = 0, TRUE, FALSE)
```

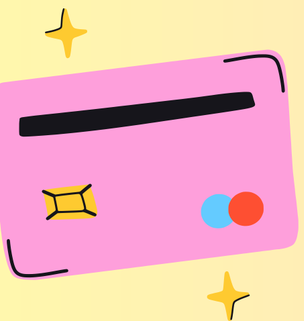
10. Delinquency Rate: Calculate the percentage of clients with Delinquent_Acc > 0.

```
Delinquency_Rate: =
DIVIDE(
    CALCULATE(
        COUNTROWS(Credit_Card),
        Credit_Card[Delinquent_Acc] > 0
    ),
    COUNTROWS(Credit_Card),
    0
)
```



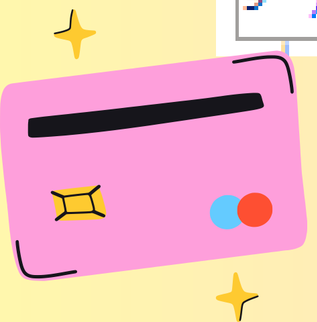
11. Credit Risk Score: Create a score for each client based on their Avg_Utilization_Ratio, Delinquent_Acc, and Total_Revolving_Bal.

```
1  Credit_Risk_Score =
2  DIVIDE(SUM(Credit_Card[Total_Revolving_Bal]),
3   SUM(Credit_Card[Credit_Limit])) * 0.4 +
4  AVERAGE(Credit_Card[Avg_Utilization_Ratio]) * 0.3 +
5  DIVIDE(SUM(Credit_Card[Delinquent_Acc]),
6  COUNTROWS(Credit_Card)) * 0.3
```



12. Income vs Credit Limit Correlation: Show the correlation between Income and Credit_Limit for all clients.

```
Income_Vs_Credit_Limit_Table =
SUMMARIZE(
    Credit_Card,
    Credit_Card[Client_Num],
    "Income",
     CALCULATE(AVERAGE(cust_add[Income])),
    "Credit_Limit",
    CALCULATE(AVERAGE(Credit_Card[Credit_Limit]))
)
```



13. Average Customer Satisfaction Score by Credit Card Category: Calculate the average Cust_Satisfaction_Score by Card_Category

```
1  Average_Customer_Satisfaction =
2  AVERAGE(customer[Cust_Satisfaction_Score])
```

14 Loan Approval vs Credit Limit: Analyze how Credit_Limit affects Personal_loan approval by calculating the average credit limit for clients with and without loans.

```
Avg_CreditLimit_With_Loan =
CALCULATE(
    AVERAGE(Credit_Card[Credit_Limit]),
    customer[Personal_loan] = "Yes"
)
```

```
Avg_CreditLimit_Without_Loan =
CALCULATE(
    AVERAGE(Credit_Card[Credit_Limit]),
    customer[Personal_loan]= "No"
)
```

15. High Risk Clients Flag: Create a flag for clients whose Total_Revolving_Bal exceeds 90% of their Credit_Limit and who have a high Avg_Utilization_Ratio.

```
exceeds_90%_of_creditlimit =
VAR ninety_percent_limit =
    credit_card[Credit_Limit] * 0.9

RETURN
    IF(
        credit_card[Total_Revolving_Bal] > ninety_percent_limit,
        TRUE,
        FALSE
    )
```