



# FINANCIAL DATA ANALYSIS

**PRESENTED BY: PRIYANSH BHANGALIA**

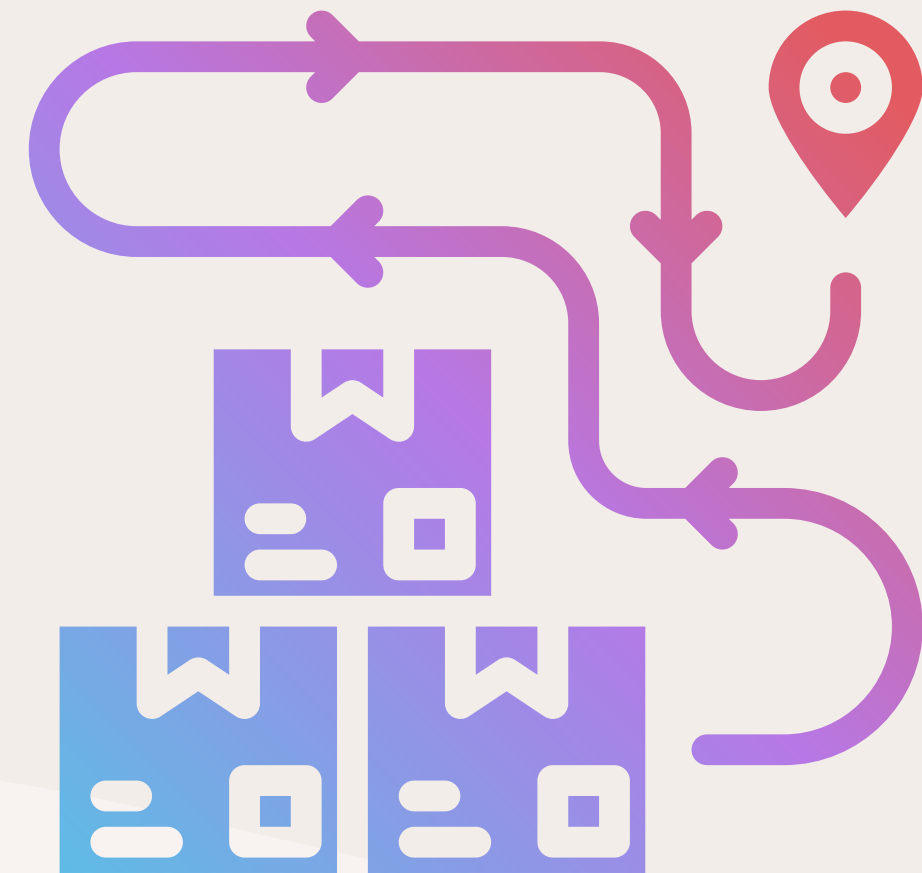


# Table of Content

Analysing financial data  
using dax formulas in Power  
BI.

DAX formulas 4-20

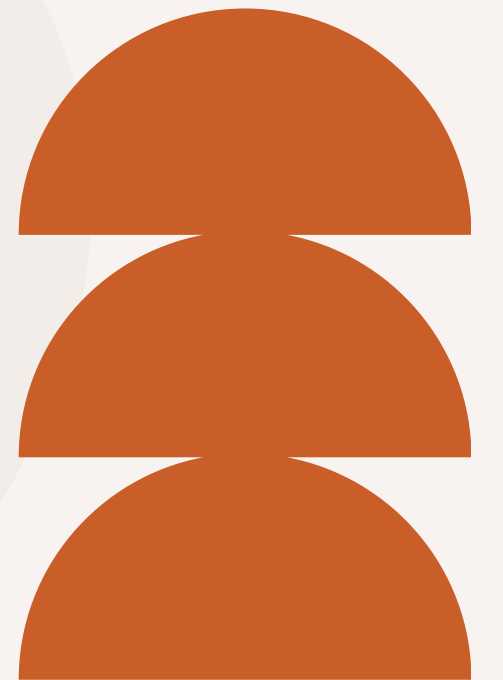
Visual's 22-24





# DAX formulas

Next slide

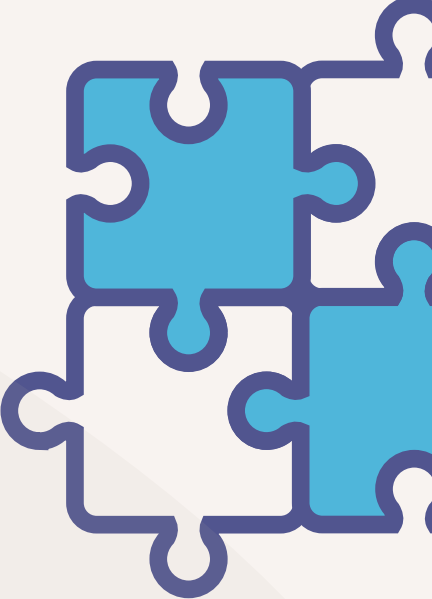





## 1. Running Total of Credit Card Transactions

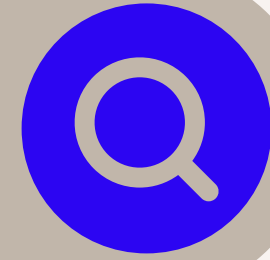


```
1 RunningTotal =  
2 CALCULATE(  
3     SUM('credit_card'[Total_Trans_Amt]),  
4     FILTER(  
5         ALL('credit_card'),  
6         'credit_card'[Week_Start_Date] <= MAX('credit_card'  
7         [Week_Start_Date])  
8     )  
9 )
```





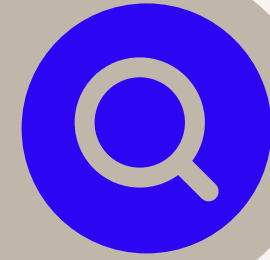
## 2. Calculate the 4-week moving average of the creditLimit for each client.



```
1 Moving Average =  
2 VAR weeks = DATESINPERIOD('calender'[Date], MAX('calender'[Date]), -28, DAY)  
3 VAR sales =  
4 |     CALCULATE(  
5 |         SUM('credit_card'[Credit_Limit]),  
6 |         weeks)  
7 VAR dis_week =  
8 |     CALCULATE(  
9 |         DISTINCTCOUNT('calender'[weeknum]),  
10 |        weeks)  
11 RETURN DIVIDE(sales, dis_week)
```



### 3. Calculate the mom% growth and wow% growth on transaction amount.



```
1 MOM%growth =
2 VAR prev_month =
3     CALCULATE(
4         SUM('credit_card'[Total_Trans_Amt]),
5         DATEADD('calender'[Date], -1, MONTH))
6 RETURN
7     DIVIDE(
8         SUM('credit_card'[Total_Trans_Amt]) - prev_month,
9         prev_month, 0)
```

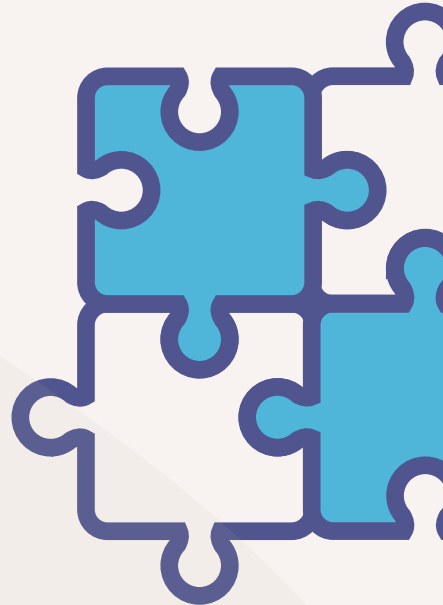
```
1 WOW%growth =
2 VAR prev_week =
3     CALCULATE(
4         SUM('credit_card'[Total_Trans_Amt]),
5         DATEADD('calender'[Date], -7, DAY))
6 RETURN
7     DIVIDE(
8         SUM('credit_card'[Total_Trans_Amt]) - prev_week,
9         prev_week, 0)
```



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



```
1 cac_ta =  
2     DIVIDE(  
3         SUM('credit_card'[Customer_Acq_Cost]),  
4         SUM('credit_card'[Total_Trans_Amt])  
5     )
```

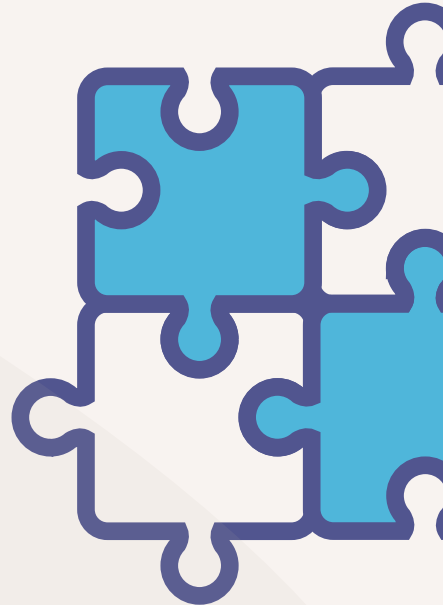




5. Calculate the yearly average of avg\_utilization\_ratio for all clients.



```
1 avg_utilization_ratio =  
2 |     AVERAGE('credit_card'[Avg_Utilization_Ratio])/  
3 |     DISTINCTCOUNT('credit_card'[current_year])
```



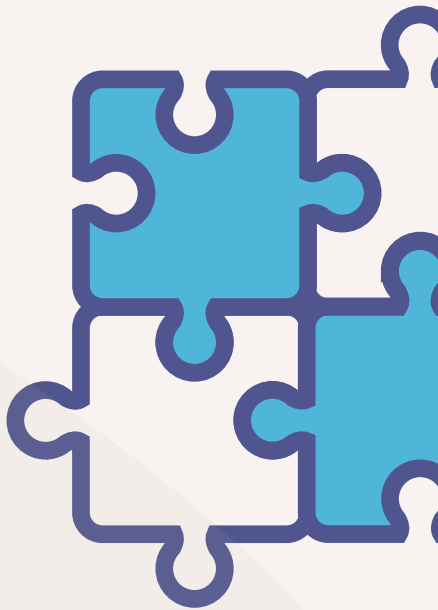




**6. Calculate the percentage of Interest\_Earned compared to Total\_Revolving\_Bal for each client.**



```
1 intreset_by_rev_bal =  
2     DIVIDE(  
3         SUM('credit_card'[Interest_Earned]),  
4         SUM('credit_card'[Total_Revolving_Bal]),  
5         0  
6     )
```

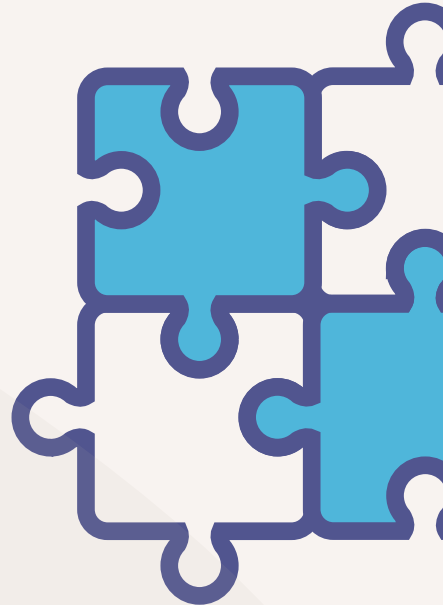




## 7. Calculate Top 5 Clients by Total Transaction Amount.

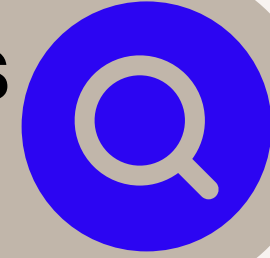


```
1 top_5_clients_by_traction_amount =  
2  
3 TOPN(5, SUMMARIZE('credit_card', 'credit_card'[Client_Num],  
4  
5 "total amount",  
6  
7 SUM('credit_card'[Total_Trans_Amt])),  
8  
9 [total amount], DESC)
```

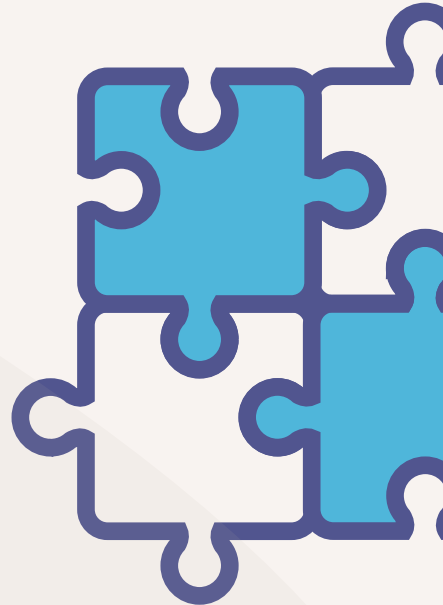




8. Identify clients whose Avg\_Utilization\_Ratio exceeds 80%.



```
1 avg_uti_exceeds_80% =  
2  
3 IF('credit_card'[Avg_Utilization_Ratio] >0.8,TRUE,FALSE)
```

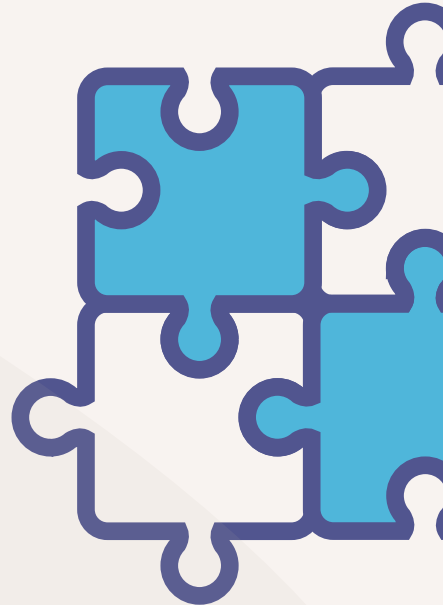




9. Customer Churn Indicator: Create a KPI that flags clients who have not made any transactions (Total\_Trans\_Amt = 0) in the last 6 months.



```
1 no_trans_in_last_6_months =  
2 VAR months_6 =  
3     CALCULATE(  
4         SUM('credit_card'[Total_Trans_Amt]),  
5         DATESINPERIOD('calender'[Date], MAX('calender'[Date]), -6, MONTH)  
6     )  
7 RETURN  
8     IF(  
9         ISBLANK(months_6),  
10        TRUE,  
11        FALSE)
```

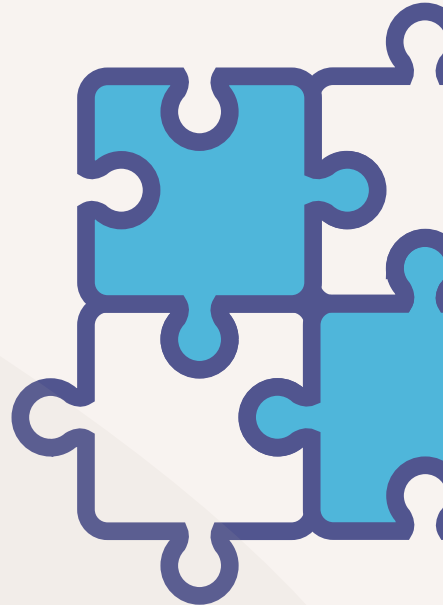




## 10.Delinquency Rate: Calculate the percentage of clients with Delinquent\_Acc > 0.



```
1 delinquency_rate =
2 VAR delinquent_acc =
3     CALCULATE(
4         COUNTROWS('credit_card'),
5         'credit_card'[Delinquent_Acc] > 0)
6 VAR total_accounts =
7     COUNTROWS('credit_card')
8 RETURN
9     DIVIDE(
10        delinquent_acc,
11        total_accounts,
12        0)
```

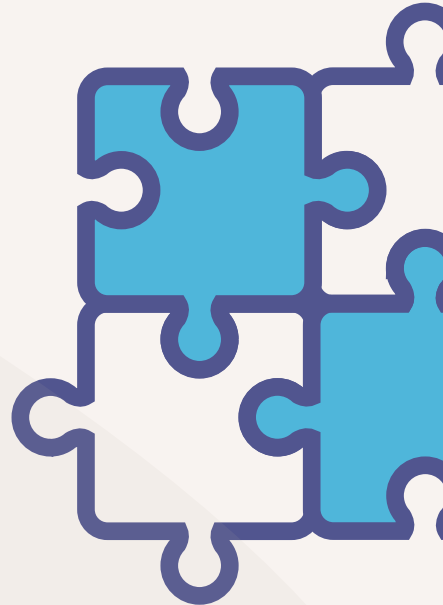




## 11.Credit Risk Score: Create a score for each client based on their Avg\_Utilization\_Ratio, Delinquent\_Acc, and Total\_Revolving\_Bal.



```
1 normalized_revolving_balance =  
2 VAR min_value =  
3 | MIN('credit_card'[Total_Revolving_Bal])  
4 VAR max_value =  
5 | MAX('credit_card'[Total_Revolving_Bal])  
6 RETURN  
7 | DIVIDE(  
8 |     'credit_card'[Total_Revolving_Bal] - min_value,  
9 |     max_value - min_value,  
10 |     0)
```



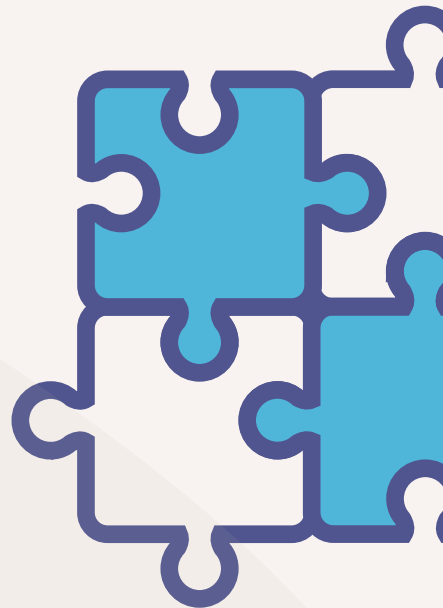


## 11. Credit Risk Score: Create a score for each client based on their Avg\_Utilization\_Ratio, Delinquent\_Acc, and Total\_Revolving\_Bal.



```
1 credit_risk_score =  
2  
3     0.5*'credit_card'[Avg_Utilization_Ratio]+  
4     0.3*'credit_card'[Delinquent_Acc]+  
5     0.2*'credit_card'[normalized_revolving_balance]
```

```
1 credit_risk_value =  
2     SWITCH(  
3         TRUE(),  
4         'credit_card'[credit_risk_score] <= 0.2, "Low Risk",  
5         'credit_card'[credit_risk_score] > 0.2 && 'credit_card'[credit_risk_score] <= 0.5,  
6         "Medium Risk",  
7         'credit_card'[credit_risk_score] > 0.5, "High Risk"  
8     )
```



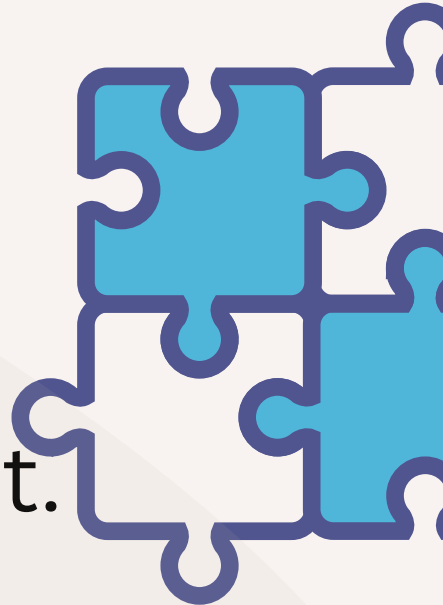


## 12.Income vs Credit Limit Correlation: Show the correlation between Income and Credit\_Limit for all clients.



In this question we can use quick measure available in home tab under calculations

- 1.Select quick measure then in quick measure select correlation coefficient.
- 2.Then in category select client\_num.
- 3.In measure X select income from customer data table.
- 4.In measure Y select Credit\_limit from credit\_card table then press add your formula will be created.





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



Calculations

Suggestions with Copilot

Correlation coefficient

Calculate the correlation coefficient between two values over a category. Originally suggested by Daniil Maslyuk in the quick measures gallery. [Learn more](#)

Category ⓘ

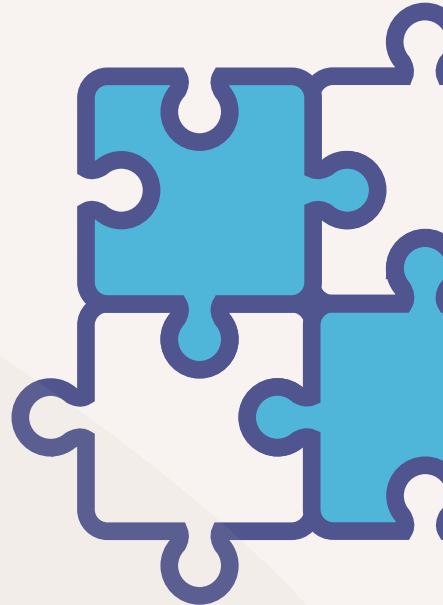
Client\_Num

Measure X ⓘ

Income

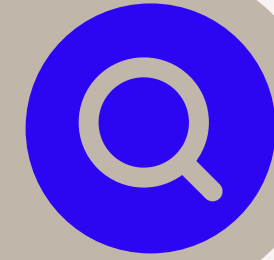
Measure Y ⓘ

Credit\_Limit



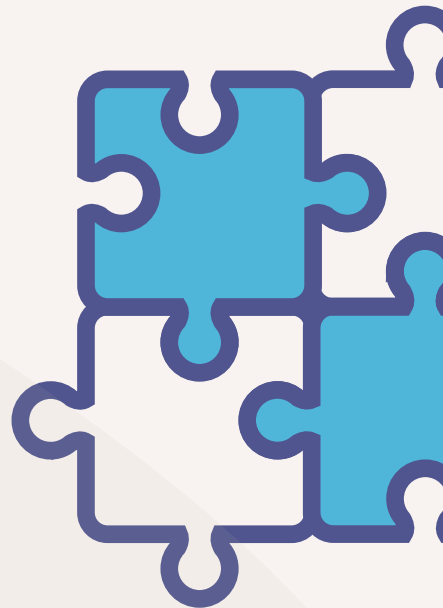


### 13.Average Customer Satisfaction Score by Credit Card Category: Calculate the average Cust\_Satisfaction\_Score by Card\_Category.



```
1 avg_score_by_card_category =  
2     SUMMARIZE(  
3         'credit_card',  
4         'credit_card'[Card_Category],  
5         "avg_score",  
6         ROUND(AVERAGE('customer data'[Cust_Satisfaction_Score]), 2)  
7     )
```

Card_Category ▾	avg_score ▾
Blue	3.2
Silver	3.22
Gold	3.05
Platinum	2.72



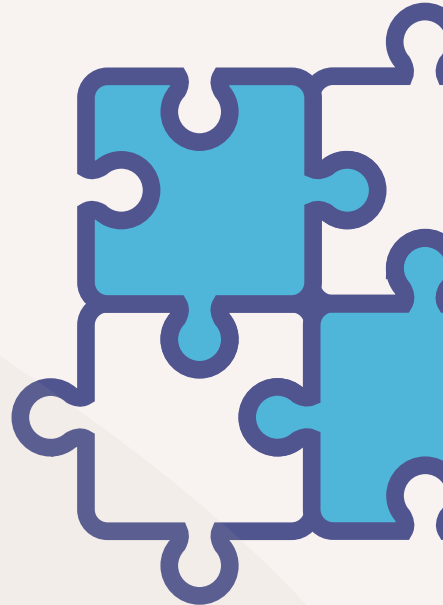


**14.Loan Approval vs Credit Limit: Analyze how Credit\_Limit affects Personal\_loan approval by calculating the average credit limit for clients with and without loans.**



```
1 loan_yes =  
2 CALCULATE(  
3     AVERAGE(  
4         'credit_card'[Credit_Limit]),  
5         'customer data'[Personal_loan] = "yes")
```

```
1 loan_no =  
2 CALCULATE(  
3     AVERAGE(  
4         'credit_card'[Credit_Limit]),  
5         'customer data'[Personal_loan] = "no")
```

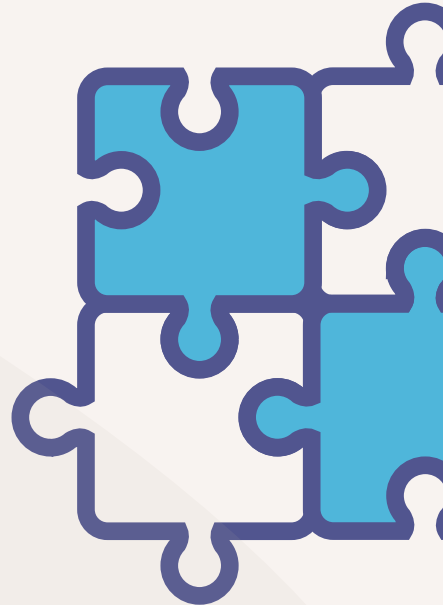


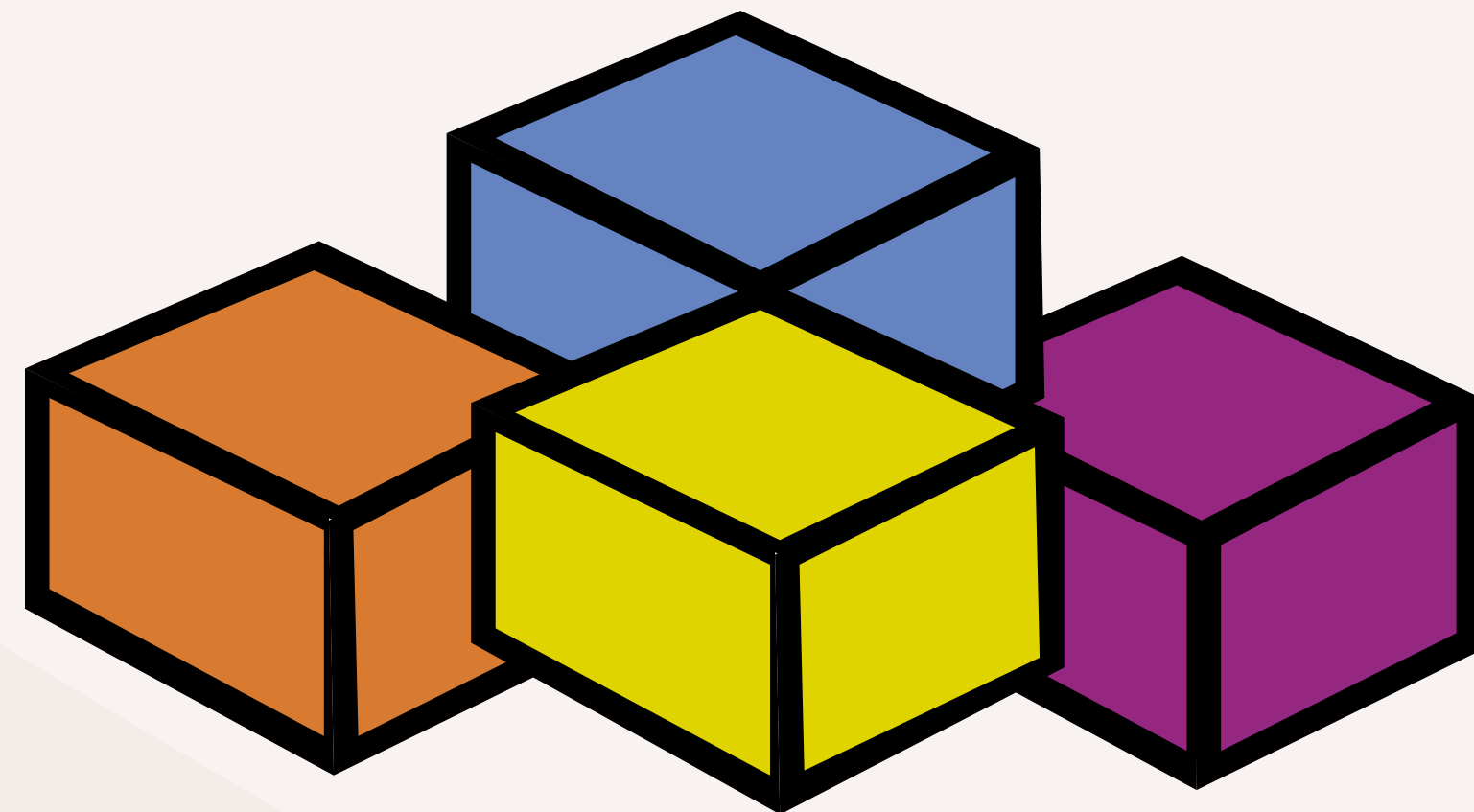


**15.High Risk Clients Flag: Create a flag for clients whose Total\_Revolving\_Bal exceeds 90% of their Credit\_Limit and who have a high Avg\_Utilization\_Ratio.**



```
1 exceeds_90%_of_creditLimit =  
2 VAR c1_90 =  
3     'credit_card'[Credit_Limit] * 0.9  
4  
5 RETURN  
6     IF('credit_card'[Total_Revolving_Bal] > c1_90, TRUE, FALSE)
```





# Visual's

Next slide



Week_Start_Date	Sum of Total_Trans_Amt	RunningTotal
01 January 2023	₹ 8,35,767	₹ 8,35,767
08 January 2023	₹ 8,44,739	₹ 16,80,506
15 January 2023	₹ 9,23,367	₹ 26,03,873
22 January 2023	₹ 8,69,235	₹ 34,73,108
29 January 2023	₹ 8,49,078	₹ 43,22,186
05 February 2023	₹ 8,98,867	₹ 52,21,053
12 February 2023	₹ 8,90,756	₹ 61,11,809
19 February 2023	₹ 8,68,091	₹ 69,79,900
26 February 2023	₹ 8,81,861	₹ 78,61,761
05 March 2023	₹ 7,93,080	₹ 86,54,841
12 March 2023	₹ 9,15,725	₹ 95,70,566
19 March 2023	₹ 8,90,081	₹ 1,04,60,647
26 March 2023	₹ 7,89,941	₹ 1,12,50,588
02 April 2023	₹ 8,09,413	₹ 1,20,60,001
<b>Total</b>	<b>₹ 4,55,33,021</b>	<b>₹ 4,55,33,021</b>

Week_Num	Sum of Total_Trans_Amt	RunningTotal
Week-53	₹ 10,11,008	₹ 4,55,33,021
Week-1	₹ 8,35,767	₹ 8,35,767
Week-2	₹ 8,44,739	₹ 16,80,506
Week-3	₹ 9,23,367	₹ 26,03,873
Week-4	₹ 8,69,235	₹ 34,73,108
Week-5	₹ 8,49,078	₹ 43,22,186
Week-6	₹ 8,98,867	₹ 52,21,053
Week-7	₹ 8,90,756	₹ 61,11,809
Week-8	₹ 8,68,091	₹ 69,79,900
Week-9	₹ 8,81,861	₹ 78,61,761
Week-10	₹ 7,93,080	₹ 86,54,841
Week-11	₹ 9,15,725	₹ 95,70,566
Week-12	₹ 8,90,081	₹ 1,04,60,647
Week-13	₹ 7,89,941	₹ 1,12,50,588
Week-14	₹ 8,09,413	₹ 1,20,60,001
<b>Total</b>	<b>₹ 4,55,33,021</b>	<b>₹ 4,55,33,021</b>

weeknum	Sum of Credit_Limit	Moving Average
1	₹ 17,04,635.7	₹ 13,03,109.62
2	₹ 16,00,959.7	₹ 16,52,797.7
3	₹ 15,26,415.1	₹ 16,10,670.166666667
4	₹ 16,47,543.2	₹ 16,19,888.425
5	₹ 17,98,645	₹ 16,43,390.75
6	₹ 17,10,629.7	₹ 16,70,808.25
7	₹ 19,76,267	₹ 17,83,271.225
8	₹ 17,33,164.8	₹ 18,04,676.625
9	₹ 15,07,517.7	₹ 17,31,894.8
10	₹ 14,41,620.2	₹ 16,64,642.425
11	₹ 16,48,313.7	₹ 15,82,654.1
12	₹ 19,37,345.6	₹ 16,33,699.3
13	₹ 14,77,819.9	₹ 16,26,274.85
14	₹ 16,16,093.4	₹ 16,69,893.15
Total	₹ 8,89,56,376	₹ 13,03,109.62



month_name	Sum of Total_Trans_Amt	MOM%growth
August	₹ 34,49,868	-24.13%
February	₹ 35,39,575	-18.11%
May	₹ 34,26,913	-17.91%
November	₹ 34,05,420	-15.93%
March	₹ 33,88,827	-4.26%
September	₹ 34,52,874	0.09%
January	₹ 43,22,186	1.91%
June	₹ 35,33,660	3.11%
October	₹ 40,50,909	17.32%
April	₹ 41,74,728	23.19%
December	₹ 42,41,103	24.54%
July	₹ 45,46,958	28.68%

weeknum	Sum of Total_Trans_Amt	WOW%growth
30	₹ 8,28,881	-0.87%
1	₹ 8,35,767	0.00%
40	₹ 7,98,938	0.20%
38	₹ 8,96,128	0.30%
20	₹ 8,37,155	0.33%
34	₹ 8,68,903	0.58%
25	₹ 8,99,730	0.92%
2	₹ 8,44,739	1.07%
9	₹ 8,81,861	1.59%
16	₹ 8,67,373	1.93%
14	₹ 8,09,413	2.46%
37	₹ 8,93,463	3.18%
24	₹ 8,91,529	3.73%
45	₹ 8,83,695	4.14%



Client_Num	intreset_by_rev_bal
708082083	264.49%
708083283	2.76%
708084558	11.44%
708085458	0.00%
708086958	134.16%
708095133	15.01%
708098133	11.27%
708099183	21.88%
708100533	43.42%
708103608	150.64%
708104658	82.25%
708100333	0.00%
<b>Total</b>	<b>66.63%</b>

6.06%

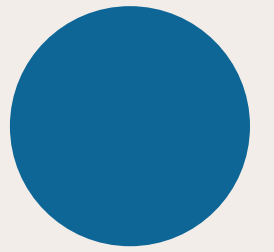
delinquency\_rate

0.13

Income and Credit\_Limit correlation  
for Client\_Num

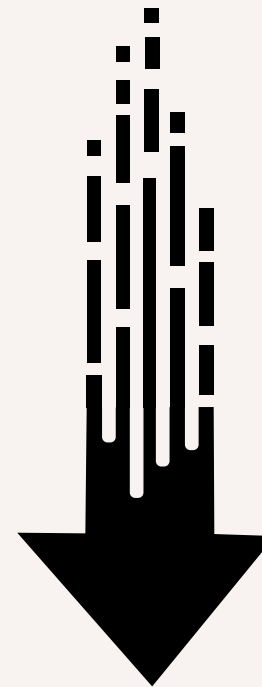






# THANK YOU!

**DO FOLLOW FOR MORE CONTENT AND DON'T FORGET TO LIKE AND  
SHARE**



**PRIYANSH BHANGALIA**