Data Validation using SQL

**BANK LOAN REPORT | SUMMARY**

**KPI - TOTAL LOAN APPLICATIONS**

1. --Total No.of Loan Applications

Select COUNT(ID) as 'Total\_Loan\_Applications' from bank\_loan\_data;



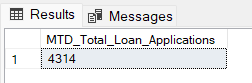
1. --MTD Total Loan Applications

Select COUNT(ID) as 'MTD\_Total\_Loan\_Applications' from bank\_loan\_data

where MONTH(issue\_date) = 12 and YEAR(issue\_date) = 2021;

-- Latest month available in the dataset is December and the Month number for December is "12"

-- MTD(Month to Date) means the period starting from the beginning of the current month up until the present date, not including the current date.



1. –- Previous MTD Total Loan Applications

Select COUNT(ID) as 'PMTD\_Total\_Loan\_Applications' from bank\_loan\_data

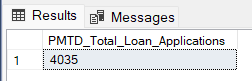
where MONTH(issue\_date) = 11 and YEAR(issue\_date) = 2021;

-- Last month for december is november and the Month number for November is "11"

--MoM(Month on Month) is an expression used to compare data from one month to the previous month. Here we are calculating only previous month data and not the comparison

-- Formula for MoM is (MTD-PMTD)/PMTD = (4314-4035)/4035 =

0.069 \* 100 = 6.9 increase in loans

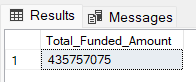


**KPI - TOTAL FUNDED AMOUNT**

1. Funded amount is Loan Amount in our dataset

--Total Funded Amount

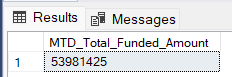
Select Sum(loan\_amount) as 'Total\_Funded\_Amount' from bank\_loan\_data;



1. --MTD Total Funded Amount

Select Sum(loan\_amount) as 'MTD\_Total\_Funded\_Amount' from bank\_loan\_data

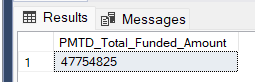
where MONTH(issue\_date) = 12 and YEAR(issue\_date) = 2021;



1. --Previous MTD Total Funded Amount

Select Sum(loan\_amount) as 'PMTD\_Total\_Funded\_Amount' from bank\_loan\_data

where MONTH(issue\_date) = 11 and YEAR(issue\_date) = 2021;

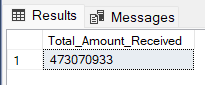


**KPI - TOTAL AMOUNT RECEIVED**

1. --Total\_Amount\_Received

Amount received is total\_payment in our dataset

Select Sum(total\_payment) as 'Total\_Amount\_Received' from bank\_loan\_data;



1. --MTD Total\_Amount\_Received

Select Sum(total\_payment) as 'MTD\_Total\_Amount\_Received' from bank\_loan\_data

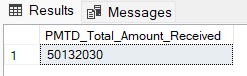
where MONTH(issue\_date) = 12 and YEAR(issue\_date) = 2021;



1. --Previous MTD Total Amount Received

Select Sum(total\_payment) as 'PMTD\_Total\_Amount\_Received' from bank\_loan\_data

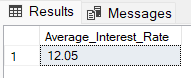
where MONTH(issue\_date) = 11 and YEAR(issue\_date) = 2021;



**KPI – AVERAGE INTEREST RATE**

1. Average\_Interest\_Rate

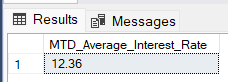
Select ROUND(avg(int\_rate),4) \* 100 as 'Average\_Interest\_Rate' from bank\_loan\_data;

****

1. MTD Average\_Interest\_Rate

Select ROUND(avg(int\_rate),4) \* 100 as 'MTD\_Average\_Interest\_Rate' from bank\_loan\_data

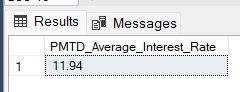
where MONTH(issue\_date) = 12 and YEAR(issue\_date) = 2021;

****

1. Previous MTD Average\_Interest\_Rate

Select ROUND(avg(int\_rate),4) \* 100 as 'PMTD\_Average\_Interest\_Rate' from bank\_loan\_data

where MONTH(issue\_date) = 11 and YEAR(issue\_date) = 2021;

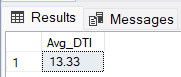


12.36 > 11.94 IR, so bank is having profit

**KPI – AVERAGE DEBT-TO-INTEREST (DTI) RATIO**

1. Avg Debt-to-Income Ratio(DTI)

Select ROUND(avg(dti),4) \* 100 as 'Avg\_DTI' from bank\_loan\_data;

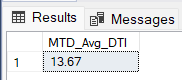


1. MTD Avg Debt-to-Income Ratio(DTI)

Range of DTI between 30-35% is considered good for most of the banks

Select ROUND(avg(dti),4) \* 100 as 'MTD\_Avg\_DTI' from bank\_loan\_data

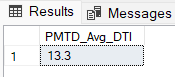
where MONTH(issue\_date) = 12 and YEAR(issue\_date) = 2021;



1. PMTD Avg Debt-to-Income Ratio(DTI)

Select ROUND(avg(dti),4) \* 100 as 'PMTD\_Avg\_DTI' from bank\_loan\_data

where MONTH(issue\_date) = 11 and YEAR(issue\_date) = 2021;

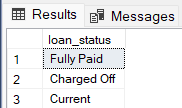


**GOOD LOAN vs BAD LOAN**

-- GOOD Loan vs BAD Loan

Select loan\_status from bank\_loan\_data;

Select DISTINCT(loan\_status) from bank\_loan\_data;



-- Good loan = "Fully Paid" & "Current", Bad loan = "Charged Off"

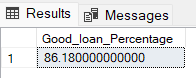
**KPI – GOOD LOAN Calculations**

1. Good Loan Application %

Select ROUND((COUNT(CASE WHEN loan\_status = 'Fully Paid' Or loan\_status = 'Current' THEN id END) \* 100.0)

/

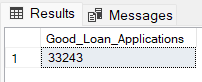
(COUNT(id)),2) AS 'Good\_loan\_Percentage' from bank\_loan\_data;



1. Good Loan Applications

select Count(id) as 'Good\_Loan\_Applications' from bank\_loan\_data

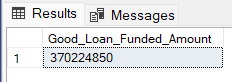
where loan\_status = 'Fully Paid' Or loan\_status = 'Current';



1. Good Loan Funded Amount

select SUM(loan\_amount) as 'Good\_Loan\_Funded\_Amount' from bank\_loan\_data

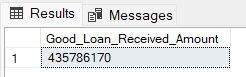
Where loan\_status = 'Fully Paid' Or loan\_status = 'Current';



1. Good Loan Total Received Amount

select sum(total\_payment) as 'Good\_Loan\_Received\_Amount' from bank\_loan\_data

Where loan\_status = 'Fully Paid' Or loan\_status = 'Current';



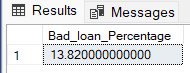
**KPI – BAD LOAN Calculations**

1. Bad Loan Application %

Select ROUND((COUNT(CASE WHEN loan\_status = 'Charged Off' THEN id END) \* 100.0)

/

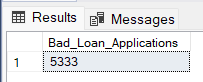
(COUNT(id)),2) AS 'Bad\_loan\_Percentage' from bank\_loan\_data;



1. Bad Loan Applications

select Count(id) as 'Bad\_Loan\_Applications' from bank\_loan\_data

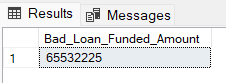
where loan\_status = 'Charged Off';



1. Bad Loan Funded Amount

select SUM(loan\_amount) as 'Bad\_Loan\_Funded\_Amount' from bank\_loan\_data

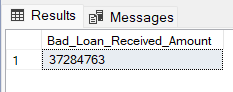
Where loan\_status = 'Charged Off';



1. Bad Loan Total Received Amount

select sum(total\_payment) as 'Bad\_Loan\_Received\_Amount' from bank\_loan\_data

Where loan\_status = 'Charged Off';



**GRID VIEW**

**KPI – Loan Status**

1. Loan Status Grid View

select loan\_status,

COUNT(id) as 'Loan\_Count',

SUM(total\_payment) as 'Total\_Received\_Amount',

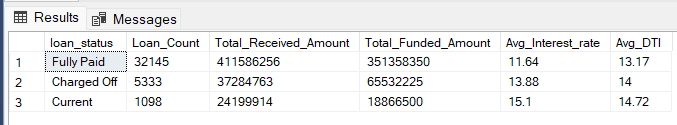
SUM(loan\_amount) as 'Total\_Funded\_Amount' ,

ROUND(AVG(int\_rate\*100),2) as 'Avg\_Interest\_rate',

ROUND(AVG(dti\*100),2) as 'Avg\_DTI'

from bank\_loan\_data

group by loan\_status;



1. MTD Loan Status Grid View

select loan\_status,

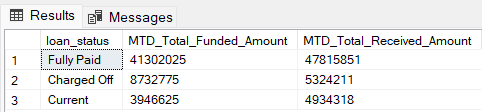
SUM(loan\_amount) as 'MTD\_Total\_Funded\_Amount' ,

SUM(total\_payment) as 'MTD\_Total\_Received\_Amount'

from bank\_loan\_data

where MONTH(issue\_date) = 12 and YEAR(issue\_date) = 2021

group by loan\_status;



**BANK LOAN REPORT | OVERVIEW**

**KPI Metrics to be shown:**

**'Total Loan Applications',**

**'Total Funded Amount',**

**'Total Amount Received**

1. Chart - Monthly trends by Issue Date

SELECT

MONTH(issue\_date) AS 'Month\_Number',

DATENAME(MONTH, issue\_date) AS 'Month\_name',

COUNT(id) AS 'Total\_Loan\_Applications',

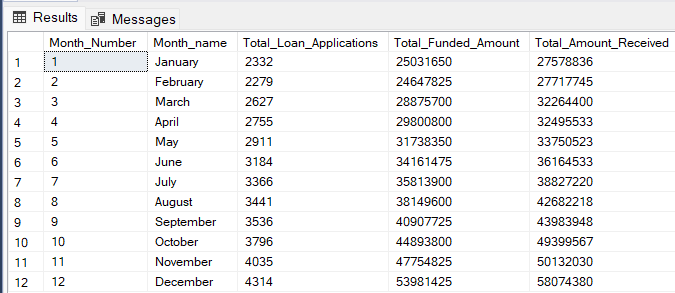
SUM(loan\_amount) AS 'Total\_Funded\_Amount',

SUM(total\_payment) AS 'Total\_Amount\_Received'

FROM bank\_loan\_data

GROUP BY MONTH(issue\_date), DATENAME(MONTH, issue\_date)

ORDER BY MONTH(issue\_date);



1. Chart - **Regional Analysis by State**

select address\_state as 'State',

COUNT(id) AS 'Total\_Loan\_Applications',

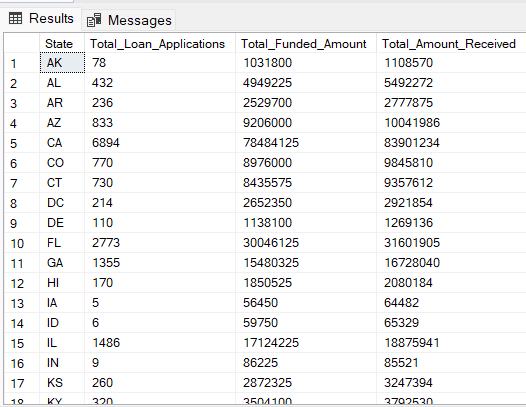
SUM(loan\_amount) AS 'Total\_Funded\_Amount',

SUM(total\_payment) AS 'Total\_Amount\_Received'

from bank\_loan\_data

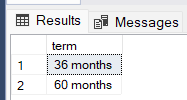
group by address\_state

order by address\_state;



1. **Chart - Loan Term Analysis**

select DISTINCT(term) from bank\_loan\_data;



select term as 'Term',

COUNT(id) AS 'Total\_Loan\_Applications',

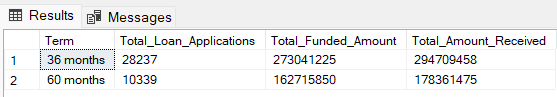
SUM(loan\_amount) AS 'Total\_Funded\_Amount',

SUM(total\_payment) AS 'Total\_Amount\_Received'

from bank\_loan\_data

group by term

order by term;



1. **Chart - Employee Length Analysis**

select emp\_length as 'Employee Length',

COUNT(id) AS 'Total\_Loan\_Applications',

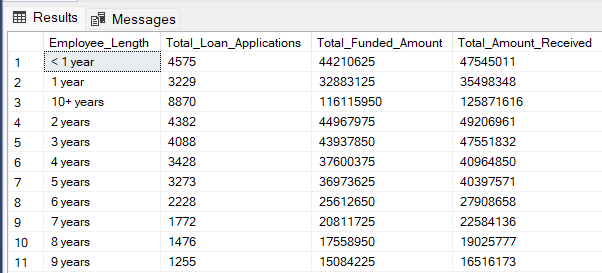
SUM(loan\_amount) AS 'Total\_Funded\_Amount',

SUM(total\_payment) AS 'Total\_Amount\_Received'

from bank\_loan\_data

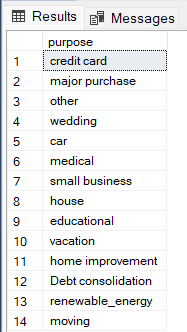
group by emp\_length

order by emp\_length;



1. Chart - **Loan Purpose Breakdown**

select DISTINCT(purpose) from bank\_loan\_data;



select purpose as 'Loan\_Purpose',

COUNT(id) AS 'Total\_Loan\_Applications',

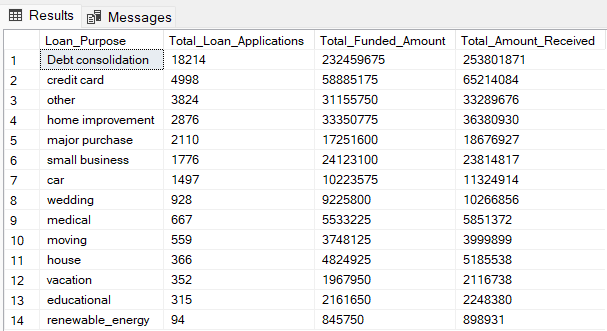
SUM(loan\_amount) AS 'Total\_Funded\_Amount',

SUM(total\_payment) AS 'Total\_Amount\_Received'

from bank\_loan\_data

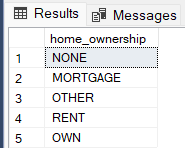
group by purpose

order by COUNT(id) desc;



1. **Chart - Home Ownership Analysis**

select DISTINCT(home\_ownership) from bank\_loan\_data;



select home\_ownership as 'Home\_Ownership',

COUNT(id) AS 'Total\_Loan\_Applications',

SUM(loan\_amount) AS 'Total\_Funded\_Amount',

SUM(total\_payment) AS 'Total\_Amount\_Received'

from bank\_loan\_data

group by home\_ownership

order by COUNT(id) desc;

