Bank Loan Analysis

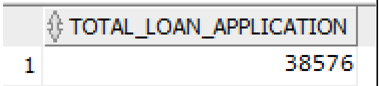
Data Validation Queries and DAX Measures

Summary Page Data Validation (Oracle SQL 11g)

Loan Applications

Total Loan Applications

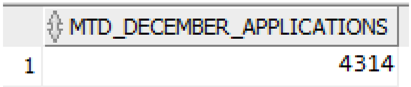
```
SELECT COUNT(*) total_loan_application FROM bank_loan_data;
-- 38576
```



	TOTAL_LOAN_APPLICATION
1	38576

MTD Loan Applications

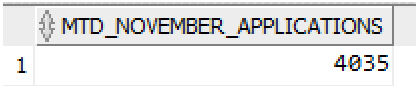
```
-- Counts all loan applications issued in December 2021.

SELECT
   COUNT(*) AS mtd_december_applications
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-12-01'
   AND issue_date < DATE '2022-01-01';
-- 4314
```



	MTD_DECEMBER_APPLICATIONS
1	4314

PMTD Loan Applications

```
-- Counts all loan applications issued in November 2021.

SELECT
   COUNT(*) AS mtd_november_applications
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-11-01'
   AND issue_date < DATE '2021-12-01';
-- 4035
```



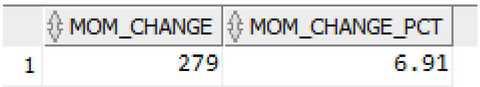
	MTD_NOVEMBER_APPLICATIONS
1	4035

```
/* MoM change and percentage change in loan application

MoM Change = Current Month Value - Previous Month Value

MoM % Change = (Current - Previous)/ Previous × 100
(4314 - 4035) / 4035 x 100 = 6.91
*/
```

MoM Total Loan Applications (MoM change in November and December)

```
SELECT
  dec_cnt - nov_cnt AS mom_change,
  ROUND((dec_cnt - nov_cnt) / nov_cnt * 100, 2) AS mom_change_pct
```

```
FROM (
  SELECT
    COUNT(CASE
            WHEN ISSUE_DATE >= DATE '2021-12-01'
             AND ISSUE_DATE <  DATE '2022-01-01'
            THEN 1
          END) AS dec_cnt,
    COUNT(CASE
            WHEN ISSUE_DATE >= DATE '2021-11-01'
             AND ISSUE_DATE <  DATE '2021-12-01'
            THEN 1
          END) AS nov_cnt
  FROM bank_loan_data
);
```

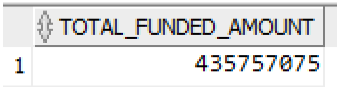


	MOM_CHANGE	MOM_CHANGE_PCT
1	279	6.91

Funded Amount

Total Funded Amount

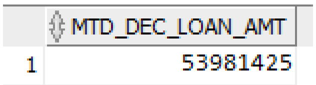
```
SELECT
   SUM(loan_amount) AS total_funded_amount
FROM
   bank_loan_data;
-- 435757075
```



	TOTAL_FUNDED_AMOUNT
1	435757075

MTD Total Funded Amount

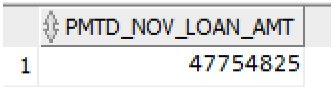
```
-- Month to Date (MTD) total loan given by bank for dec

SELECT
   SUM(loan_amount) AS mtd_dec_loan_amt
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-12-01'
   AND issue_date < DATE '2022-01-01';
-- 53981425
```



	MTD_DEC_LOAN_AMT
1	53981425

-- PMTD Total Funded Amount

```
-- Previous Month to Date (PMTD) loan amount (nov)

SELECT
   SUM(loan_amount) AS pmtd_nov_loan_amt
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-11-01'
   AND issue_date < DATE '2021-12-01';
-- 47754825
```

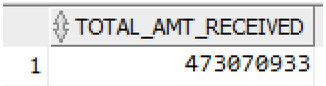


	PMTD_NOV_LOAN_AMT
1	47754825

Amount Received

Total Amount Received

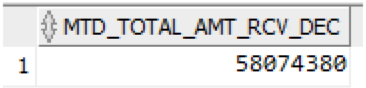
```
--- Total loan Received / recovered from customers

SELECT
   SUM(total_payment) AS total_amt_received
FROM
   bank_loan_data;
-- 473070933
```



	TOTAL_AMT_RECEIVED
1	473070933

MTD Total Amount Received

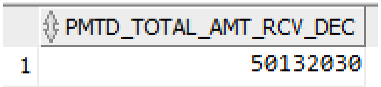
```
-- Month to Date (MTD) total loan Amt Received -- December

SELECT
   SUM(total_payment) AS mtd_total_amt_rcv_dec
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-12-01'
   AND issue_date < DATE '2022-01-01';
-- 58074380
```



	MTD_TOTAL_AMT_RCV_DEC
1	58074380

PMTD Total Amount Received

```
-- Previous Month to Date (PMTD) total loan Amt Received - November

SELECT
   SUM(total_payment) AS pmtd_total_amt_rcv_dec
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-11-01'
   AND issue_date < DATE '2021-12-01';
-- 50132030
```

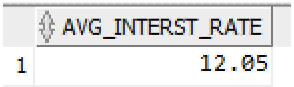


	PMTD_TOTAL_AMT_RCV_DEC
1	50132030

Average Interest Rate

-- Avg interest rate

```
SELECT
   round(AVG(int_rate) * 100, 2) AS avg_interst_rate
FROM
   bank_loan_data;
-- 12.05
```

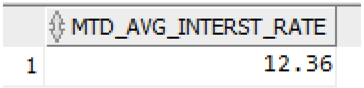


	AVG_INTERST_RATE
1	12.05

-- MTD Average Interest

```
-- Month to Date (MTD) Avg interest rate for December

SELECT
```

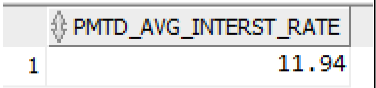
```
   round(AVG(int_rate) * 100, 2) AS mtd_avg_interst_rate
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-12-01'
   AND issue_date < DATE '2022-01-01';
-- 12.36
```



	MTD_AVG_INTERST_RATE
1	12.36

--PMTD Average Interest

```
-- Previous Month to Date (PMTD) Avg interest rate for November

SELECT
   round(AVG(int_rate) * 100, 2) AS pmtd_avg_interst_rate
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-11-01'
   AND issue_date < DATE '2021-12-01';
-- 11.94
```

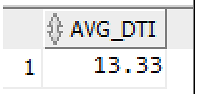


	PMTD_AVG_INTERST_RATE
1	11.94

Avg Debt-to-Income (DTI) Ratio

Avg Debt-to-Income (DTI) Ratio

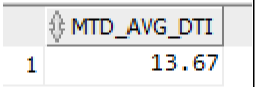
```
SELECT
   round(AVG(dti) * 100, 2) as Avg_DTI
FROM
   bank_loan_data;
-- 13.33
```



	AVG_DTI
1	13.33

MTD Avg DTI Ratio

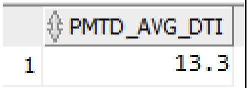
```
-- Month-to-Date (MTD) Avg Debt-to-Income (DTI) for December

SELECT
   round(AVG(dti * 100), 2) AS mtd_avg_dti
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-12-01'
   AND issue_date < DATE '2022-01-01';
-- 13.67
```



	MTD_AVG_DTI
1	13.67

PMTD Avg DTI Ratio

```
-- Previous Month-to-Date (PMTD) Avg Debt-to-Income (DTI) for November
SELECT
   round(AVG(dti * 100), 2) AS pmtd_avg_dti
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-11-01'
```



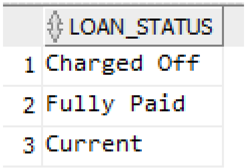
	PMTD_AVG_DTI
1	13.3

```
   AND issue_date < DATE '2021-12-01';
-- 13.3
```

Good loan issued

Types of Loan Status

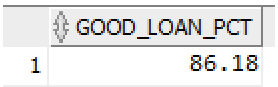
```
SELECT DISTINCT
   ( loan_status )
FROM
   bank_loan_data;
```



	LOAN_STATUS
1	Charged Off
2	Fully Paid
3	Current

Good Loan Percentage

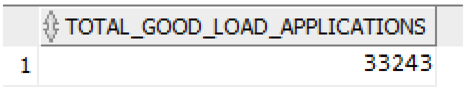
```
SELECT
   round((COUNT(
      CASE
         WHEN loan_status = 'Fully Paid'
              OR loan_status = 'Current' THEN
            id
      END
   ) * 100) / COUNT(id), 2) AS good_loan_pct
FROM
   bank_loan_data;
-- 86.18
```



	GOOD_LOAN_PCT
1	86.18

Good Loan Applications

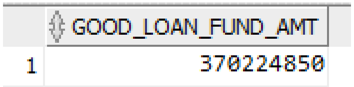
```
-- Number of Good load Applications

SELECT
   COUNT(loan_status) AS total_good_load_applications
FROM
   bank_loan_data
WHERE
   loan_status = 'Fully Paid'
   OR loan_status = 'Current';
-- 33243
```



	TOTAL_GOOD_LOAD_APPLICATIONS
1	33243

Good Loan Funded Amount

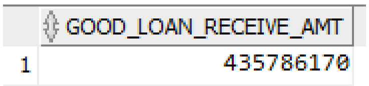
```
SELECT
   SUM(loan_amount) AS good_loan_fund_amt
FROM
   bank_loan_data
WHERE
   loan_status = 'Fully Paid'
   OR loan_status = 'Current';
-- 370224850
```



	GOOD_LOAN_FUND_AMT
1	370224850

Good Loan Received Amount

```
SELECT
   SUM(total_payment) AS good_loan_receive_amt
FROM
   bank_loan_data
WHERE
   loan_status = 'Fully Paid'
   OR loan_status = 'Current';
-- 435786170
```

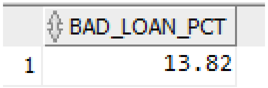


	GOOD_LOAN_RECEIVE_AMT
1	435786170

Bad loan issued

Bad Loan Percentage

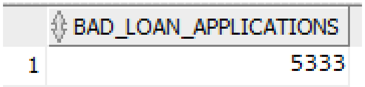
```
SELECT
   round((COUNT(
      CASE
         WHEN loan_status = 'Charged Off' THEN  id
      END
   ) * 100) / COUNT(id), 2) AS bad_loan_pct
FROM bank_loan_data;
-- 13.824
```



	BAD_LOAN_PCT
1	13.82

Bad Loan Applications

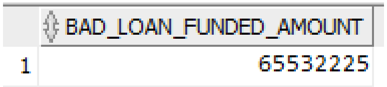
```
-- Total applications of Bad loan

SELECT
   COUNT(id) AS bad_loan_applications
FROM
   bank_loan_data
WHERE
   loan_status = 'Charged Off';
-- 5333
```



	BAD_LOAN_APPLICATIONS
1	5333

Bad Loan Funded Amount

```
SELECT
   SUM(loan_amount) AS bad_loan_funded_amount
FROM
   bank_loan_data
WHERE
   loan_status = 'Charged Off';
-- 65532225
```

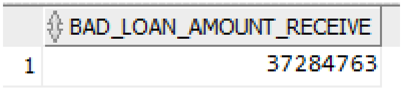


	BAD_LOAN_FUNDED_AMOUNT
1	65532225

Bad Loan Received Amount

```
SELECT
   SUM(total_payment) AS bad_loan_amount_receive
```

```
FROM
   bank_loan_data
WHERE
   loan_status = 'Charged Off';
-- 37284763
```

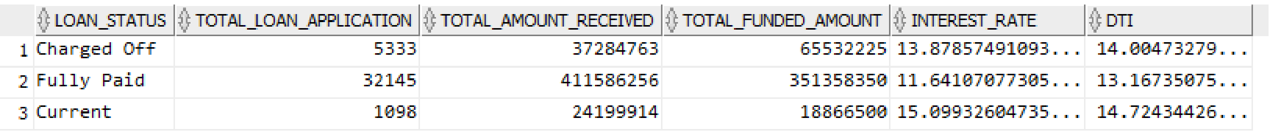


	BAD_LOAN_AMOUNT_RECEIVE
1	37284763

Loan status

Details of Loan Status

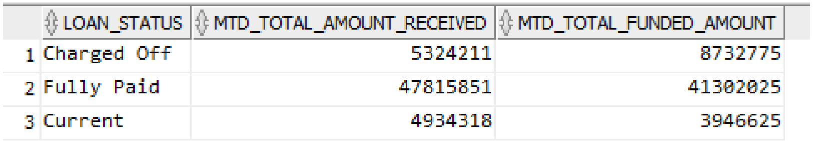
```
SELECT
   loan_status,
   COUNT(id) AS total_loan_application,
   SUM(total_payment) AS total_amount_received,
   SUM(loan_amount) AS total_funded_amount,
   AVG(int_rate * 100) AS interest_rate,
   AVG(dti * 100) AS dti
FROM
   bank_loan_data
GROUP BY
   loan_status;
```



	LOAN_STATUS	TOTAL_LOAN_APPLICATION	TOTAL_AMOUNT_RECEIVED	TOTAL_FUNDED_AMOUNT	INTEREST_RATE	DTI
1	Charged Off	5333	37284763	65532225	13.87857491093...	14.00473279...
2	Fully Paid	32145	411586256	351358350	11.64107077305...	13.16735075...
3	Current	1098	24199914	18866500	15.09932604735...	14.72434426...

-- Month-to-Date MTD loan amount given and loan amount received

```
SELECT
   loan_status,
   SUM(total_payment) AS mtd_total_amount_received,
   SUM(loan_amount) AS mtd_total_funded_amount
FROM
   bank_loan_data
WHERE
   issue_date >= DATE '2021-12-01'
   AND issue_date < DATE '2022-01-01'
GROUP BY
   loan_status;
```

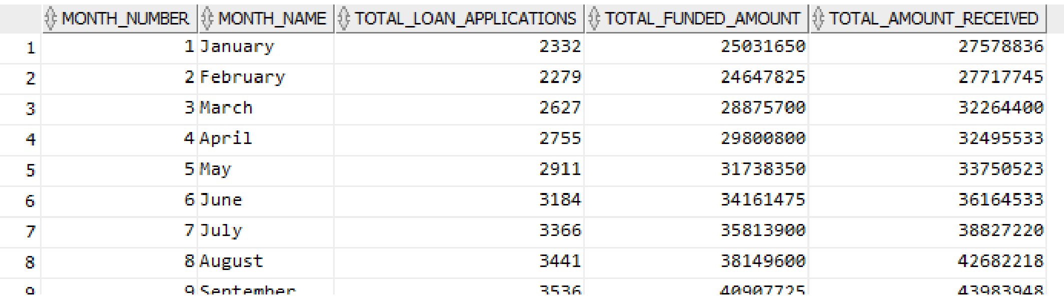


	LOAN_STATUS	MTD_TOTAL_AMOUNT_RECEIVED	MTD_TOTAL_FUNDED_AMOUNT
1	Charged Off	5324211	8732775
2	Fully Paid	47815851	41302025
3	Current	4934318	3946625

Bank Loan Report | Overview

Monthly Trend of Total_loan_application, Total_funded_amount, Total_amount_receive

```
SELECT
  EXTRACT(MONTH FROM issue_date) AS month_number,
  TO_CHAR(issue_date, 'Month')  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
  EXTRACT(MONTH FROM issue_date),
  TO_CHAR(issue_date, 'Month')
ORDER BY
  month_number;
```

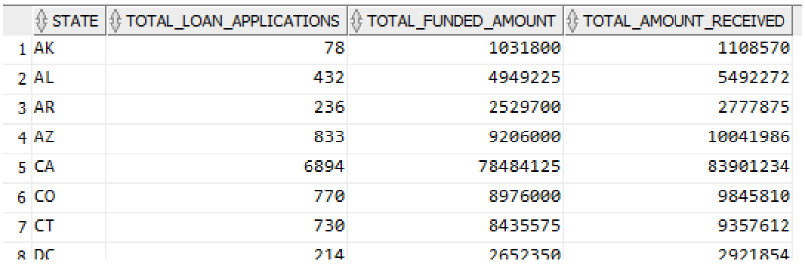


	MONTH_NUMBER	MONTH_NAME	TOTAL_LOAN_APPLICATIONS	TOTAL_FUNDED_AMOUNT	TOTAL_AMOUNT_RECEIVED
1	1	January	2332	25031650	27578836
2	2	February	2279	24647825	27717745
3	3	March	2627	28875700	32264400
4	4	April	2755	29800800	32495533
5	5	May	2911	31738350	33750523
6	6	June	3184	34161475	36164533
7	7	July	3366	35813900	38827220
8	8	August	3441	38149600	42682218
9	9	September	3536	40907725	43983948

State

-- State wise Total_Loan_Applications, Total_Funded_Amount, Total_Amount_Received

```
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;

-- we can do with Total_Loan_Applications in Desc, Total_Funded_Amount in Desc
-- replace address_state with COUNT(id), or SUM(loan_amount)
```

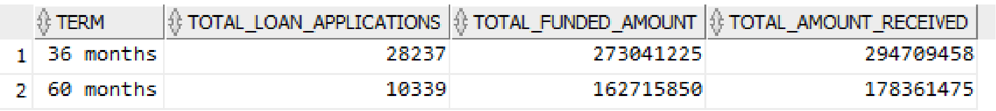


	STATE	TOTAL_LOAN_APPLICATIONS	TOTAL_FUNDED_AMOUNT	TOTAL_AMOUNT_RECEIVED
1	AK	78	1031800	1108570
2	AL	432	4949225	5492272
3	AR	236	2529700	2777875
4	AZ	833	9206000	10041986
5	CA	6894	78484125	83901234
6	CO	770	8976000	9845810
7	CT	730	8435575	9357612
8	DC	214	2652350	2921854

Term

Loan Term (months) wise Total_Loan_Applications, Total_Funded_Amount, Total_Amount_Received

```
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;
```

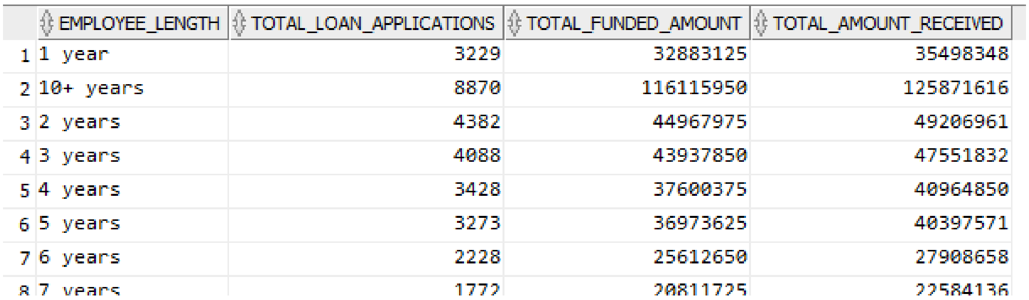


	TERM	TOTAL_LOAN_APPLICATIONS	TOTAL_FUNDED_AMOUNT	TOTAL_AMOUNT_RECEIVED
1	36 months	28237	273041225	294709458
2	60 months	10339	162715850	178361475

Employee Length

Employees working Stability (emp_lenght) wise Total_Loan_Applications, Total_Funded_Amount, Total_Amount_Received

```
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;
```



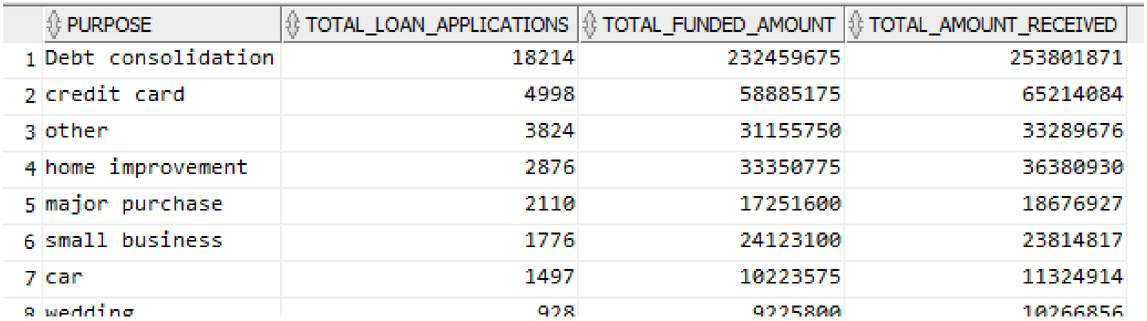
	EMPLOYEE_LENGTH	TOTAL_LOAN_APPLICATIONS	TOTAL_FUNDED_AMOUNT	TOTAL_AMOUNT_RECEIVED
1	1 year	3229	32883125	35498348
2	10+ years	8870	116115950	125871616
3	2 years	4382	44967975	49206961
4	3 years	4088	43937850	47551832
5	4 years	3428	37600375	40964850
6	5 years	3273	36973625	40397571
7	6 years	2228	25612650	27908658
8	7 years	1772	20811725	22584136

Loan Purpose

Loan Purpose wise Total_Loan_Applications, Total_Funded_Amount, Total_Amount_Received

```
SELECT
      purpose AS 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;
```

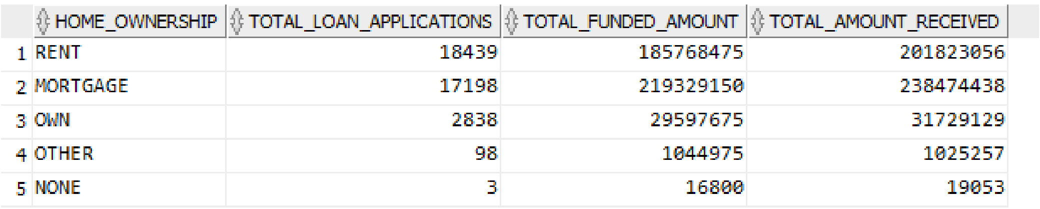


	PURPOSE	TOTAL_LOAN_APPLICATIONS	TOTAL_FUNDED_AMOUNT	TOTAL_AMOUNT_RECEIVED
1	Debt consolidation	18214	232459675	253801871
2	credit card	4998	58885175	65214084
3	other	3824	31155750	33289676
4	home improvement	2876	33350775	36380930
5	major purchase	2110	17251600	18676927
6	small business	1776	24123100	23814817
7	car	1497	10223575	11324914
8	wedding	928	9225800	10266856

Home Ownership

Home ownership wise Total_Loan_Applications, Total_Funded_Amount, Total_Amount_Received

```
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;
```



	HOME_OWNERSHIP	TOTAL_LOAN_APPLICATIONS	TOTAL_FUNDED_AMOUNT	TOTAL_AMOUNT_RECEIVED
1	RENT	18439	185768475	201823056
2	MORTGAGE	17198	219329150	238474438
3	OWN	2838	29597675	31729129
4	OTHER	98	1044975	1025257
5	NONE	3	16800	19053

DAX Measures | Column | Table | Parameter Field | Data Modelling

- **Date Table & Month columns -** Enable time intelligence and correct date-based analysis
- **Total measures (Applications, Funded, Received) -** Track overall lending performance
- **MTD / PMTD measures -** Compare current month performance with previous month
- **MoM measures -** Measure growth or decline trends over time
- **Average Interest Rate & DTI -** Analyze pricing strategy and borrower risk
- **Good vs Bad Loan measures -** Evaluate portfolio quality and credit risk
- **Field Parameters -** Dynamically switch KPIs in visuals for better interactivity
- **Navigation Buttons -** Improve dashboard usability and user experience

---

New Date/Calender Table

- Creates a continuous date table based on loan issue_dates column (required for time intelligence).

```
Date_Table = CALENDAR(MIN(BANK_LOAN_DATA[ISSUE_DATE]),MAX(BANK_LOAN_DATA[ISSUE_DATE]))
```

Adding new column in New Date_Table

- Calculated Columns / Date Attributes display month name in visuals
- Correct month sorting (Jan–Dec)

```
Month = FORMAT(Date_Table[Date], "mmm")
Month number = MONTH(Date_Table[Date])
```

KPI

DAX Measure Calculates:

- Total
- Month-To-Date
- Previous month’s MTD
- Month-over-Month growth/decline

Loan Applications

```
Total Loan Applications = COUNT(BANK_LOAN_DATA[ID])

MTD Total Funded Amount = CALCULATE(TOTALMTD([Total Funded Amount], Date_Table[Date]))

PMTD Total Loan Applications = CALCULATE([Total Loan Applications],
DATESMTD(DATEADD(Date_Table[Date],-1,MONTH)))

MoM Total Loan Applications = ([MTD Total Loan Applications] - [PMTD Total Loan
Applications])/ [PMTD Total Loan Applications]
```

Funded Amount

```
Total Funded Amount = SUM(BANK_LOAN_DATA[LOAN_AMOUNT])

MTD Total Funded Amount = CALCULATE(TOTALMTD([Total Funded Amount], Date_Table[Date]))

PMTD Total Funded Amount = CALCULATE([Total Funded Amount],
DATESMTD(DATEADD(Date_Table[Date],-1,MONTH)))
```

- MoM Total Funded Amount = ([MTD Total Funded Amount] - [PMTD Total Funded Amount]) / [PMTD Total Funded Amount]

Amount Received

- Total Amount Received = SUM(BANK_LOAN_DATA[TOTAL_PAYMENT])

- MTD Total Amount Received = CALCULATE(TOTALMTD([Total Amount Received], Date_Table[Date]))

- PMTD Total Amount Received = CALCULATE([Total Amount Received], DATESMTD(DATEADD(Date_Table[Date],-1,MONTH)))

- MoM Total Amount Received = ([MTD Total Amount Received] - [PMTD Total Amount Received])/ [PMTD Total Amount Received]

Avg Interest Rate

- Avg Interest Rate = AVERAGE(BANK_LOAN_DATA[INT_RATE])

- MTD Avg Int Rate = CALCULATE(TOTALMTD([Avg Interest Rate], Date_Table[Date]))

- PMTD Avg Int Rate = CALCULATE([Avg Interest Rate], DATESMTD(DATEADD(Date_Table[Date],-1,MONTH)))

- MoM Avg Int Rate = ([MTD Avg Int Rate] - [PMTD Avg Int Rate])/ [PMTD Avg Int Rate]

Debt to Income Ratio

- Avg DTI = AVERAGE(BANK_LOAN_DATA[DTI])

- MTD Avg DTI Ratio = CALCULATE(TOTALMTD([Avg DTI], Date_Table[Date]))

- PMTD Avg DTI Ratio = CALCULATE([Avg DTI], DATESMTD(DATEADD(Date_Table[Date],-1,MONTH)))

- MoM Avg DTI Ratio = ([MTD Avg DTI Ratio] - [PMTD Avg DTI Ratio])/ [PMTD Avg DTI Ratio]

Create New Group for Good Loan vs Bad Loan on *loan_status* column

---

Fully paid | Current: as Good_loan
Charged off: as Bad_loan

- Simplifies analysis of loan quality
- Enables portfolio risk comparison (Segmentation)

Good Loan

- Good Loan % = (CALCULATE([Total Loan Applications], BANK_LOAN_DATA[Good Loan Vs Bad loan]= "Good Loan"))/ [Total Loan Applications]

- Good Loan Applications = CALCULATE([Total Loan Applications], BANK_LOAN_DATA[Good Loan Vs Bad loan] = "Good Loan")

- Good Loan Funded Amount = CALCULATE([Total Funded Amount], BANK_LOAN_DATA[Good Loan Vs Bad loan] = "Good Loan")

- Good Loan Received Amount = CALCULATE([Total Amount Received], BANK_LOAN_DATA[Good Loan Vs Bad loan] = "Good Loan")

Bad Loan

- Bad Loan % = (CALCULATE([Total Loan Applications], BANK_LOAN_DATA[Good Loan Vs Bad loan]= "Bad Loan"))/ [Total Loan Applications]

- Bad Loan Applications = CALCULATE([Total Loan Applications], BANK_LOAN_DATA[Good Loan Vs Bad loan] = "Bad Loan")

- Bad Loan Funded Amount = CALCULATE([Total Funded Amount], BANK_LOAN_DATA[Good Loan Vs Bad loan] = "Bad Loan")

- Bad Loan Received Amount = CALCULATE([Total Amount Received], BANK_LOAN_DATA[Good Loan Vs Bad loan] = "Bad Loan")

New Field Parameter

Total Amount Received, Total Funded Amount, Total Loan Applications

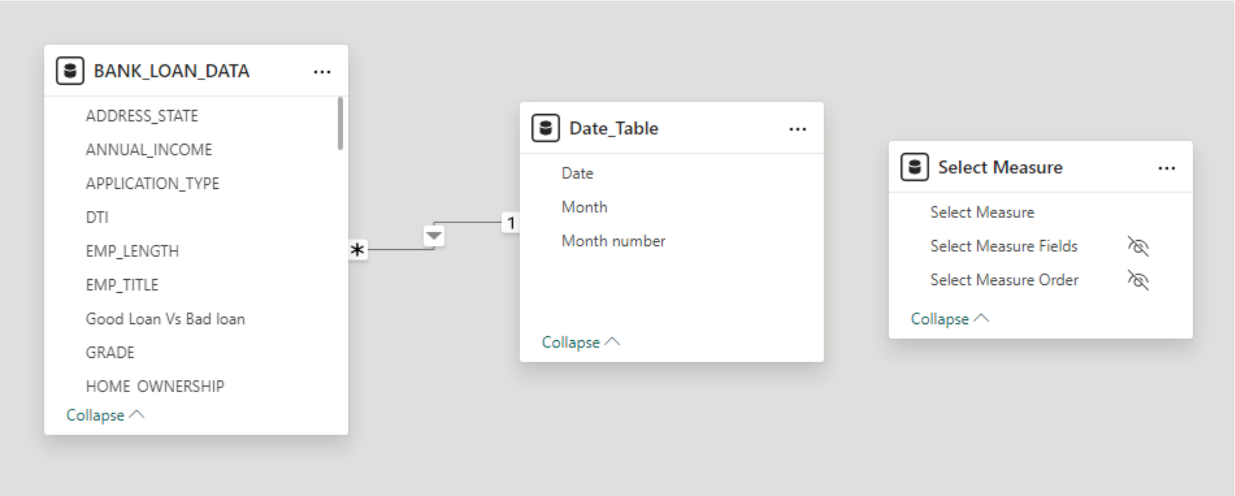
- Allows users to dynamically switch KPIs
- Reduces multiple visuals into one
- Improves dashboard interactivity
- **Dynamic Measure Selection / UX Optimization**

```
Select Measure = {
    ("Total Amount Received", NAMEOF('BANK_LOAN_DATA'[Total Amount Received]), 0),
    ("Total Funded Amount", NAMEOF('BANK_LOAN_DATA'[Total Funded Amount]), 1),
    ("Total Loan Applications", NAMEOF('BANK_LOAN_DATA'[Total Loan Applications]), 2)
}
```

Added navigation Buttons

- Enable page-to-page navigation
- Improve user experience
- Make dashboard interactive and guided
- Report Navigation

Data Modeling


BANK_LOAN_DATA
ADDRESS_STATE
ANNUAL_INCOME
APPLICATION_TYPE
DTI
EMP_LENGTH
EMP_TITLE
Good Loan Vs Bad loan
GRADE
HOME OWNERSHIP
Collapse
*
1
Date_Table
Date
Month
Month number
Collapse
Select Measure
Select Measure
Select Measure Fields
Select Measure Order
Collapse