Bank Loan Report Query Document
1. Bank Loan Report

KPI's:

Total Loan Applications
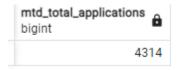
SELECT COUNT(id) AS Total_Applications FROM loans

total_applications 🔒 bigint
38576

MTD Loan Applications

SELECT COUNT(id) AS MTD_Total_Applications FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 12 AND EXTRACT(YEAR FROM issue_date) = 2021;

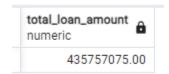| mtd_total_applications 🔒 bigint |
|---|
| 4314 |

PMTD Loan Applications

SELECT COUNT(id) AS PMTD_Total_Applications FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 11 AND EXTRACT(YEAR FROM issue_date) = 2021;

pmtd_total_applications 🔒 bigint
4035

Total Loan Amount

SELECT SUM(loan_amount) AS Total_Loan_Amount FROM loans

total_loan_amount 🔒 numeric
435757075.00

MTD Loan Amount

SELECT SUM(loan_amount) AS Total_Loan_Amount FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 12 AND EXTRACT(YEAR FROM issue_date) = 2021;

mtd_total_loan_amount 🔒 numeric
53981425.00

PMTD Loan Amount

```sql
SELECT SUM(loan_amount) AS Total_Loan_Amount FROM loans
WHERE EXTRACT(MONTH FROM issue_date) = 11 AND EXTRACT(YEAR FROM issue_date)
= 2021;
```

pmtd_total_loan_amount numeric 🔒
47754825.00

Total Amount Received

```sql
SELECT SUM(total_payment) AS Total_Amount_Collected FROM loans
```

total_amount_received numeric 🔒
473070933.00

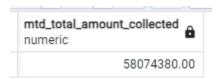### MTD Total Amount Received

```sql
SELECT SUM(total_payment) AS MTD_Total_Amount_Collected FROM loans
WHERE EXTRACT(MONTH FROM issue_date) = 12  AND EXTRACT(YEAR FROM
issue_date) = 2021;
```

mtd_total_amount_collected numeric 🔒
58074380.00

PMTD Total Amount Received

```sql
SELECT SUM(total_payment) AS PMTD_Total_Amount_Collected FROM loans
WHERE EXTRACT(MONTH FROM issue_date) = 12  AND EXTRACT(YEAR FROM
issue_date) = 2021;
```

pmtd_total_amount_collected numeric 🔒
50132030.00

Average Interest Rate

```sql
SELECT AVG(int_rate)*100 AS Avg_Int_Rate FROM loans
```

avg_int_rate numeric 🔒
12.04883139776026545000

DTM Average Interest Rate

SELECT AVG(int_rate)*100 AS DTM_Avg_Int_Rate FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 12;

dtm_avg_int_rate numeric 🔒
12.35604079740380157600

PDTM Average Interest Rate

SELECT AVG(int_rate)*100 AS PDTM_Avg_Int_Rate FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 12;

pdtm_avg_int_rate numeric 🔒
11.94171747211895910800

Average DTI

SELECT ROUND(AVG(dti),2)*100 AS Avg_DTI FROM loans

avg_dti numeric 🔒
13.33

DTM Average DTI

SELECT ROUND(AVG(dti),2)*100 AS DTM_Avg_DTI FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 12;

dtm_avg_dti numeric 🔒
13.67

PDTM Average DTI

SELECT ROUND(AVG(dti),2)*100 AS PDTM_Avg_DTI FROM loans

WHERE EXTRACT(MONTH FROM issue_date) = 11;

pdtm_avg_dti numeric 🔒
13.30

GOOD LOAN ISSUED

Good Loan Percentage

SELECT
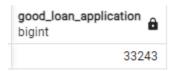 (COUNT(CASE WHEN loan_status = 'Fully Paid' OR loan_status = 'Current' THEN id END) *
100.0) /
      COUNT(id) AS Good_Loan_Percentage
FROM loans

good_loan_percentage
bigint

good_loan_percentage bigint
86

Good Loan Applications

SELECT COUNT(id) AS Good_Loan_Applications FROM loans
WHERE loan_status = 'Fully Paid' OR loan_status = 'Current';

good_loan_application bigint
33243

Good Loan Funded Amount

SELECT SUM(loan_amount) AS Good_Loan_Funded_amount FROM loans
WHERE loan_status = 'Fully Paid' OR loan_status = 'Current';

good_loan_funded_amount numeric
370224850.00

Good Loan Amount Received

SELECT SUM(total_payment) AS Good_Loan_amount_received FROM loans
WHERE loan_status = 'Fully Paid' OR loan_status = 'Current';

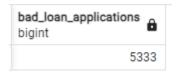| good_loan_amount_received
numeric |
| --- |
| 435786170.00 |

BAD LOAN ISSUED

Bad Loan Percentage

```
SELECT
    (COUNT(CASE WHEN loan_status = 'Charged Off' THEN id END) * 100.0) /
        COUNT(id) AS Bad_Loan_Percentage
FROM loans;
```

bad_loan_percentage 🔒
numeric
13.8246578183326421

Bad Loan Applications

```
SELECT COUNT(id) AS Bad_Loan_Applications FROM loans
WHERE loan_status = ''Charged Off';
```

bad_loan_applications 🔒
bigint
5333

Bad Loan Funded Amount

```
SELECT SUM(loan_amount) AS Bad_Loan_Funded_amount FROM loans
WHERE loan_status = 'Charged Off;
```

bad_loan_funded_amount 🔒
numeric
65532225.00

Bad Loan Amount Received

```
SELECT SUM(total_payment) AS Bad_Loan_amount_received FROM loans
WHERE loan_status = 'Charged Off';
```

bad_loan_amount_received 🔒
numeric
37284763.00

LOAN STATUS

```sql
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
    loans
GROUP BY
    Loan_status;
```

loan_status character varying (50)	total_loan_application bigint	total_amount_received numeric	total_funded_amount numeric	interest_rate numeric	dti numeric
Current	1098	24199914.00	18866500.00	15.0993260473588342	14.7243442622950820
Fully Paid	32145	411586256.00	351358350.00	11.6410707730595738	13.1673507543941515
Charged Off	5333	37284763.00	65532225.00	13.8785749109319332	14.0047327957997375

```sql
SELECT
    loan_status,
    SUM(total_payment) AS MTD_Total_Amount_Received,
    SUM(loan_amount) AS MTD_Total_Funded_Amount
FROM loans
WHERE MONTH(issue_date) = 12
GROUP BY loan_status;
```

loan_status character varying (50)	mtd_total_amount_received numeric	mtd_total_funded_amount numeric
Charged Off	5324211.00	8732775.00
Current	4934318.00	3946625.00
Fully Paid	47815851.00	41302025.00

BANK LOAN REPORT | OVERVIEW

MONTH

```sql
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 loans
GROUP BY EXTRACT(MONTH FROM issue_date), TO_CHAR(issue_date, 'Month')
ORDER BY Month_Number;
```

month_number numeric	month_name text	total_loan_applications bigint	total_funded_amount numeric	total_amount_received numeric
1	January	2332	25031650.00	27578836.00
2	February	2279	24647825.00	27717745.00
3	March	2627	28875700.00	32264400.00
4	April	2755	29800800.00	32495533.00
5	May	2911	31738350.00	33750523.00
6	June	3184	34161475.00	36164533.00
7	July	3366	35813900.00	38827220.00
8	August	3441	38149600.00	42682218.00
9	September	3536	40907725.00	43983948.00
10	October	3796	44893800.00	49399567.00
11	November	4035	47754825.00	50132030.00
12	December	4314	53981425.00	58074380.00

STATE

```sql
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 loans
GROUP BY address_state
ORDER BY address_state;
```

	state character varying (2)	total_loan_applications bigint	total_funded_amount numeric	total_amount_received numeric
1	AK	78	1031800.00	1108570.00
2	AL	432	4949225.00	5492272.00
3	AR	236	2529700.00	2777875.00
4	AZ	833	9206000.00	10041986.00
5	CA	6894	78484125.00	83901234.00
6	CO	770	8976000.00	9845810.00
7	CT	730	8435575.00	9357612.00
8	DC	214	2652350.00	2921854.00
9	DE	110	1138100.00	1269136.00
10	FL	2773	30046125.00	31601905.00
11	GA	1355	15480325.00	16728040.00
12	HI	170	1850525.00	2080184.00
13	IA	5	56450.00	64482.00
14	ID	6	59750.00	65329.00
15	IL	1486	17124225.00	18875941.00

TERM

```
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 loans
GROUP BY term
ORDER BY term;
```

	term character varying (20)	total_loan_applications bigint	total_funded_amount numeric	total_amount_received numeric
1	36 months	28237	273041225.00	294709458.00
2	60 months	10339	162715850.00	178361475.00

EMPLOYEE LENGTH

```
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 loans
GROUP BY emp_length
ORDER BY emp_length;
```

	employee_length character varying (20)	total_loan_applications bigint	total_funded_amount numeric	total_amount_received numeric
1	< 1 year	4575	44210625.00	47545011.00
2	1 year	3229	32883125.00	35498348.00
3	10+ years	8870	116115950.00	125871616.00
4	2 years	4382	44967975.00	49206961.00
5	3 years	4088	43937850.00	47551832.00
6	4 years	3428	37600375.00	40964850.00
7	5 years	3273	36973625.00	40397571.00
8	6 years	2228	25612650.00	27908658.00
9	7 years	1772	20811725.00	22584136.00
10	8 years	1476	17558950.00	19025777.00
11	9 years	1255	15084225.00	16516173.00

PURPOSE

```sql
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 loans
GROUP BY purpose
ORDER BY purpose;
```

	purpose character varying (50) 🔒	total_loan_applications 🔒 bigint	total_funded_amount 🔒 numeric	total_amount_received 🔒 numeric
1	car	1497	10223575.00	11324914.00
2	credit card	4998	58885175.00	65214084.00
3	Debt consolidation	18214	232459675.00	253801871.00
4	educational	315	2161650.00	2248380.00
5	home improvement	2876	33350775.00	36380930.00
6	house	366	4824925.00	5185538.00
7	major purchase	2110	17251600.00	18676927.00
8	medical	667	5533225.00	5851372.00
9	moving	559	3748125.00	3999899.00
10	other	3824	31155750.00	33289676.00
11	renewable_energy	94	845750.00	898931.00
12	small business	1776	24123100.00	23814817.00
13	vacation	352	1967950.00	2116738.00
14	wedding	928	9225800.00	10266856.00

HOME OWNERSHIP

```
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 loans
GROUP BY home_ownership
ORDER BY home_ownership;
```

	home_ownership character varying (20)	total_loan_applications bigint	total_funded_amount numeric	total_amount_received numeric
1	MORTGAGE	17198	219329150.00	238474438.00
2	NONE	3	16800.00	19053.00
3	OTHER	98	1044975.00	1025257.00
4	OWN	2838	29597675.00	31729129.00
5	RENT	18439	185768475.00	201823056.00

Note: We have applied multiple Filters on all the dashboards. You can check the results for the filters as well by modifying the query and comparing the results.

For e.g

See the results when we hit the Grade A in the filters for dashboards.

```
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 loans
WHERE grade = 'A'
GROUP BY purpose
ORDER BY purpose;
```