**PROJECT REPORT**

**BANK LOAN REPORT QUERY DOCUMENT**

**📊 KPI Requirements Explained**

**1. Total Loan Applications**

* Meaning: The total number of loan applications received.
* What to track:
  + **Overall Total Applications** → all-time total

SELECT COUNT(id) AS TOTAL\_LOAN\_APPLICATION FROM bank\_loan\_data ;



* + **Month-to-Date (MTD) Applications** → how many applications came from the start of the current month up to today

SELECT COUNT(id) as TOTAL\_APPLICATION FROM bank\_loan\_data WHERE

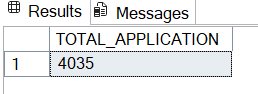
MONTH(issue\_date) = 12 ;



* + **Month-over-Month (MoM) Change** → compare the number of applications this month vs last month (percentage increase/decrease).

SELECT COUNT(id) as TOTAL\_APPLICATION FROM bank\_loan\_data WHERE

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

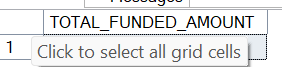


👉 This shows how many new customers are applying for loans and whether the trend is increasing or decreasing.

**2. Total Funded Amount**

* Meaning: The total amount of money the bank has given out as loans.
* What to track:
  + **Overall Funded Amount** → all-time total loan disbursed

SELECT SUM(loan\_amount) as TOTAL\_FUNDED\_AMOUNT FROM bank\_loan\_data;



* + **MTD Funded Amount** → loan disbursed in the current month up to today

SELECT SUM(loan\_amount) as MTD\_FUNDED\_AMOUNT FROM bank\_loan\_data

WHERE MONTH(issue\_date) = 12 AND YEAR(issue\_date) = 2021 ;



* + **MoM Change** → compare funded amount with the previous month.

SELECT SUM(loan\_amount) as PMTD\_FUNDED\_AMOUNT FROM bank\_loan\_data

WHERE MONTH(issue\_date) = 11 AND YEAR(issue\_date) = 2021 ;



👉 This shows how much lending the bank is actually doing.

**3. Total Amount Received**

* Meaning: The total repayment received from borrowers.
* What to track:
  + **Overall Amount Received** → all-time repayments

SELECT SUM(total\_payment) as TOTAL\_PAYMENT\_RECEIVED FROM bank\_loan\_data ;



* + **MTD Amount Received** → repayments in the current month

SELECT SUM(total\_payment) as MTD\_PAYMENT\_RECEIVED FROM bank\_loan\_data

WHERE MONTH(issue\_date) = 12 AND YEAR(issue\_date) = 2021 ;



* + **MoM Change** → compare repayments with the previous month.

SELECT SUM(total\_payment) as PMTD\_PAYMENT\_RECEIVED FROM bank\_loan\_data

WHERE MONTH(issue\_date) = 11 AND YEAR(issue\_date) = 2021 ;



👉 This tells us about **cash inflow** and loan repayment performance.

**4. Average Interest Rate**

* Meaning: The average interest rate applied across all loans.
* What to track:
  + **Overall Average Interest Rate**

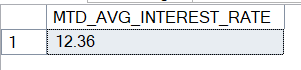
SELECT ROUND(AVG(int\_rate),4)\*100 as AVG\_INTEREST\_RATE FROM bank\_loan\_data ;



* + **MTD Average Interest Rate**

SELECT ROUND(AVG(int\_rate),4)\*100 as MTD\_AVG\_INTEREST\_RATE FROM bank\_loan\_data

WHERE MONTH(issue\_date) = 12 AND YEAR(issue\_date) = 2021 ;



* + **MoM Change** → how interest rates are changing from month to month.

SELECT ROUND(AVG(int\_rate),4)\*100 as PMTD\_AVG\_INTEREST\_RATE FROM

bank\_loan\_data

WHERE MONTH(issue\_date) = 11 AND YEAR(issue\_date) = 2021 ;



👉 This indicates the **cost of borrowing** for customers and profitability for the bank.

**5. Average Debt-to-Income Ratio (DTI)**

* Meaning: Average ratio of borrowers’ monthly debt to their monthly income.
* What to track:

**📝 Note on DTI (Debt-to-Income Ratio)**

**Definition:**  
The Debt-to-Income Ratio (DTI) measures how much of a borrower’s monthly income goes towards debt payments.

**Formula:**

DTI= (TOTAL DEBT / MONTHLY INCOME) \* 100

**Example:**

* Monthly Income = $5,000
* Monthly Debt Payments = $1,500
* DTI = (1,500 ÷ 5,000) × 100 = **30%**

**Interpretation:**

* **Low DTI (<35%)** → Borrower is financially healthy and has low repayment risk.
* **High DTI (>50%)** → Borrower is risky because a large portion of income is already committed to debts.

**Importance in Banking:**

* Banks use DTI to evaluate a borrower’s financial health.
* A lower DTI indicates a safer loan, while a higher DTI signals higher risk of default.

**In this Project (KPIs to track):**

1. **Overall Average DTI** – average across all borrowers
2. **MTD Average DTI** – average for current/latest month
3. **MoM Change in DTI** – month-over-month comparison
   * **Overall Average DTI**

SELECT ROUND(AVG(dti),4)\*100 as Debt\_to\_Income\_Ratio FROM bank\_loan\_data ;



* + **MTD Average DTI**

SELECT ROUND(AVG(dti),4)\*100 as MTD\_Debt\_to\_Income\_Ratio FROM

bank\_loan\_data

WHERE MONTH(issue\_date) = 12 AND YEAR(issue\_date) = 2021 ;



* + **MoM Change** in DTI

SELECT ROUND(AVG(dti),4)\*100 as PMTD\_Debt\_to\_Income\_Ratio FROM

bank\_loan\_data

WHERE MONTH(issue\_date) = 11 AND YEAR(issue\_date) = 2021 ;



👉 This shows the **financial health** of borrowers and helps assess risk.

**GOOD LOAN ISSUED**

**📌 Project Requirements**

**1. Good Loan vs Bad Loan KPIs**

**✅ Good Loan**

A loan is classified as a **Good Loan** if its status is either **“Fully Paid”** or **“Current”**.  
The following KPIs need to be calculated:

* **Good Loan Application Percentage** → Percentage of total applications that are good loans.

select

COUNT(CASE

WHEN loan\_status = 'Fully Paid' OR loan\_status = 'Current' THEN id

END ) \*100 /

COUNT(id) AS Good\_Loan\_Percentage from bank\_loan\_data ;



* **Good Loan Applications** → Total count of applications with a good loan status.

SELECT COUNT(id) as Total\_Good\_loan\_Status from bank\_loan\_data

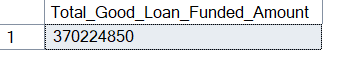
WHERE loan\_status = 'Fully Paid' OR loan\_status = 'Current';



* **Good Loan Funded Amount** → Total funded amount disbursed for good loans.

SELECT SUM(loan\_amount) as Total\_Good\_Loan\_Funded\_Amount from bank\_loan\_data

WHERE loan\_status = 'Fully Paid' OR loan\_status = 'Current';



* **Good Loan Total Received Amount** → Total repayment amount received from borrowers for good loans.

SELECT SUM(total\_payment) as Good\_Loan\_Total\_Received\_Amount from bank\_loan\_data

WHERE loan\_status = 'Fully Paid' OR loan\_status = 'Current';



* **❌ Bad Loan**

A loan is classified as a **Bad Loan** if its status is either **“Charged Off”** or **“Default”**.  
The following KPIs need to be calculated:

* **Bad Loan Application Percentage** → Percentage of total applications that are bad loans.

SELECT

(COUNT(CASE

WHEN loan\_status = 'Charged Off' THEN id

END ) \* 100 / COUNT(id)) as Bad\_LoanApplication\_Percentage

from bank\_loan\_data ;



* **Bad Loan Applications** → Total count of applications with a bad loan status.

SELECT COUNT(id) as Total\_Bad\_Loan\_Applications from bank\_loan\_data

WHERE loan\_status = 'Charged Off';



* **Bad Loan Funded Amount** → Total funded amount disbursed for bad loans.

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

WHERE loan\_status = 'Charged Off';



* **Bad Loan Total Received Amount** → Total repayment amount received from borrowers for bad loans.

SELECT SUM(total\_payment) as Total\_Bad\_Loan\_Total\_Received\_Amount from bank\_loan\_data

WHERE loan\_status = 'Charged Off';

**2. Loan Status Grid View**

To gain a comprehensive overview of loan performance, a **grid view report** should be created, categorized by **Loan Status**.  
The grid must display the following metrics for each loan status category:

* **Total Loan Applications** → Number of applications per loan status.
* **Total Funded Amount** → Sum of the total funded amount per loan status.
* **Total Amount Received** → Sum of the repayment amount received per loan status.
* **Month-to-Date (MTD) Funded Amount** → Funded amount disbursed in the current month per loan status.
* **MTD Amount Received** → Amount received from borrowers in the current month per loan status.
* **Average Interest Rate** → Average interest rate per loan status.
* **Average Debt-to-Income Ratio (DTI)** → Average DTI value per loan status.

SELECT loan\_status , COUNT(id) as Total\_loan\_application,

SUM(loan\_amount) as Total\_funded\_amount ,

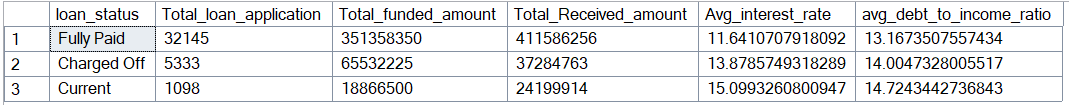
SUM(total\_payment) as Total\_Received\_amount ,

AVG(int\_rate\*100) as Avg\_interest\_rate ,

Avg(dti\*100) as avg\_debt\_to\_income\_ratio

from bank\_loan\_data

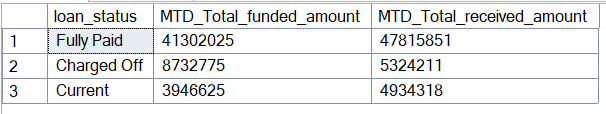
group by loan\_status ;



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 GROUP BY loan\_status ;



This grid view will provide a detailed breakdown of the lending portfolio, enabling stakeholders to:

* Monitor loan performance effectively.
* Identify healthy vs risky loans.
* Track month-to-date progress.
* Make informed, data-driven decisions.

**📌 Requirements for SQL Queries**

**1️⃣ Monthly Trends by Issue Date (Line Chart)**

* **Purpose:** Identify seasonality and long-term lending trends.
* **Requirements:**
  + Group data by **Month (Issue\_Date)**
  + Calculate:
    - COUNT(id) → Total Loan Applications
    - SUM(funded\_amount) → Total Funded Amount
    - SUM(amount\_received) → Total Amount Received

SELECT MONTH(issue\_date) as Month\_Number ,

DATENAME(MONTH , issue\_date) as Month\_Name ,

COUNT(id) as Total\_Loan\_Application ,

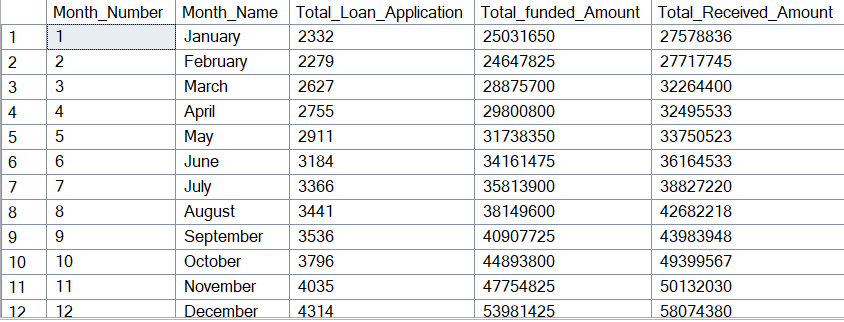
SUM(loan\_amount) as Total\_funded\_Amount ,

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

FROM bank\_loan\_data

GROUP BY MONTH(issue\_date) ,

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



**2️⃣ Regional Analysis by State (Filled Map)**

* **Purpose:** Compare lending activity across regions.
* **Requirements:**
  + Group data by **State**

Calculate the same 3 metrics (Applications, Funded Amount, Amount Received)

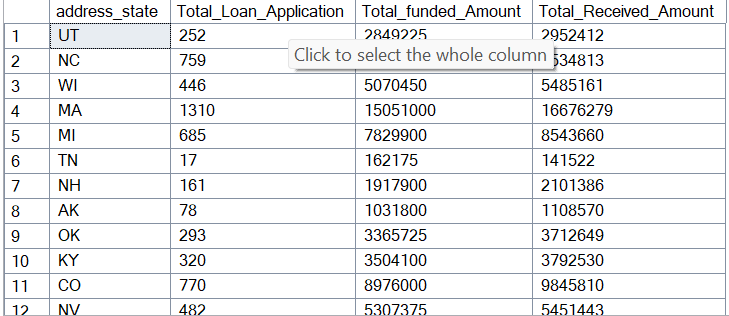
SELECT address\_state , COUNT(id) AS Total\_Loan\_Application ,

SUM(loan\_amount) as Total\_funded\_Amount ,

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

FROM bank\_loan\_data

GROUP BY address\_state ;



**3️⃣ Loan Term Analysis (Donut Chart)**

* **Purpose:** Show distribution of loans by term length.
* **Requirements:**
  + Group data by **Loan\_Term**
  + Calculate: Applications, Funded Amount, Amount Received

SELECT term , COUNT(id) AS Total\_Loan\_Application ,

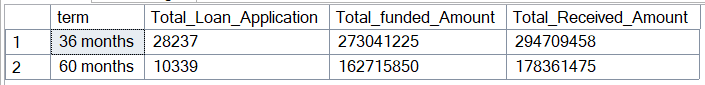
SUM(loan\_amount) as Total\_funded\_Amount ,

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

FROM bank\_loan\_data

GROUP BY term

ORDER BY term ;



**4️⃣ Employment Length Analysis (Bar Chart)**

* **Purpose:** See how employment history impacts lending.
* **Requirements:**
  + Group data by **Employment\_Length**
  + Calculate: Applications, Funded Amount, Amount Received

SELECT emp\_length , COUNT(id) AS Total\_Loan\_Application ,

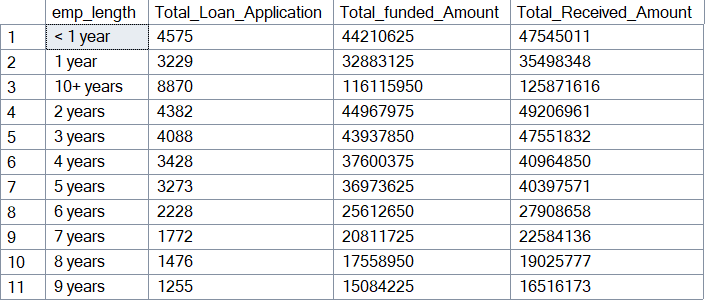
SUM(loan\_amount) as Total\_funded\_Amount ,

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

FROM bank\_loan\_data

GROUP BY emp\_length

ORDER BY emp\_length ;



**5️⃣ Loan Purpose Breakdown (Bar Chart)**

* **Purpose:** Show why borrowers take loans.
* **Requirements:**
  + Group data by **Loan\_Purpose**
  + Calculate: Applications, Funded Amount, Amount Received

SELECT purpose ,

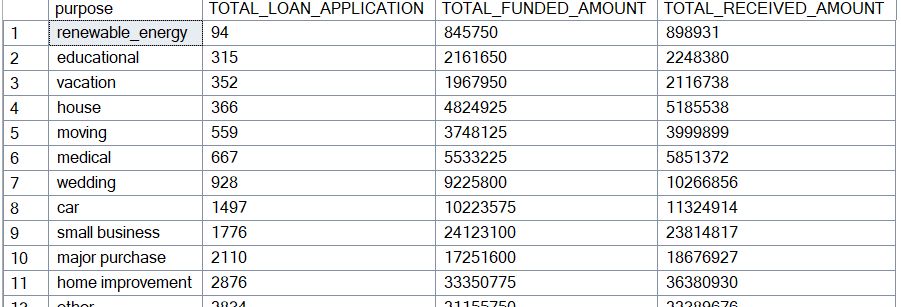
COUNT(id) AS TOTAL\_LOAN\_APPLICATION ,

SUM(loan\_amount) AS TOTAL\_FUNDED\_AMOUNT ,

SUM(total\_payment) AS TOTAL\_RECEIVED\_AMOUNT

FROM bank\_loan\_data

GROUP BY purpose ORDER BY COUNT(id) ;



**6️⃣ Home Ownership Analysis (Tree Map)**

* **Purpose:** Show loan distribution by home ownership status.
* **Requirements:**
  + Group data by **Home\_Ownership**
  + Calculate: Applications, Funded Amount, Amount Received

SELECT home\_ownership , COUNT(id) AS TOTAL\_LOAN\_APPLICATION ,

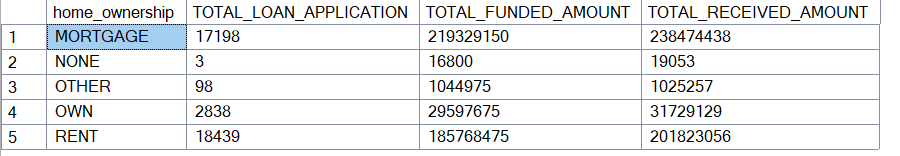
SUM(loan\_amount) AS TOTAL\_FUNDED\_AMOUNT ,

SUM(total\_payment) AS TOTAL\_RECEIVED\_AMOUNT

FROM bank\_loan\_data

GROUP BY home\_ownership

ORDER BY home\_ownership ;



**PROJECT EXPLANATION**

**Project Explanation Framework**

**1. Introduction: The Hook & Business Problem (1-2 Sentences)**

* **Goal:** Briefly state what the project is and why it's important to a bank.
* **Suggested Script:** "My project is a **Bank Loan Analysis Dashboard** created to help a bank's lending department efficiently track its portfolio health. The core business problem was to transform raw transactional data into **actionable insights** for **credit risk management** and **profitability analysis**."

**2. Technology Stack & Role (1 Sentence)**

* **Goal:** Mention the tools and your specific contribution.
* **Suggested Script:** "I used an end-to-end data stack: **SQL Server** for all data transformation and querying, and **Power BI** for visualization and dynamic reporting."

**3. Data Processing (SQL Server) (Focus on "What I Did")**

* **Goal:** Explain the data preparation and how you used SQL to solve business problems.
* Suggested Script:

"The initial step involved loading the raw loan data (financial\_loan.csv) into SQL Server. My main tasks in SQL were:

* + **Data Transformation:** I ensured data cleanliness, handled date formats, and made sure all key fields (like issue\_date and loan\_status) were ready for analysis.
  + **Metric Pre-calculation:** I wrote optimized SQL queries to pre-calculate all the core metrics and necessary aggregations. For example, I used SUM() and COUNT() functions, coupled with date logic (like MONTH(issue\_date)) to calculate **Month-to-Date (MTD)** and **Prior Month-to-Date (PMTD)** KPIs directly in SQL. This ensured high performance in Power BI.
  + **Segmentation:** I prepared segmented data by using GROUP BY clauses across critical dimensions like **Loan Grade**, **Loan Term**, **Purpose**, and **Address State**, which formed the basis of our detailed reports."

**4. Visualization & Reporting (Power BI) (Focus on "What the Report Shows")**

* **Goal:** Describe the final product—the dashboards—and the key metrics.
* Suggested Script:

"The output is a comprehensive Power BI solution consisting of three interactive dashboards:

* 1. **Summary Dashboard:** This is the high-level overview. It focuses on the main **Key Performance Indicators (KPIs)**, such as **Total Loan Applications**, **Total Funded Amount**, **Total Amount Received**, and **Average Interest Rate**. Crucially, it provides MTD vs. PMTD comparisons to track current performance against the previous month.
  2. **Loan Status Dashboard:** This is dedicated to **Credit Risk**. It breaks down all the core metrics by the loan\_status (e.g., 'Fully Paid,' 'Charged Off,' 'Current'). This allows us to instantly see the scale of our performing vs. non-performing loans.
  3. **Details Dashboard:** This offers deeper, segmented analysis. I created visuals to show the distribution of loans by **Risk Grade (A-G)**, **Purpose (e.g., Debt Consolidation)**, **Term (36 vs. 60 months)**, and **Geographical State**.
  4. **DAX/Time Intelligence:** I implemented DAX formulas for complex **Time Intelligence** calculations and dynamic filtering, ensuring the user could slice the data instantly by date or any other variable."

**5. Key Insight/Conclusion (The "So What?")**

* **Goal:** Provide one example of a key insight a manager could take away.
* **Suggested Script:** "The main value of this project is providing a single source of truth for portfolio performance. For example, a key takeaway is the ability to instantly identify which **Loan Grades** or **States** contribute the most to the **'Charged Off'** (defaulted) status. This information is vital for the bank to adjust its lending policies and pricing to better manage risk and improve profitability."

**Full Answer in Interview Format**

"My project was a **Bank Loan Analysis Dashboard** designed to provide actionable business intelligence on a bank's loan portfolio, with the primary goal of improving **credit risk management** and understanding **portfolio profitability**.

**The stack I used was SQL Server for data processing and Power BI for visualization.**

First, I imported the raw loan data into **SQL Server**. My focus here was on data integrity and transformation. I wrote optimized SQL queries to pre-calculate all the necessary metrics. This included complex calculations like **Month-to-Date** and **Prior Month-to-Date** figures for KPIs like **Total Funded Amount** and **Total Applications**. I also aggregated the data by all key dimensions—such as **Loan Grade, Term, and Purpose**—to ensure the report had the detailed breakdowns the business required.

In **Power BI**, I built an interactive reporting suite with three main dashboards:

1. A **Summary Dashboard** for C-level tracking of core KPIs and time-based comparisons.
2. A **Loan Status Dashboard** that breaks down metrics specifically by the default status, which is critical for risk assessment.
3. A **Details Dashboard** that allows analysts to dive deep into loan distribution across risk grades, geography, and borrower demographics.

Ultimately, this project delivers immediate value by allowing the bank to **instantly identify high-risk segments**—for instance, spotting if a certain loan purpose or risk grade has an unusually high default rate—so they can make data-driven decisions on lending strategy and pricing."