```sql
--Create a new database
create database [Bank Loan DB];

--Switch to the new database
use [Bank Loan DB];

--Retrieve Records
select * from bank_loan_data;

--Data Cleaning
/*Delete duplicate rows*/
delete from bank_loan_data
where id NOT IN (select MIN(id) from bank_loan_data
group by address_state, application_type, emp_length, grade, home_ownership,
  issue_date,
last_credit_pull_date, last_payment_date, loan_status, next_payment_date,
  member_id, purpose, sub_grade,
term, annual_income, dti, installment, int_rate, loan_amount,
  total_acc,total_payment
);

/*checking for null values in columns*/
select * from bank_loan_data
where id IS NULL
   OR address_state IS NULL
   OR emp_length IS NULL
   OR grade IS NULL
   OR home_ownership IS NULL
   OR issue_date IS NULL
   OR last_credit_pull_date IS NULL
   OR last_payment_date IS NULL
   OR loan_status IS NULL
   OR next_payment_date IS NULL
   OR member_id IS NULL
   OR purpose IS NULL
   OR sub_grade IS NULL
   OR term IS NULL
   OR annual_income IS NULL
   OR dti IS NULL
   OR installment IS NULL
   OR int_rate IS NULL
   OR loan_amount IS NULL
   OR total_acc IS NULL
   OR total_payment IS NULL;

/*Creating a new column*/
alter table bank_loan_data
add [Good Vs Bad Loan] varchar(50);

```

```sql
/*Adding data in new column [Good Vs Bad Loan] using condition*/
update bank_loan_data
set [Good Vs Bad Loan] =
(CASE WHEN loan_status='Current' OR loan_status='Fully Paid' THEN 'Good Loan'
ELSE 'Bad Loan'
END);

--Writing queries for Problem Statement questions
--i)KPI Requirements
--1) Total Loan Applications:
select COUNT(id) AS [Total Loan Applications] from bank_loan_data;

--MTD (Month-to-Date) 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;

--PMTD (Previous Month-to-Date) 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;

--2) Total Funded Amount:
select SUM(loan_amount) AS [Total Funded Amount] from bank_loan_data;

--MTD (Month-to-Date) 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;

--PMTD (Previous Month-to-Date) 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;

--3) Total Amount Received:
select SUM(total_payment) AS [Total Amount Received] from bank_loan_data;

--MTD (Month-to-Date) 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;

--PMTD (Previous Month-to-Date)  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;

--4) Average Interest Rate:
select ROUND(AVG(int_rate),4)*100 AS [Average Interest Rate] from
  bank_loan_data;

--MTD (Month-to-Date) Average Interest Rate:
select ROUND(AVG(int_rate),4)*100 AS [MTD Average Interest Rate] from
  bank_loan_data
```

```sql
where MONTH(issue_date)=12 AND YEAR(issue_date)=2021;

--PMTD (Previous Month-to-Date) 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;

--5) Average Debt-to-Income Ratio (DTI):
select ROUND(AVG(dti),4)*100 AS [Average Debt-to-Income Ratio] from
  bank_loan_data;

--MTD (Month-to-Date) Average Debt-to-Income Ratio (DTI):
select ROUND(AVG(dti),4)*100 AS [MTD Average Debt-to-Income Ratio] from
  bank_loan_data
where MONTH(issue_date)=12 AND YEAR(issue_date)=2021;

--PMTD (Previous Month-to-Date) Average Debt-to-Income Ratio (DTI):
select ROUND(AVG(dti),4)*100 AS [PMTD Average Debt-to-Income Ratio] from
  bank_loan_data
where MONTH(issue_date)=11 AND YEAR(issue_date)=2021;

--ii) Good Loan KPI:
--1.Good Loan Application Percentage:
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;

--2.Good Loan Applications:
select COUNT(id) AS [Good Loan Applications] from bank_loan_data
where loan_status IN ('Fully Paid','Current');

--3.Good Loan Funded Amount:
select SUM(loan_amount) AS [Good Loan Funded Amount] from bank_loan_data
where loan_status IN ('Fully Paid','Current');

--4.Good Loan Amount Received
select SUM(total_payment) AS [Good Loan Amount Received] from bank_loan_data
where loan_status IN ('Fully Paid','Current');

--iii) Bad Loan KPI:
--1.Bad Loan Application Percentage:
select (COUNT(
CASE
    WHEN loan_status='Charged Off' THEN id
    END) *100)/COUNT(id) AS [Bad Loan Percentage] from bank_loan_data;
```

```sql
--2.Bad Loan Applications:
select COUNT(id) AS [Bad Loan Applications] from bank_loan_data
where loan_status='Charged Off';

--3.Bad Loan Funded Amount:
select SUM(loan_amount) AS [Bad Loan Funded Amount] from bank_loan_data
where loan_status='Charged Off';

--4.Bad Loan Total Received Amount:
select SUM(total_payment) AS [Bad Loan Amount Received] from bank_loan_data
where loan_status='Charged Off';

--Loan Status
select loan_status, COUNT(id) AS [Total Loan Applications],
SUM(loan_amount) AS [Total Funded Amount],
SUM(total_payment) AS [Total Amount Received],
AVG(int_rate*100) AS [Average Interest Rate],
AVG(dti*100) AS [Average Debt-to-Income Ratio (DTI)] from bank_loan_data
group by loan_status;

--1. Total Loan Applications by Issue Date:
select MONTH(issue_date) AS [Month Number],DATENAME(MONTH, issue_date) AS
  [Month Name],
COUNT(id) AS [Total Loan Applications] from bank_loan_data
group by MONTH(issue_date),DATENAME(MONTH, issue_date)
order by [Month Number];

--2.Total Loan Applications by State:
select address_state AS State,COUNT(id) AS [Total Loan Applications] from
  bank_loan_data
group by address_state
order by [Total Loan Applications] DESC;

--3.Total Loan Applications by Loan Term
select term AS [Loan Term],COUNT(id) AS [Total Loan Applications] from
  bank_loan_data
group by term
order by [Total Loan Applications] DESC;

--4.Total Loan Applications by Employee Length
select emp_length AS [Employee Length],COUNT(id) AS [Total Loan Applications]
  from bank_loan_data
group by emp_length
order by [Total Loan Applications] DESC;

--5.Total Loan Applications by Loan Purpose
select purpose AS [Loan Purpose],COUNT(id) AS [Total Loan Applications] from
  bank_loan_data
group by purpose
```

```sql
order by [Total Loan Applications] DESC;

--6. Total Loan Applications by Home Ownership
select home_ownership AS [Home Ownership],COUNT(id) AS [Total Loan
  Applications] from bank_loan_data
group by home_ownership
order by [Total Loan Applications] DESC;

--Details Dashboard
select
    id AS ID,
    address_state AS [Address State],
    application_type AS [Application Type],
    emp_length AS [Employee Length],
    grade AS Grade,
    home_ownership AS [Home Ownership],
    issue_date AS [Issue Date],
    last_credit_pull_date AS [Last Credit Pull Date],
    last_payment_date AS [Last Payment Date],
    loan_status AS [Loan Status],
    next_payment_date AS [Next Payment Date],
    member_id AS [Member ID],
    purpose AS Purpose,
    sub_grade AS [Sub Grade],
    term AS Term,
    CONCAT('$', annual_income) AS [Annual Income],
    CONCAT(ROUND(dti * 100, 2), '%') AS DTI,
    CONCAT('$', installment) AS Installment,
    CONCAT(ROUND(int_rate * 100, 2), '%') AS [Interest Rate],
    CONCAT('$', loan_amount) AS [Loan Amount],
    total_acc AS [Total Accounts],
    CONCAT('$', total_payment) AS [Amount Received],
    [Good Vs Bad Loan]
from
    bank_loan_data;

```