

```
Select * from financial_loan;

-- Total Number of Applications --
select COUNT(id) as Total_Applications from financial_loan;

-- Total Number of Applications for Dec,2021 month --
select COUNT(id) as Month_to_Date_Applications from financial_loan
where MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021;

--Total Funded Amount --
select SUM(loan_amount) as Total_Funded_Amount from financial_loan;

--Total Funded Amount DEC 2021 --
select SUM(loan_amount) as MTD_Total_Funded_Amount from financial_loan
where MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021;

-- Total Amount Recieved --
select SUM(total_payment) as Total_Amount_Recieved from financial_loan;

-- Total Amount Recieved current month --
select SUM(total_payment) as Total_Amount_Recieved from financial_loan
where MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021;

-- Average Interest Rate ---
select Round(AVG(int_rate),4)*100 AS Avg_Interest_Rate from financial_loan;

-- Average Interest Rate for Current Month --
select Round(AVG(int_rate),4)*100 AS MTD_Avg_Interest_Rate from financial_loan
where MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021;

-- Average DTI (Debt to Income) Ratio --
select Round(AVG(dti),4)*100 AS Avg_DTI from financial_loan;

-- Average DTI (Debt to Income) Ratio MTD --
select Round(AVG(dti),4)*100 AS MTD_Avg_DTI from financial_loan
where MONTH(issue_date) = 12 AND YEAR(issue_date) = 2021;

-- Good Loan Percentage --
select COUNT(case when loan_status = 'Fully Paid' OR loan_status = 'Current' then id
END) * 100 / COUNT (id) as Good_Loan_Perc
from financial_loan;

-- Total Count of Good Loan Applications--
select COUNT(id) as Good_Loan_Application
from financial_loan
where loan_status = 'Fully Paid' OR loan_status = 'Current';

-- Good Loan Funded Amounts --
select SUM(total_payment) as Good_Loan_Fund
from financial_loan
```

```
where loan_status IN ('Fully Paid', 'Current');
```

```
-- Bad Loan Percentage --
```

```
select COUNT(case when loan_status = 'Charged Off' then id END)* 100 / COUNT (id) as      ↗  
    Bad_Loan_Perc  
from financial_loan;
```

```
-- Total Count of Bad Loan Applications--
```

```
select COUNT(id) as Bad_Loan_Application  
from financial_loan  
where loan_status IN ('Charged Off');
```

```
-- Bad Loan Funded Amount --
```

```
select SUM(loan_amount) as Bad_Loan_Amount  
from financial_loan  
where loan_status IN ('Charged Off');
```

```
-- Bad Loan Received Amount --
```

```
select SUM(total_payment) as Bad_Loan_Amount_Received  
from financial_loan  
where loan_status IN ('Charged Off');
```

```
--Total sum of the loan amount received by each state and Ranking them --
```

```
SELECT  
    address_state,  
    SUM(loan_amount) AS total_loan_amount,  
    RANK() OVER (ORDER BY SUM(loan_amount) DESC) AS state_rank  
FROM  
    financial_loan  
GROUP BY  
    address_state;
```

```
-- find the ID that has taken the highest loan for each purpose and rank them --
```

```
WITH RankedLoans AS (  
    SELECT  
        id,  
        purpose,  
        loan_amount,  
        ROW_NUMBER() OVER (PARTITION BY purpose ORDER BY loan_amount DESC) AS purpose_rank  
    FROM  
        financial_loan  
)  
  
SELECT  
    id,  
    purpose,  
    loan_amount,  
    purpose_rank
```

```
FROM
    RankedLoans
WHERE
    purpose_rank IN ('1','2');

-- find the ID that has taken the highest loan for home_ownership and rank them --

WITH RankedLoan AS (
SELECT id, home_ownership, loan_amount,
ROW_NUMBER() OVER(PARTITION BY home_ownership ORDER BY loan_amount DESC) AS
    Purpose_Rank
FROM financial_loan
)

SELECT id, home_ownership, loan_amount
FROM RankedLoan
where Purpose_Rank = '1';

--find the relationship between loan status and income--
SELECT
    loan_status,
    AVG(annual_income) AS average_income,
    MIN(annual_income) AS min_income,
    MAX(annual_income) AS max_income,
    COUNT(*) AS total_loans
FROM
    financial_loan
GROUP BY
    loan_status;

-- rank the emp_title based on the total sum of loans for each region
SELECT
    emp_title,
    address_state,
    SUM(loan_amount) AS total_loan_amount,
    RANK() OVER (PARTITION BY emp_title ORDER BY SUM(loan_amount) DESC) AS
        rank_in_region
FROM
    financial_loan
GROUP BY
    emp_title, address_state
ORDER BY
    address_state, rank_in_region;
```