**Task 2: Data Engineering on Reporting Platform**

**First notebook, I generated the dataset.**

Generates loan data over a three-year period (from three years ago up to April 22, 2024). It randomly creates loan records with details such as loan ID, borrower ID, loan amount, start date, maturity date, loan type, status (active or completed), investor count, profit percentage, repayment amount with interest, and risk rating.

The script iterates through each month within the date range, randomly determines the number of loans for each month, and populates loan details accordingly. Finally, it saves the generated loan data into a JSON file named "loans\_data.json" with a structured format using the `json.dump()` function.

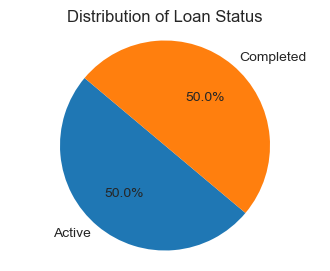
**Second notebook, I cleaned the dataset, analyzed it and got the required insights from it.**

Handling missing values, convert Data Types, remove duplicates, validate Data Integrity: Ensure 'start\_date' is before 'maturity\_date', detect and remove outliers. Then answering the required questions.

1. **Understanding the total active loans and completed loans.**

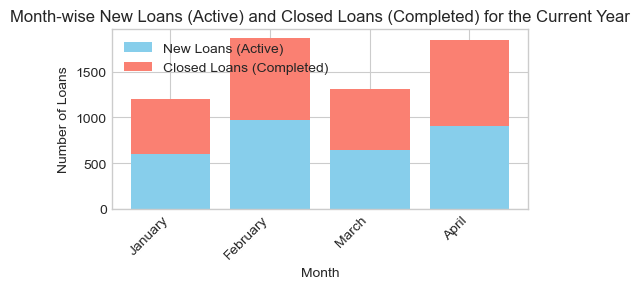
Total Active Loans: 27040

Total Completed Loans: 27037



### Identify how many new loans created and closed loans every month wise for current year.

1. Month-wise New Loans (Active) and Closed Loans (Completed) for the Current Year:
2. January: New Loans = 598, Closed Loans = 602
3. February: New Loans = 974, Closed Loans = 896
4. March: New Loans = 642, Closed Loans = 666
5. April: New Loans = 903, Closed Loans = 943
6. May: New Loans = 0, Closed Loans = 0
7. June: New Loans = 0, Closed Loans = 0
8. July: New Loans = 0, Closed Loans = 0
9. August: New Loans = 0, Closed Loans = 0
10. September: New Loans = 0, Closed Loans = 0
11. October: New Loans = 0, Closed Loans = 0
12. November: New Loans = 0, Closed Loans = 0
13. December: New Loans = 0, Closed Loans = 0



**3. Identify how much amount totally financed for the loans every month wise for current year.**

Month-wise Total Amount Financed for Loans (Current Year):

January: $32939253.00

February: $51002471.00

March: $36460677.00

April: $50312630.00

May: $0.00

June: $0.00

July: $0.00

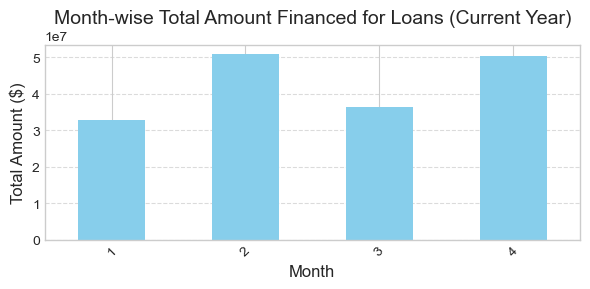
August: $0.00

September: $0.00

October: $0.00

November: $0.00

December: $0.00



### 4. How much interest generated month wise.

##### *for the whole dataset monthly wise not only for the current year*

year month

2021 4 5035717.27

5 3316719.23

6 2935280.48

7 4341485.66

8 3109659.65

9 4877065.25

10 3292213.92

11 4077234.52

12 2854879.03

2022 1 4039825.56

2 4128771.68

3 3127995.73

4 4110710.17

5 5406158.71

6 2957772.51

7 3798802.83

8 4057238.19

9 3985658.37

10 4867576.35

11 4869700.93

12 5000540.75

2023 1 3297826.12

2 5332072.81

3 3179685.59

4 3844681.11

5 2902342.07

6 3616180.53

7 5064990.59

8 3294883.94

9 5512426.09

10 3055112.03

11 4425303.10

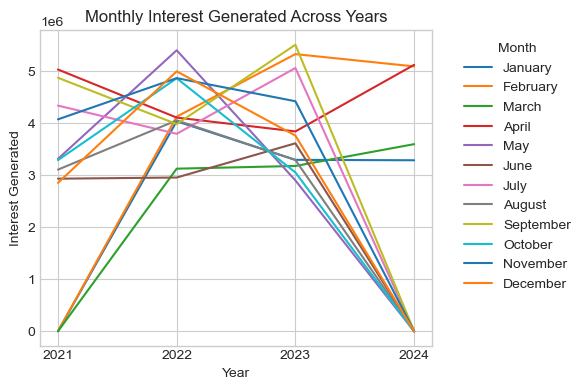
12 3764643.73

2024 1 3289665.82

2 5095506.96

3 3599791.28

4 5125662.09



### 5. Total outstanding amount with interest as of current date.

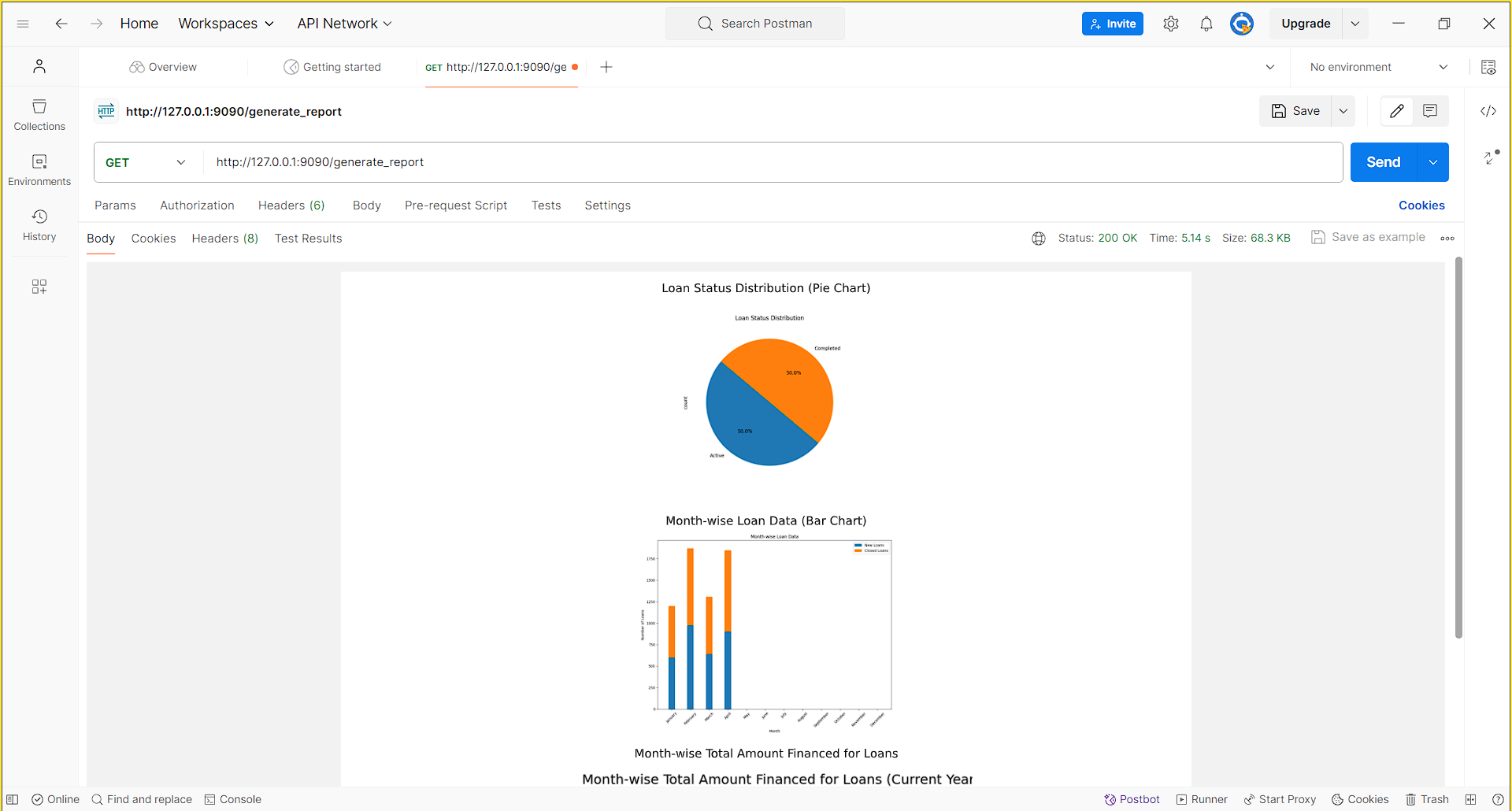
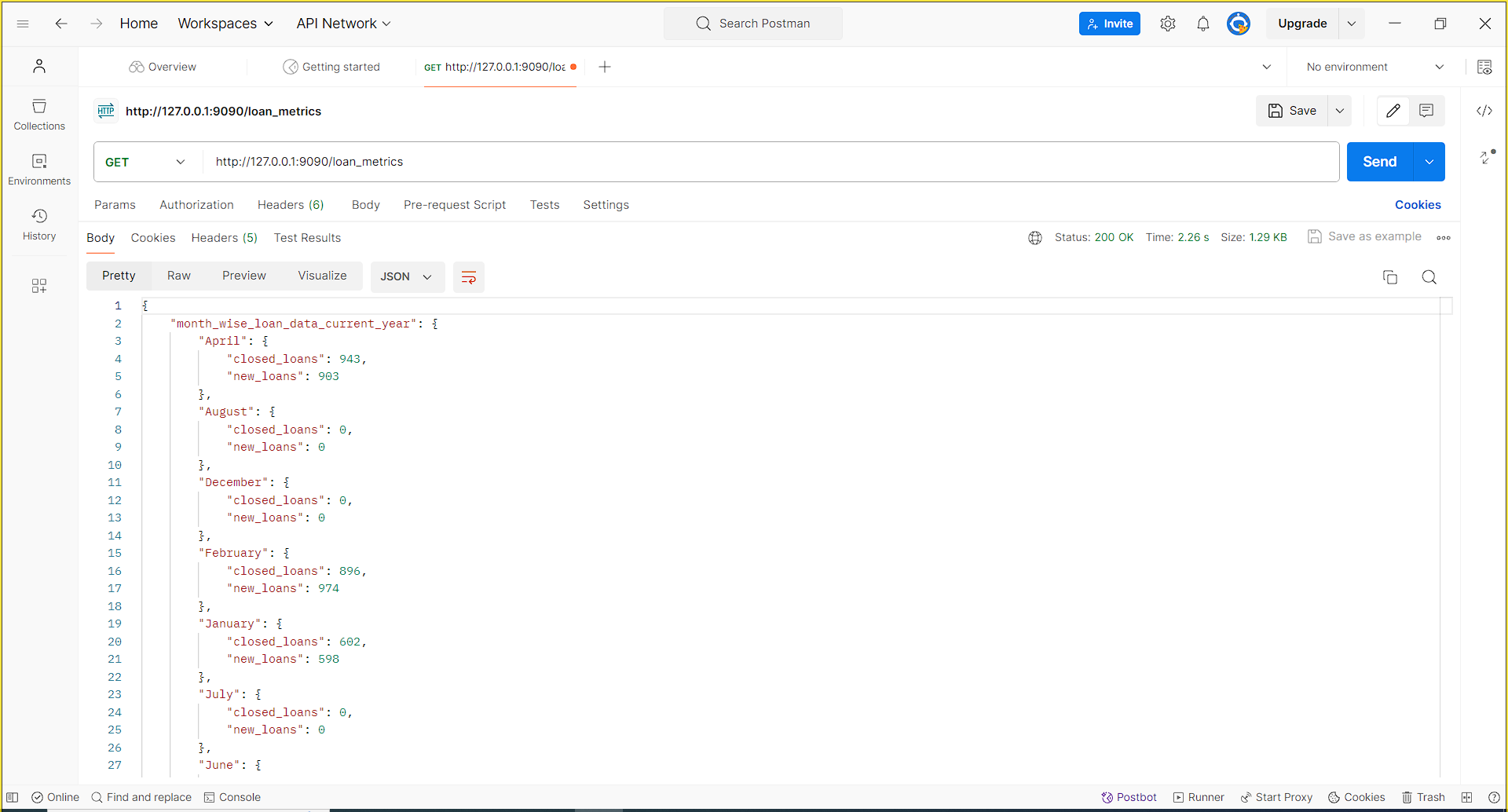
Total Outstanding Amount with Interest (Current Date): $765603535.67

After finishing the notebook, I started my Flask application which should generate a report about the insights in numbers and also a pdf version of the visualizations that we can download it from the browser or view it through postman.

The script called app.py

Run it through VS Code from the terminal using: python app.py (after navigating to the file location)

Then open postman and navigate to the http link and see the results.



I also provided a video that describe these steps.

**Thank you**