

```
In [2]: <import pandas as pd
import matplotlib.pyplot as plt
import plotly.express as px
import seaborn as sns

In [3]: df = pd.read_csv(r'C:\Users\User\Desktop\VP\Financial_loan.csv')

In [4]: df

Out[4]:
   id  address_state  application_type  emp_length  emp_title  grade  home_ownership  issue_date  last_credit_pull_date  last_payment_date  ...  sub_grade  term  verification_status  annual
0    1077430         GA      INDIVIDUAL    < 1 year      Ryder      C      RENT      11-02-2021      13-09-2021      13-04-2021  ...  C4      60 months      Source Verified
1    1072053         CA      INDIVIDUAL      9 years      MKC Accounting      E      RENT      01-01-2021      14-12-2021      15-01-2021  ...  E1      36 months      Source Verified
2    1069243         CA      INDIVIDUAL      4 years      Chemat Technology Inc      C      RENT      05-01-2021      12-12-2021      09-01-2021  ...  C5      36 months      Not Verified
3    1041756         TX      INDIVIDUAL    < 1 year      Barnes distribution      B      MORTGAGE  25-02-2021      12-12-2021      12-03-2021  ...  B2      60 months      Source Verified
4    1068350         IL      INDIVIDUAL    10+ years      J&J Steel Inc      A      MORTGAGE  01-01-2021      14-12-2021      15-01-2021  ...  A1      36 months      Verified
...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...  ...
38571  803452         NJ      INDIVIDUAL    < 1 year      Joseph M Sanzani Company      C      MORTGAGE  11-07-2021      16-05-2021      16-05-2021  ...  C1      60 months      Verified
38572  970377         NY      INDIVIDUAL      8 years      Swat Fame      C      RENT      11-10-2021      16-04-2021      16-05-2021  ...  C1      60 months      Verified
38573  875376         CA      INDIVIDUAL      5 years      Anahem Regional Medical Center      D      RENT      11-09-2021      16-05-2021      16-05-2021  ...  D5      60 months      Verified
38574  972997         NY      INDIVIDUAL      5 years      Brooklyn Radiology      D      RENT      11-10-2021      16-05-2021      16-05-2021  ...  D5      60 months      Verified
38575  682952         NY      INDIVIDUAL      4 years      Allen Edmonds      F      RENT      11-07-2021      16-05-2021      16-05-2021  ...  F3      60 months      Verified

38576 rows x 24 columns

In [5]: df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 38576 entries, 0 to 38575
Data columns (total 24 columns):
 #   Column              Non-Null Count  Dtype
---  -
 0   id                  38576 non-null  int64
 1   address_state       38576 non-null  object
 2   application_type     38576 non-null  object
 3   emp_length          38576 non-null  object
 4   emp_title           37138 non-null  object
 5   grade               38576 non-null  object
 6   home_ownership       38576 non-null  object
 7   issue_date          38576 non-null  object
 8   last_credit_pull_date 38576 non-null  object
 9   last_payment_date    38576 non-null  object
10  loan_status          38576 non-null  object
11  next_payment_date     38576 non-null  object
12  member_id            38576 non-null  int64
13  purpose              38576 non-null  object
14  sub_grade           38576 non-null  object
15  term                 38576 non-null  object
16  verification_status   38576 non-null  object
17  annual_income        38576 non-null  float64
18  dti                  38576 non-null  float64
19  installment          38576 non-null  float64
20  int_rate              38576 non-null  float64
21  loan_amount           38576 non-null  int64
22  total_acc             38576 non-null  int64
23  total_payment         38576 non-null  int64
dtypes: float64(4), int64(5), object(15)
memory usage: 7.1+ MB

In [6]: df.dtypes

Out[6]:
id                  int64
address_state       object
application_type     object
emp_length          object
emp_title           object
grade               object
home_ownership       object
issue_date          object
last_credit_pull_date  object
last_payment_date    object
loan_status          object
next_payment_date    object
member_id           int64
purpose              object
sub_grade           object
term                 object
verification_status   object
annual_income        float64
dti                  float64
installment          float64
int_rate             float64
loan_amount          int64
total_acc            int64
total_payment        int64
dtype: object

In [7]: df.columns

Index(['id', 'address_state', 'application_type', 'emp_length', 'emp_title',
      'grade', 'home_ownership', 'issue_date', 'last_credit_pull_date',
      'last_payment_date', 'loan_status', 'next_payment_date', 'member_id',
      'purpose', 'sub_grade', 'term', 'verification_status', 'annual_income',
      'dti', 'installment', 'int_rate', 'loan_amount', 'total_acc',
      'total_payment'],
      dtype='object')

In [8]: df.duplicated().any()

Out[8]: False

In [9]: df.shape

Out[9]: (38576, 24)

In [10]: df.isnull().sum()

Out[10]:
address_state      0
application_type    0
emp_length         0
emp_title         1438
grade              0
home_ownership     0
issue_date         0
last_credit_pull_date 0
last_payment_date  0
loan_status        0
next_payment_date  0
member_id         0
purpose            0
sub_grade         0
term              0
verification_status 0
annual_income     0
dti               0
installment       0
int_rate          0
loan_amount       0
total_acc         0
total_payment     0
dtype: int64

In [11]: df.nunique()

Out[11]:
id                  38576
address_state       50
application_type     1
emp_length         11
emp_title          28525
grade              7
home_ownership      5
issue_date         65
last_credit_pull_date 107
last_payment_date  102
loan_status         3
next_payment_date  102
member_id          38576
purpose            14
sub_grade         35
term              2
verification_status 3
annual_income     5996
dti              2663
installment       15132
int_rate          371
loan_amount       880
total_acc         82
total_payment     19525
dtype: int64

In [12]: #Total Loan Applications
df['id'].count()

Out[12]: 38576

In [13]: #Total Amount Of Funded Amount
df['loan_amount'].sum()

Out[13]: 435757075

In [14]: #Total repayment amount
df['total_payment'].sum()

Out[14]: 473070933

In [15]: #Average Interest Rate
100*df['int_rate'].mean()

Out[15]: 12.048831397760178

In [16]: #Average DTI rate
100*df['dti'].mean()

Out[16]: 13.32743311903776

In [17]: #Number of Good Loans
len(df[(df['loan_status']=='Current') | (df['loan_status']=='Fully Paid')])

Out[17]: 33243

In [18]: #Number of bad Loans applications
len(df[(df['loan_status']=='Charged Off')])

Out[18]: 5333

In [19]: #Percentage of Good loan applications
len(df[(df['loan_status']=='Fully Paid') | (df['loan_status']=='Current')])/len(df['id')*100

Out[19]: 86.17534218166736

In [20]: #Percentage of Bad Loan applications
len(df[(df['loan_status']=='Charged Off')])/len(df['id')*100

Out[20]: 13.824657818332643

In [21]: df1 = df.groupby('loan_status').agg({'id':'count'}).sort_values(by='id',ascending=False)

In [22]: df1.plot(x = 'loan_status', y = 'id',kind='pie', autopct='%1.2F%%', figsize=(10,6))
plt.legend().set_visible(False)



In [26]: df2 = df.groupby('loan_status').agg({'loan_amount':'sum'}).sort_values(by='loan_amount',ascending=False)

Out[26]:
      loan_status  loan_amount
Fully Paid      851358350
Charged Off     65532225
Current         18860500

In [25]: df2.plot(kind='bar', figsize=(10,6))
plt.xlabel('loan status')
plt.ylabel('Sum of loan Amount')
plt.title('Sum of loan Amount by loan status')
plt.show()



In [28]: df3 = df.groupby('loan_status').agg({'total_payment':'sum'}).sort_values(by='total_payment',ascending=False)

Out[28]:
      loan_status  total_payment
Fully Paid      411586256
Charged Off     37284763
Current         24199914

In [33]: df3.plot(kind='bar', figsize=(10,6))
plt.xlabel('loan status')
plt.ylabel('Sum of total payment')
plt.title('Sum of loan total payment by loan status')
plt.show()



In [46]: df4 = df.groupby('issue_date').agg({'id':'count','loan_amount':'sum','total_payment':'sum'}).sort_values(by='id',ascending=False)

Out[46]:
      issue_date  id  loan_amount  total_payment
11-12-2021     2188  30860375  33587713
11-11-2021     2120  27214075  29601942
11-10-2021     2025  26371175  29485936
11-09-2021     1980  24949725  26503436
11-08-2021     1873  22566150  25233153
...
12-12-2021      1    5000      963
17-07-2021      1    8000     5837
19-11-2021      1    6000     7218
22-07-2021      1    5400     6277
25-02-2021      1    4500     4911

65 rows x 3 columns

In [48]: df4.plot(figsize=(10,6))

<AxesSubplot: xlabel='issue_date'>



In [68]: df5 = df.groupby('home_ownership').agg({'id':'count'}).sort_values(by='id',ascending=False)

Out[68]:
home_ownership  id
0      RENT      18439
1  MORTGAGE     17198
2      OWN      2838
3      OTHER      98
4      NONE        3

In [71]: df5.plot(kind = 'bar',figsize=(10,6))
plt.show()



In [73]: df6 = df.groupby('emp_length').agg({'loan_amount':'sum'}).sort_values(by='loan_amount',ascending=False)

Out[73]:
      emp_length  loan_amount
10+ years      116115950
2 years        44967975
< 1 year      44210625
3 years        43937850
4 years        37600375
5 years        36973025
1 year         32883125
6 years        25612650
7 years        20811725
8 years        17558950
9 years        15084225

In [75]: df6.plot(kind='bar',figsize=(10,6))
plt.show()



In [76]: df7 = df.groupby('purpose').agg({'total_payment':'sum'}).sort_values(by='total_payment',ascending=False)

Out[76]:
      purpose  total_payment
Debt consolidation  253801871
credit card        65214084
home improvement   36380930
other              33289676
small business     23814817
major purchase     18670927
car                11324914
wedding           10260856
medical            5851372
house             5185538
moving            3990899
educational        2248380
vacation           2116738
renewable_energy  898931

In [77]: df7.plot(kind='bar',figsize=(10,6))
plt.show()



In [78]: df8 = df.groupby('term').agg({'loan_amount':'sum'}).sort_values(by='loan_amount',ascending=False)

Out[78]:
      term  loan_amount
36 months  273041225
60 months  162715850

In [82]: df8.plot(x='term',y='loan_amount',kind='pie', autopct='%1.2F%%', figsize=(10,6))
plt.legend().set_visible(False)


```