LENDING CLUB CASE STUDY

Presented by - Manoj Kumar.

Index

- Problem Statement.
- Data Understanding.
- Problem Solving Approach.
- Analysis
    - Univariate Analysis.
    - Bivariate Analysis.
- Conclusion & Recommendations.

PROBLEM STATEMENT

Problem Statement

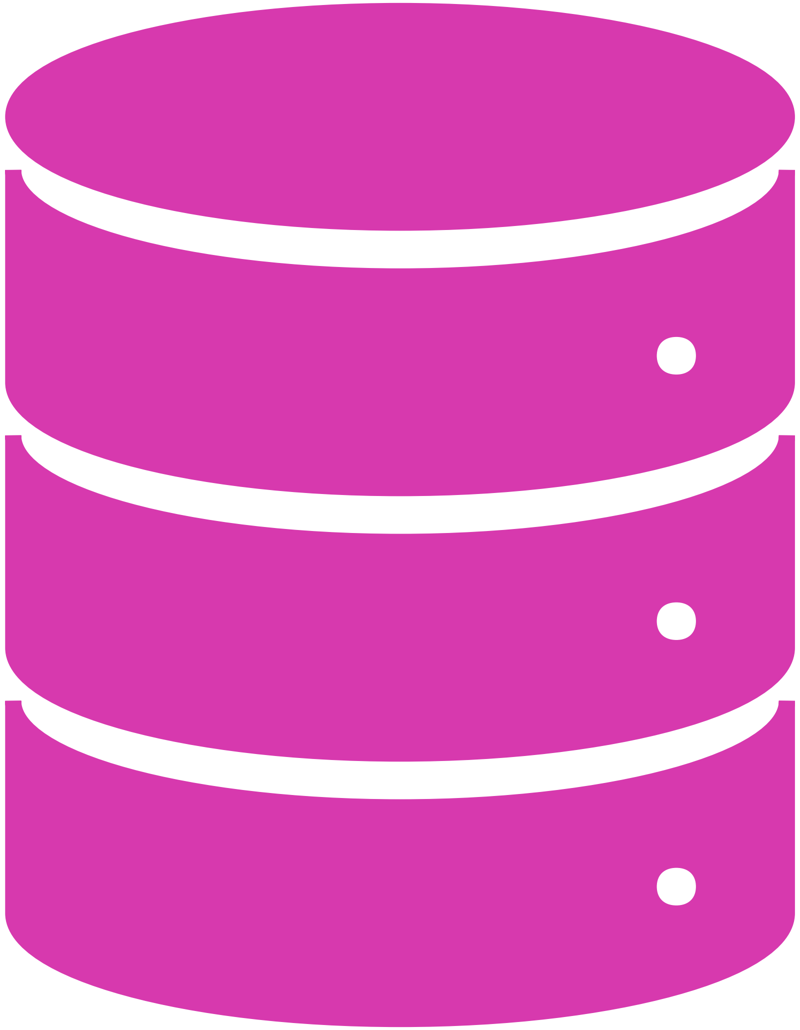
A consumer finance company which is specializes in lending various types of loans to urban customers. When the company receives a loan application, the company has to make a decision for loan approval based on the applicant's profile. Two **types of risks** are associated with the bank's decision:

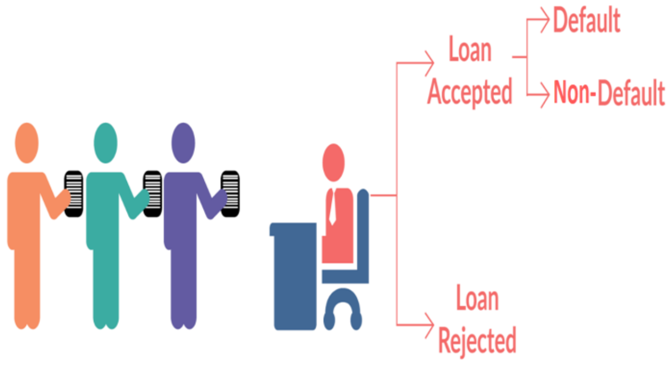
If the applicant is **likely to repay the loan**, then not approving the loan results in a **loss of business** to the company.	If the applicant is **not likely to repay the loan,** i.e., he/she is likely to default, then approving the loan may lead to a **financial loss** for the company.

The Lending club wanted to analyze the loan application data to derive meaningful insights which can help lenders to minimize the risk of losing business opportunity due to excessive rejection or financial loss due to not having repay of loan.

They also wanted to understand all possible driving factors behind the loan defaults, i.e., which features are strong indication of loan defaults. So that they can act and update rule engine to reject such application or reduce the loan amount or can also offer loan at higher interest rate.

DATA UNDERSTANDING





Data understanding

There are total 39717 records with 111 feature variables available in dataset.

The dataset contains data only for the accepted loan application from past.

There are 54 columns which have fill rate 0%, means null values in all rows hence we have removed these columns before performing analysis.

There are 3 columns (mths_since_last_deling, mths_since_last_record and next_payment_d) having 64, 92 and 97% missing values respectively, so dropped those columns.

Thera are 24 features variables, which we can say customer behavioral data and are not available during loan application time, so not suitable for our analysis, hence we can drop those as well before performing analysis.

Few features have unique values across all rows hence not suitable for analysis, so we can remove them before analysis.

There are 3 types of loan status we have in dataset. “Full Paid”, “Charged Off(defaulter)”, and “Current”.

There are 1140 rows with loan status = ‘current’ which can not be used to make any business decision as they are still ongoing. Hence, we can drop them from our further analysis.

Data understanding

Observation :

1. Below columns have still more than 60 percent of missing values so we can drop them safely.
   mths_since_last_delinq
   mths_since_last_record
   next_pymnt_d

In [11]:
```
cols = ['mths_since_last_delinq','mths_since_last_record','next_pymnt_d']
```

In [12]:
```
df.drop(cols,axis=1,inplace=True)
```

In [13]:
```
df.isnull().sum()
```

Out[13]:
```
id                          0
member_id                   0
loan_amnt                   0
funded_amnt                 0
funded_amnt_inv             0
term                        0
int_rate                    0
installment                 0
grade                       0
sub_grade                   0
emp_title                2459
emp_length               1075
home_ownership              0
annual_inc                  0
verification_status         0
issue_d                     0
loan_status                 0
pymnt_plan                  0
url                         0
desc                    12940
purpose                     0
title                      11
zip_code                    0
addr_state                  0
dti                         0
```

Observation:

The below columns seems customer behavioural data, which doesn't have any impact on loan approval decision making. so we can exclude them from our analysis.

In [14]:
```
### Drop customer behavioural data from the analysis.
cols_to_drop = ['url',
 'delinq_2yrs',
 'earliest_cr_line',
 'inq_last_6mths',
 'open_acc',
 'pub_rec',
 'revol_bal',
 'revol_util',
 'total_acc',
 'out_prncp',
 'out_prncp_inv',
 'total_pymnt',
 'total_pymnt_inv',
 'total_rec_prncp',
 'total_rec_int',
 'total_rec_late_fee',
 'recoveries',
 'collection_recovery_fee',
 'last_pymnt_d',
 'last_pymnt_amnt',
 'last_credit_pull_d',
 'collections_12_mths_ex_med',
 'acc_now_delinq',
 'desc'
 ]
```

In [15]:
```
df.drop(cols_to_drop,axis=1,inplace=True)
```

DATA UNDERSTANDING

In [16]:
```
df.isnull().mean().round(2)
```

Out[16]:
```
id                            0.00
member_id                     0.00
loan_amnt                     0.00
funded_amnt                   0.00
funded_amnt_inv               0.00
term                          0.00
int_rate                      0.00
installment                   0.00
grade                         0.00
sub_grade                     0.00
emp_title                     0.06
emp_length                    0.03
home_ownership                0.00
annual_inc                    0.00
verification_status           0.00
issue_d                       0.00
loan_status                   0.00
pymnt_plan                    0.00
purpose                       0.00
title                         0.00
zip_code                      0.00
addr_state                    0.00
dti                           0.00
initial_list_status           0.00
policy_code                   0.00
application_type              0.00
chargeoff_within_12_mths      0.00
delinq_amnt                   0.00
pub_rec_bankruptcies          0.02
tax_liens                     0.00
dtype: float64
```

Note: We have removed all non-relevant null value columns as a part of data clean-up activity

In [17]:
```
### size of dataframe after removing the null value columns
df.shape
```

Out[17]:
```
(39717, 30)
```

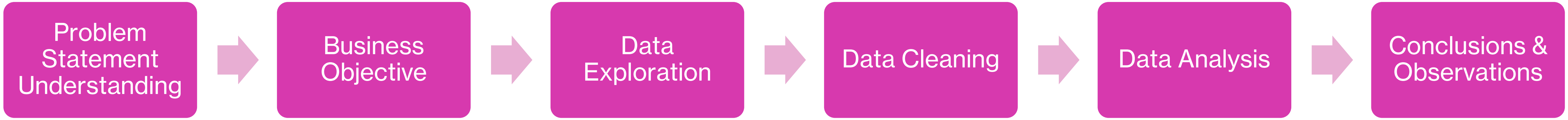
DATA UNDERSTANDING

Data Description:

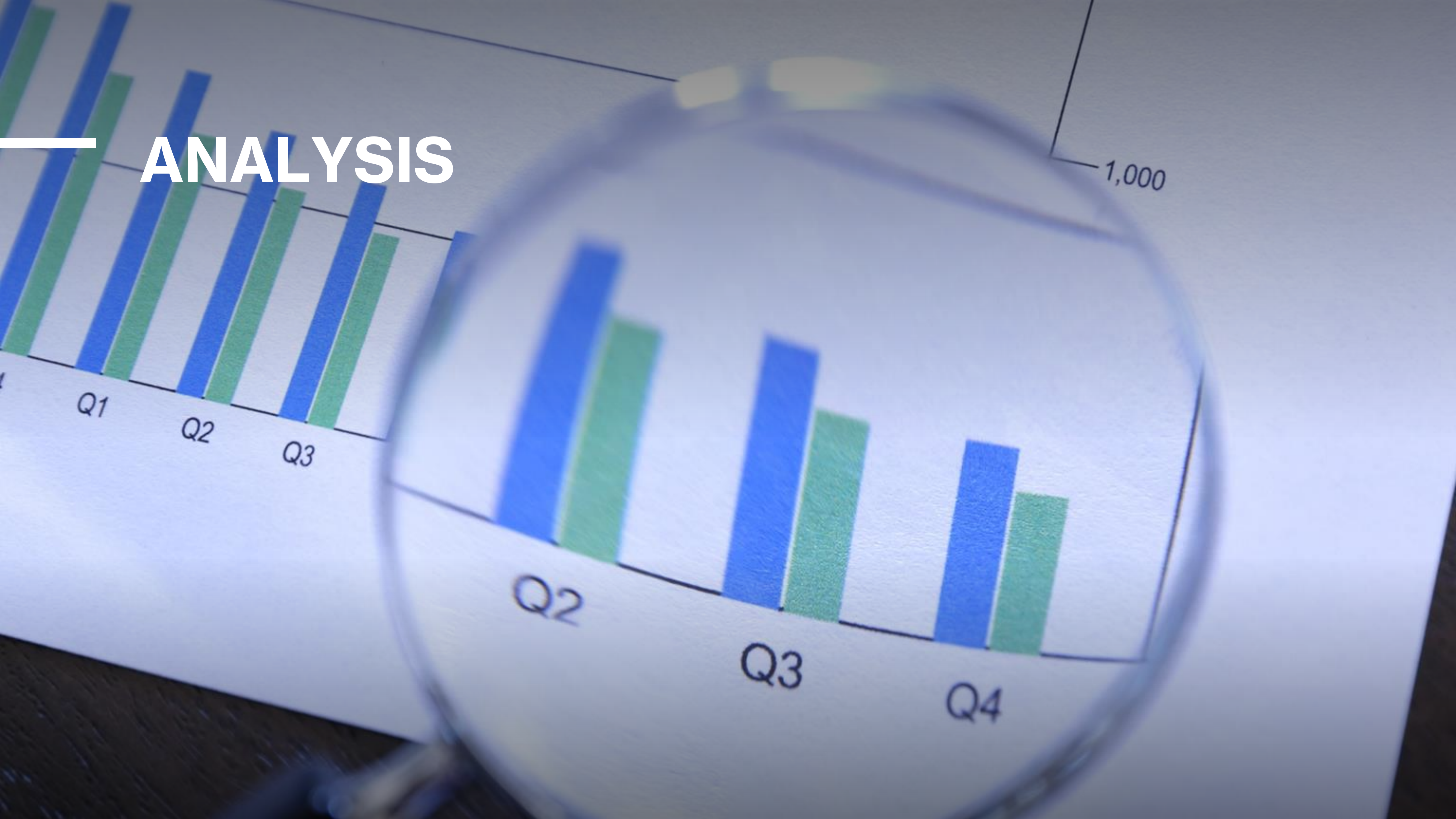
1. id, member-id : These are customer's id
2. loan_amnt : This is loan amount borrower has request for.
3. funded_amnt: This is loan amount the agency has approved.
4. funded_amnt_inv : this is the loan amount the invester/lender ha
5. term: this is the total tenure of installment.
6. int_rate : Interest rate on loan.
7. installment: this is installment amount.
8. grade : This is grade assigned by agency(lending club).
9. subgrade : This is subgrade assigned by agency(Lending club)
10. emp_title : Emp_title of the borrower.
11. emp_length : since when the borrower is employed.
12. home_ownership : whether the borrower is owner or tenent?
13. annual_inc: Annual income of borrower.
14. Verification_status : whether the source of income is verified or
15. Loan_status : this is our tager variable, which we will be analyzi
16. purpose : what is the purpose of loan request.
17. title : title of borrower
18. zip_code and addr_state : geo location of borrowers
19. A ratio calculated using the borrower's total monthly debt payme
    by the borrower's self-reported monthly income.

PROBLEM SOLVING APPROACH

PROBLEM SOLVING APPROACH



ANALYSIS

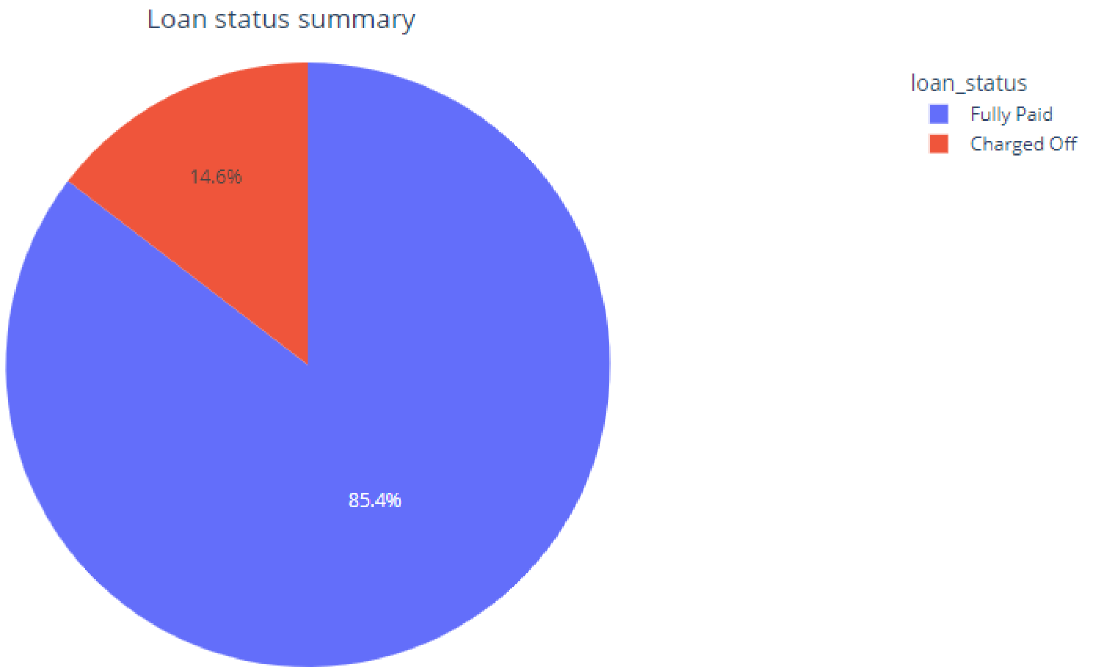


UNIVARIATE ANALYSIS

UNIVARIATE ANALYSIS

In [26]:

```
import plotly.express as px
fig = px.pie(df.loan_status.value_counts().to_frame(),values='loan_status',names=df.loan_status.value_counts().to_frame().ind
fig.update_layout(
    title={
        'text': "Loan status summary",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="loan_status")
fig.show()
```



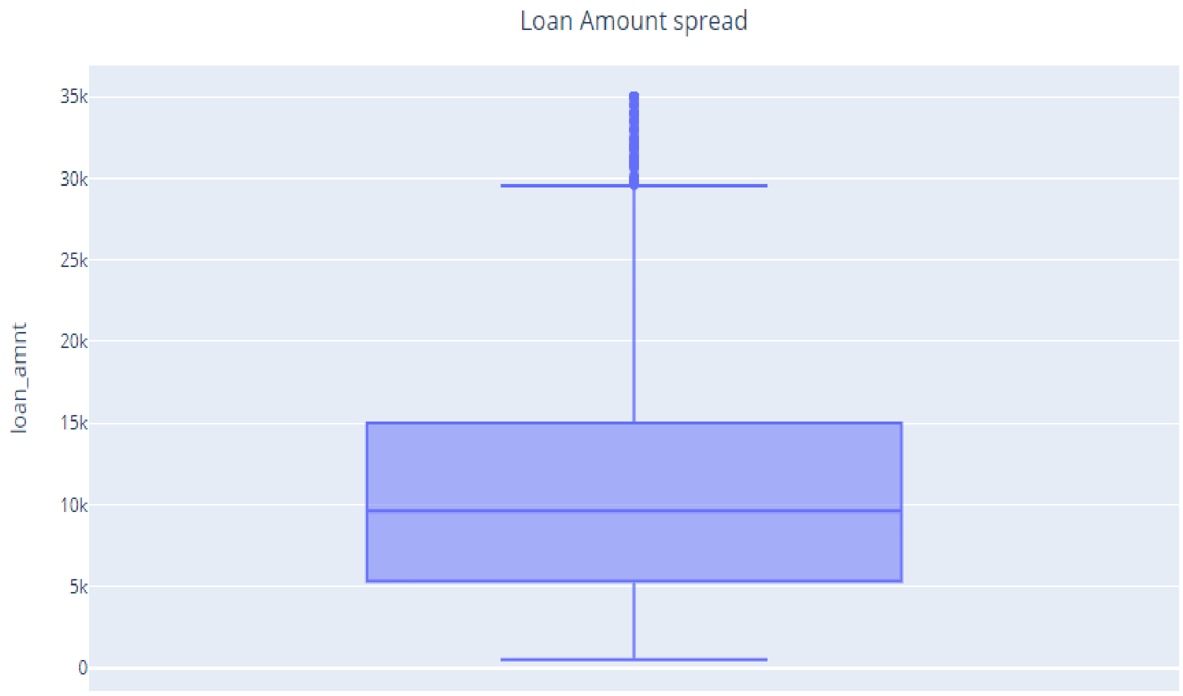
There are 85% loan with status 'fully paid'.

Remaining 14.6% are with status 'Charged off.

UNIVARIATE ANALYSIS

In [28]:

```
fig = px.box(df,y='loan_amnt')
fig.update_layout(
    title={
        'text': "Loan Amount spread",
        'y':0.95,
        'x':0.50,
        'xanchor': 'center',
        'yanchor': 'top'})

fig.show()
```



After analyzing distribution plot and box plot, we can say that the loan amount is not normally distributed. majority of the population falls between 5k - 15k.

The first quantile(q1): 5.3k, second quantile(q2) or median is: 9.6K and the third quantile is : 15k

The lower whisker is -8750 and the upper whisker is 29550.00

There are 1088 loan application for which the loan amount is greater than upper whisker. though they are greater than upper whisker, but we should avoid telling them outliers, they might be legitimate applications.

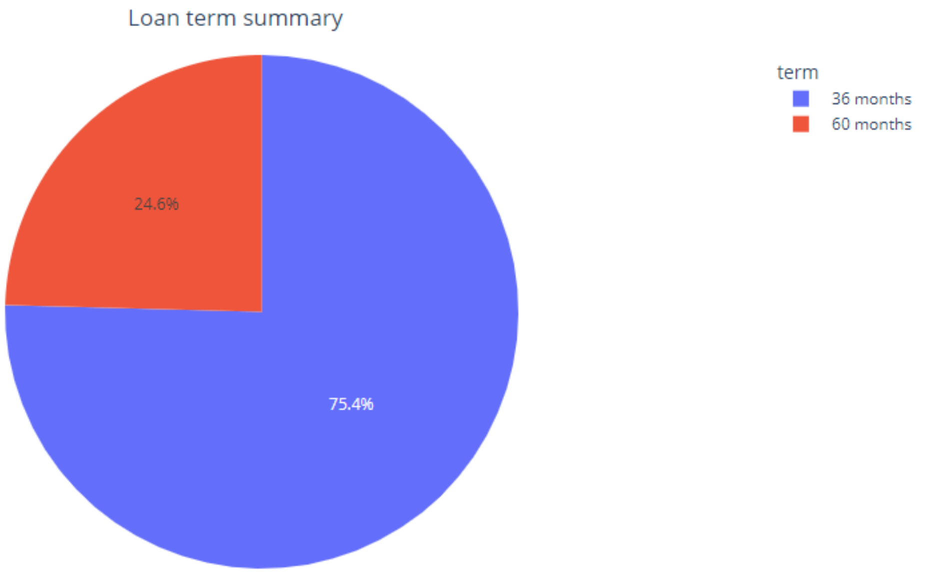
UNIVARIATE ANALYSIS

In [31]:
```
df.term.value_counts().to_frame()
```

Out[31]:

	term
36 months	29096
60 months	9481

In [32]:
```
import plotly.express as px
fig = px.pie(df.term.value_counts().to_frame(),values='term',names=df.term.value_counts().to_frame().index)
fig.update_layout(
    title={
        'text': "Loan term summary",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="term")

fig.show()
```



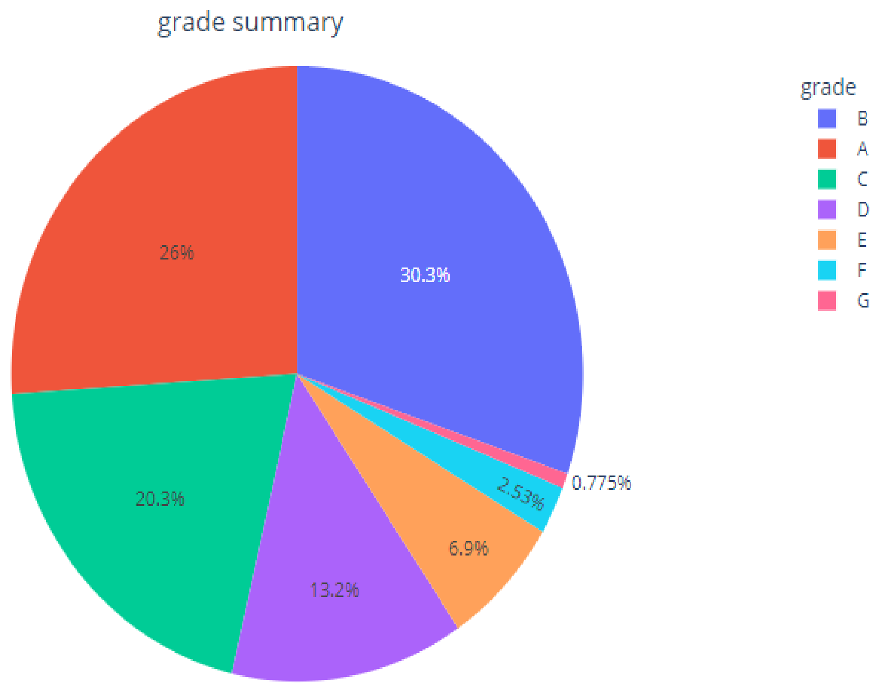
Around 75% of borrowers opted for loan term 36 months.

Remaining 25% of borrowers opted longer loan term i.e., 60 months.

UNIVARIATE ANALYSIS

In [34]:

```
import plotly.express as px
fig = px.pie(df.grade.value_counts().to_frame(),values='grade',names=df.grade.value_counts().to_frame().index)
fig.update_layout(
    title={
        'text': "grade summary",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="grade")

fig.show()
```



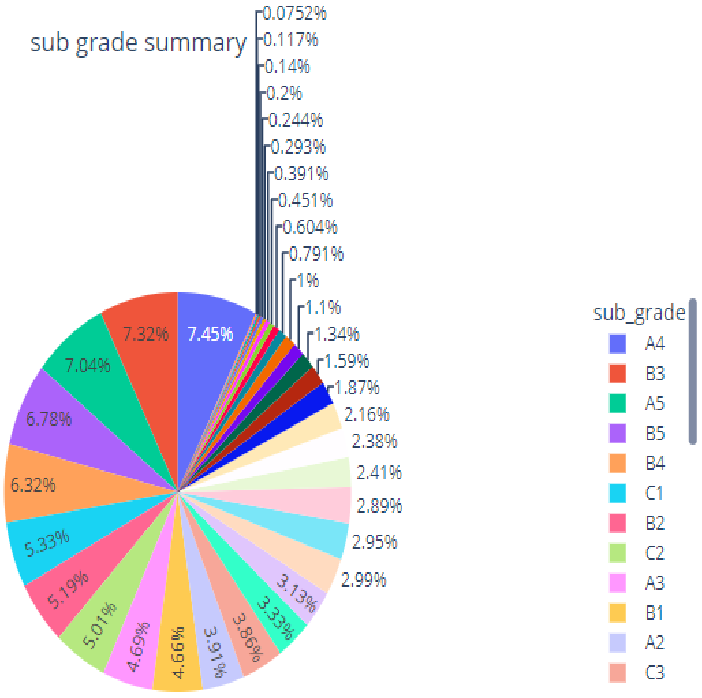
Most borrowers have assigned either A, B, C grade.

Few of borrowers also been assigned E, F, G grade as well.

UNIVARIATE ANALYSIS

In [36]:

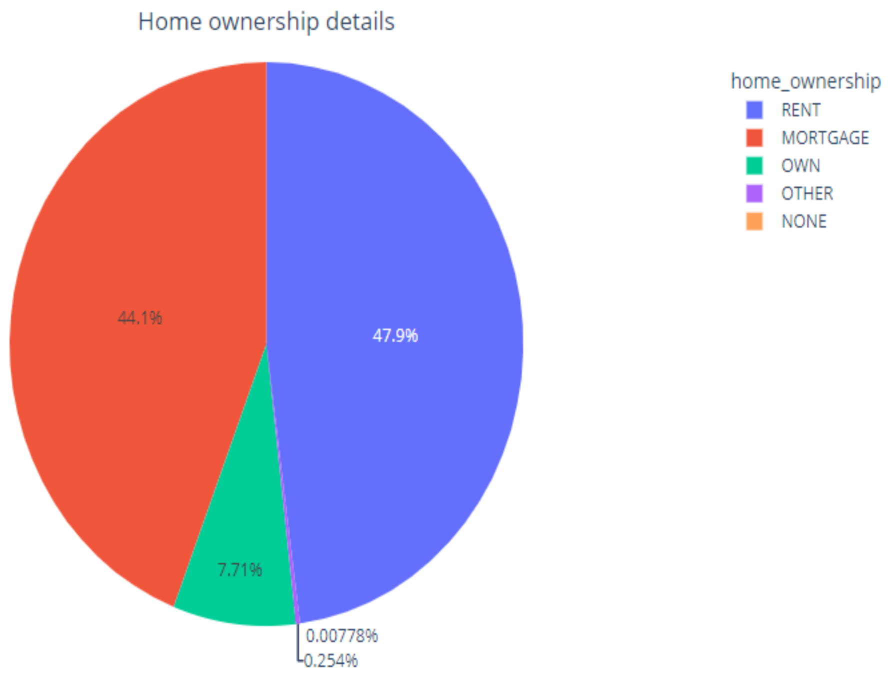
```
import plotly.express as px
fig = px.pie(df.sub_grade.value_counts().to_frame(),values='sub_grade',names=df.sub_grade.value_counts().to_frame().index)
fig.update_layout(
    title={
        'text': "sub grade summary",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="sub_grade")

fig.show()
```



The grades in this pie chart have further extended to subgrades and it is based on risk score.

UNIVARIATE ANALYSIS

In [38]:
```
import plotly.express as px
fig = px.pie(df.home_ownership.value_counts().to_frame(),values='home_ownership',names=df.home_ownership.value_counts().to_fr
fig.update_layout(
    title={
        'text': "Home ownership details",
        'y':0.95,
        'x':0.50,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="home_ownership")

fig.show()
```



47% borrowers lives in rented property.

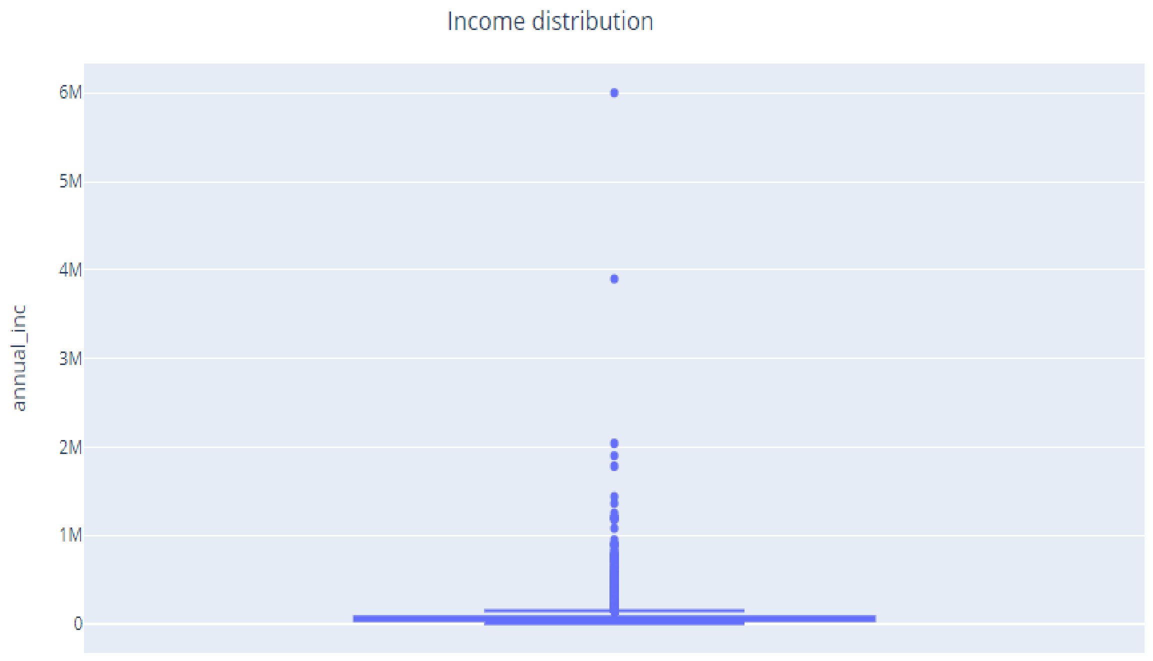
44% borrowers lives in mortgage property.

Very few 7% borrowers have their own house.

Based on this data people who lives in rented accommodation have higher tendency of taking loan.

UNIVARIATE ANALYSIS

In [40]:
```
import plotly.express as px
fig = px.box(df,y='annual_inc')
fig.update_layout(
    title={
        'text': "Income distribution",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'})

fig.show()
```



Median of income is 58K

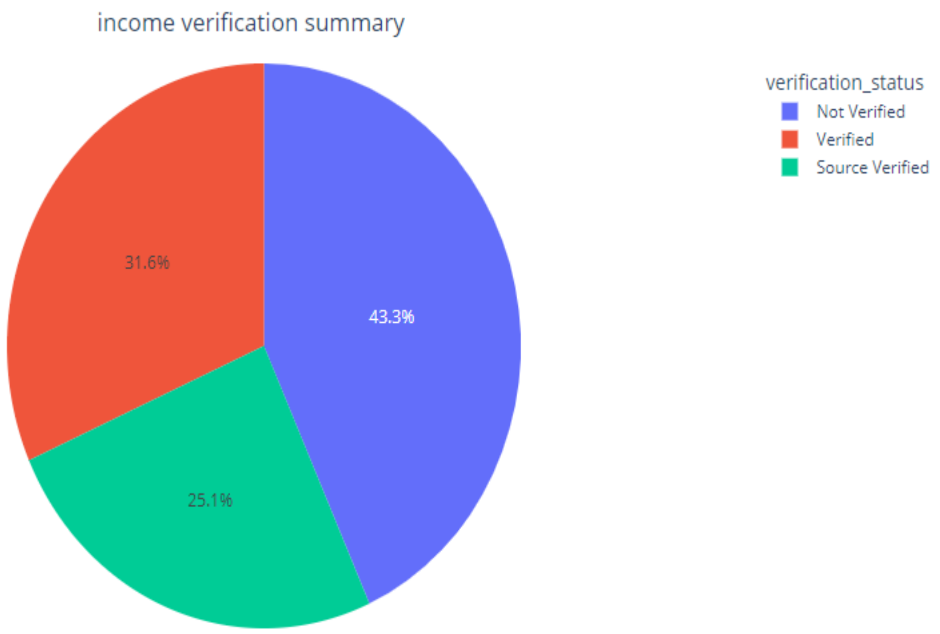
Tq1 = 40K, q3 = 82K

Upper whisker = 145K, lower whisker = 4K

The income range seems having outlier at upper range side.

UNIVARIATE ANALYSIS

```
In [42]:  import plotly.express as px
          fig = px.pie(df.verification_status.value_counts().to_frame(),values='verification_status',names=df.verification_status.value
          fig.update_layout(
              title={
                  'text': "income verification summary",
                  'y':0.95,
                  'x':0.45,
                  'xanchor': 'center',
                  'yanchor': 'top'},legend_title="verification_status")

          fig.show()
```



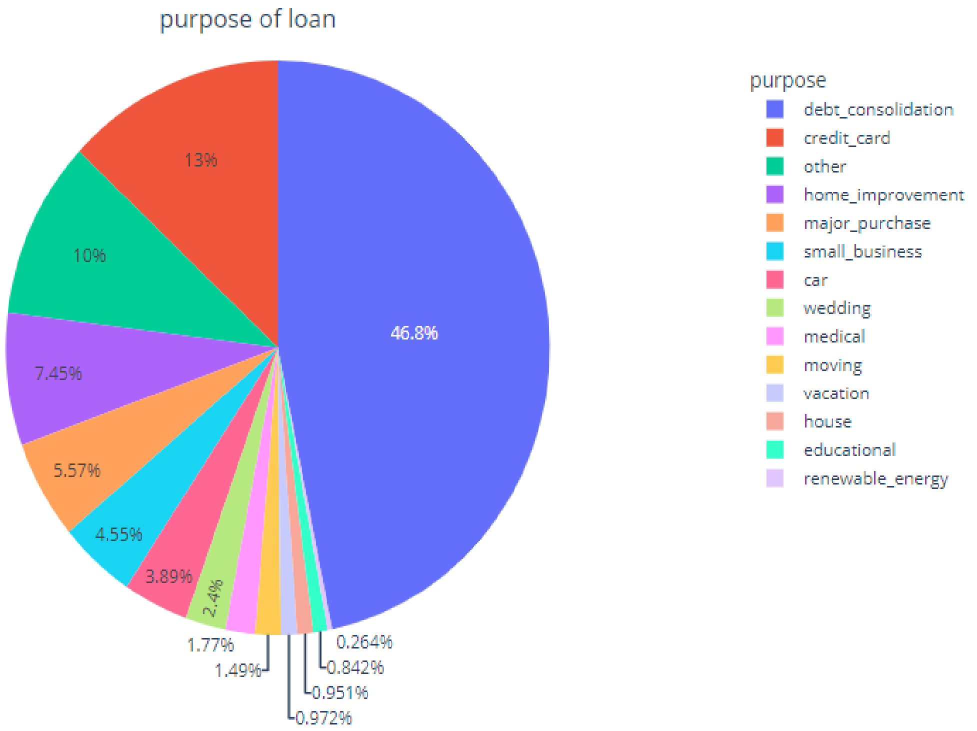
Around 55% population have verified source of income.

Remaining population have unverified source of income.

UNIVARIATE ANALYSIS

In [45]:

```
import plotly.express as px
fig = px.pie(df.purpose.value_counts().to_frame(),values='purpose',names=df.purpose.value_counts().to_frame().index)
fig.update_layout(
    title={
        'text': "purpose of loan",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="purpose")

fig.show()
```

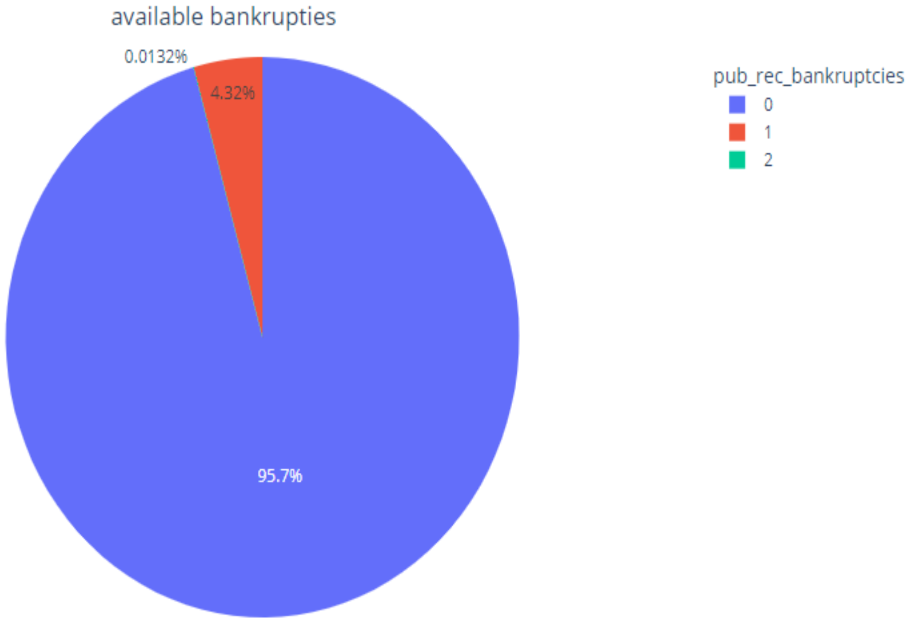


Around 46.8% borrowers have taken loan for debt consolidation, while 13% have taken for credit card.

Remaining have taken for other purpose.

UNIVARIATE ANALYSIS

In [49]:
```
fig = px.pie(df.pub_rec_bankruptcies.value_counts().to_frame(),values='pub_rec_bankruptcies',names=df.pub_rec_bankruptcies.va
fig.update_layout(
    title={
        'text': "available bankrupties",
        'y':0.95,
        'x':0.45,
        'xanchor': 'center',
        'yanchor': 'top'},legend_title="pub_rec_bankruptcies")
fig.show()
```



Around 95% borrowers have no public record for bankruptcies.

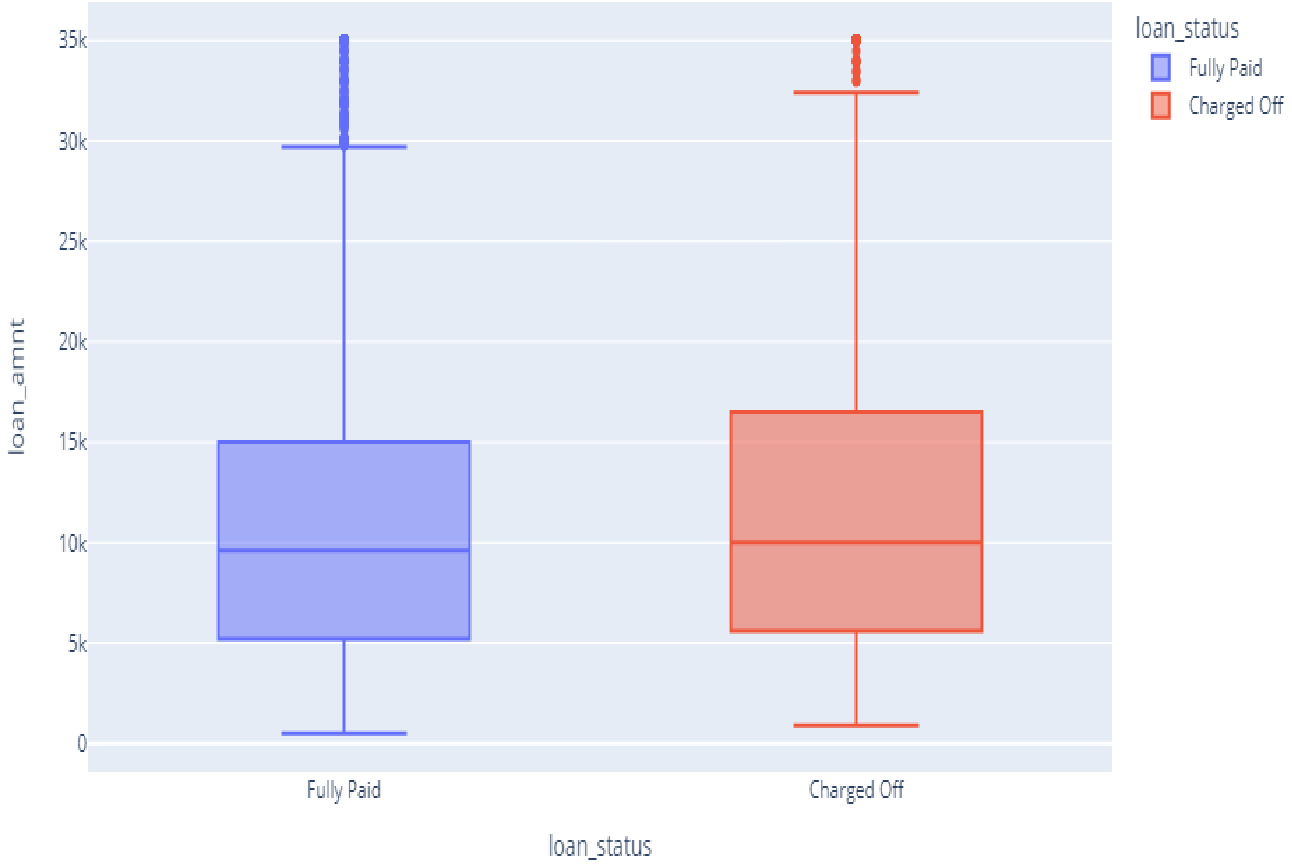
4% borrowers have 1 public records for bankruptcies.

0.01% borrowers have 2 public records for bankruptcies.

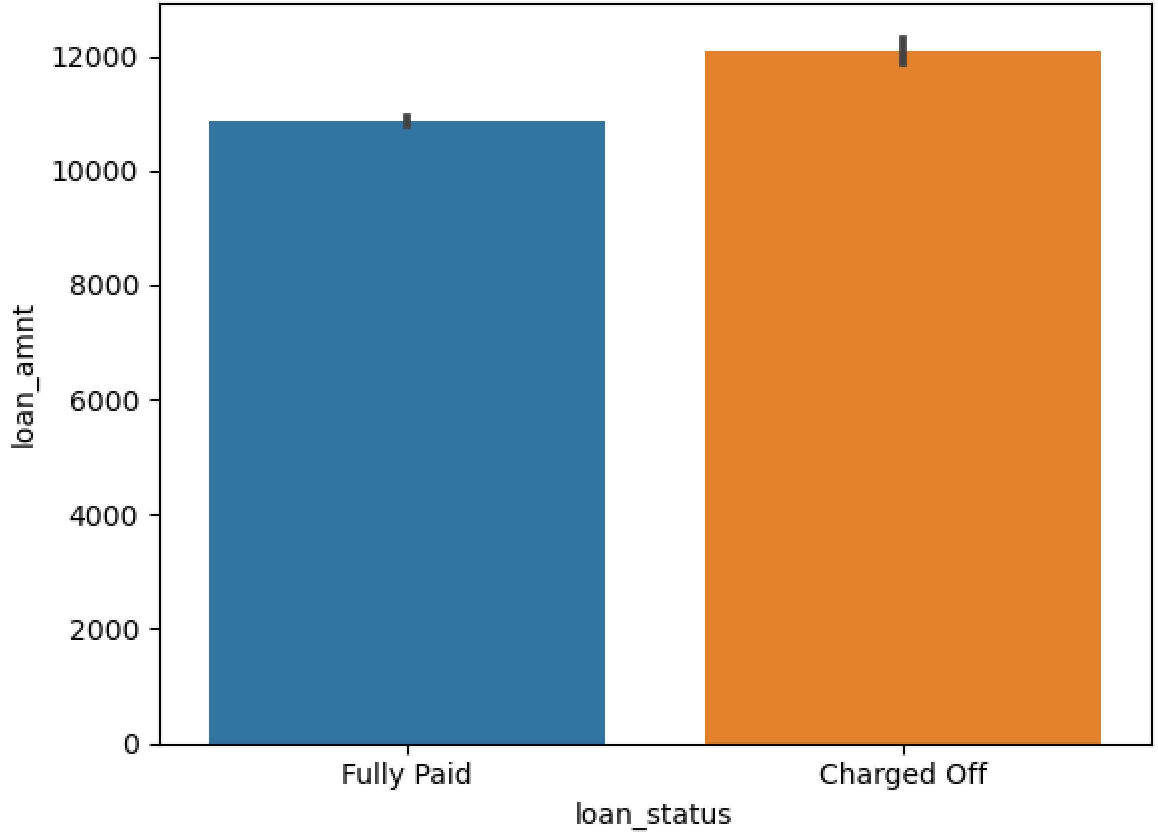
BIVARIATE ANALYSIS

BIVARIATE ANALYSIS

```
In [54]:  # sns.boxplot(data=df,x='loan_status',y='loan_amnt')
          # plt.show()
          import plotly.express as px
          fig = px.box(df,x='loan_status',y='loan_amnt',color="loan_status")
          fig.show()
```




```
In [52]:  sns.barplot(data=df,x='loan_status', y='loan_amnt')
          plt.show()

          # import plotly.express as px
          # fig = px.bar(df,x="loan_status", y="loan_amnt")
          # fig.show()
```

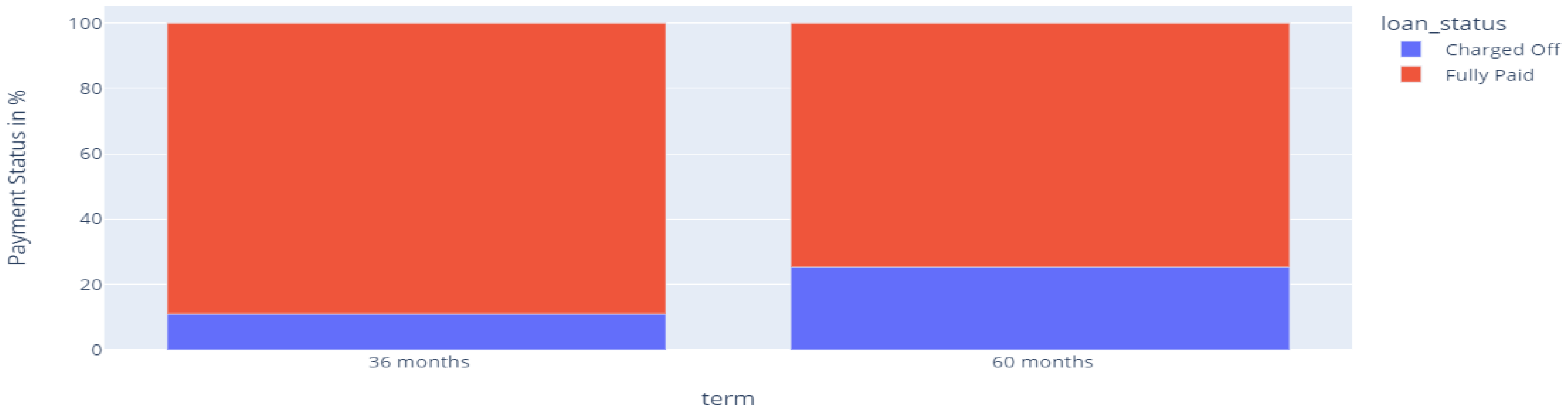


The average loan amount for Charged off cases is slightly higher than fully paid cases

BIVARIATE ANALYSIS

Term VS Loan_Status

```
In [59]:  import plotly.express as px

          fig = px.bar(df3, x='term', y='loan_status%',hover_data=['count'],
                        color='loan_status',
                       labels={'loan_status%':'Payment Status in %'}, height=400)
          fig.show()
```

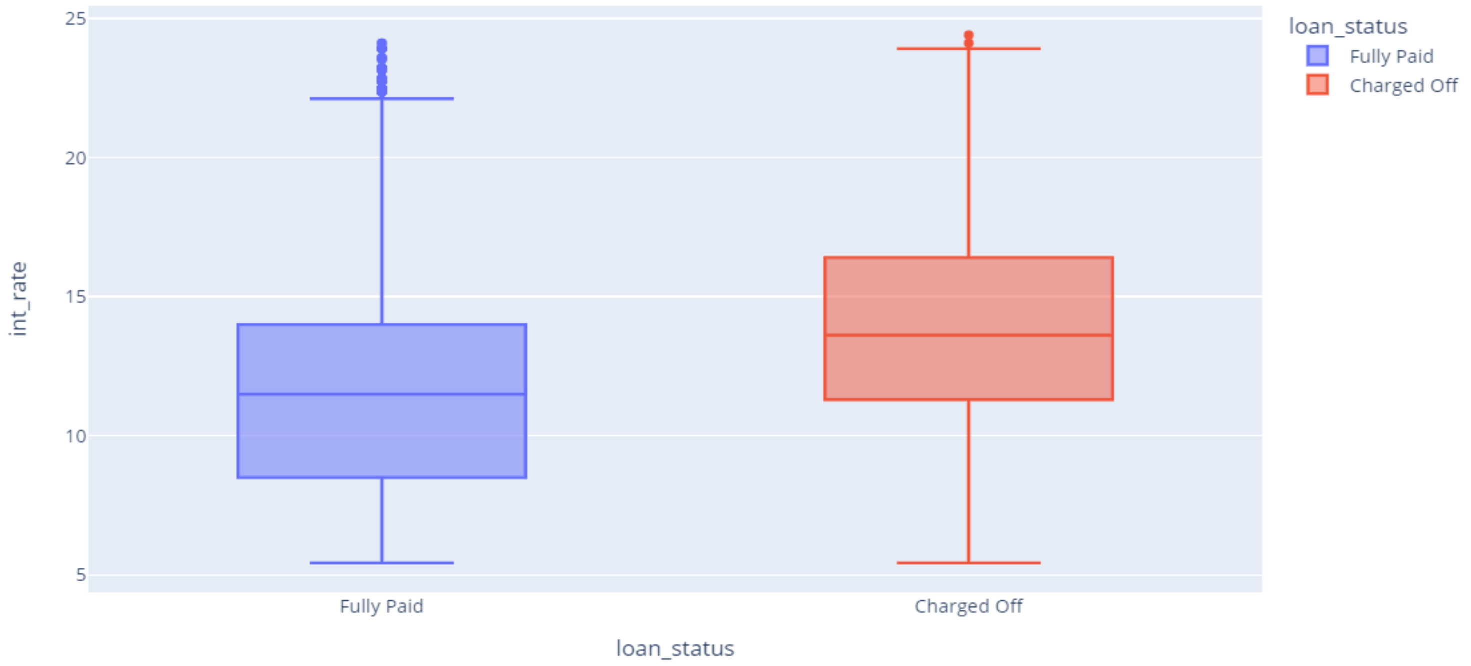


Observation: The chances of getting default is 2 times higher for term 60 months as compared to 36 months.

BIVARIATE ANALYSIS

Int_rate VS Loan_Status

In [65]:
```
fig = px.box(df4,x='loan_status',y='int_rate',color="loan_status")
fig.show()
```

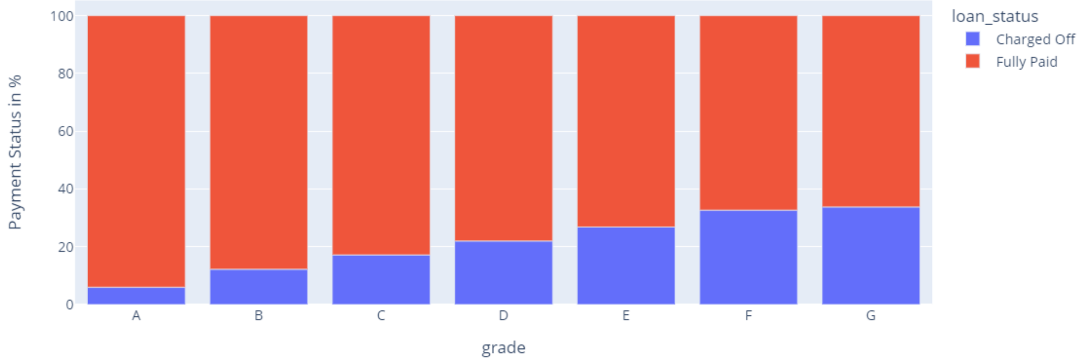


Observation: The loan with higher interest rates have more tendency to go charged off.

BIVARIATE ANALYSIS

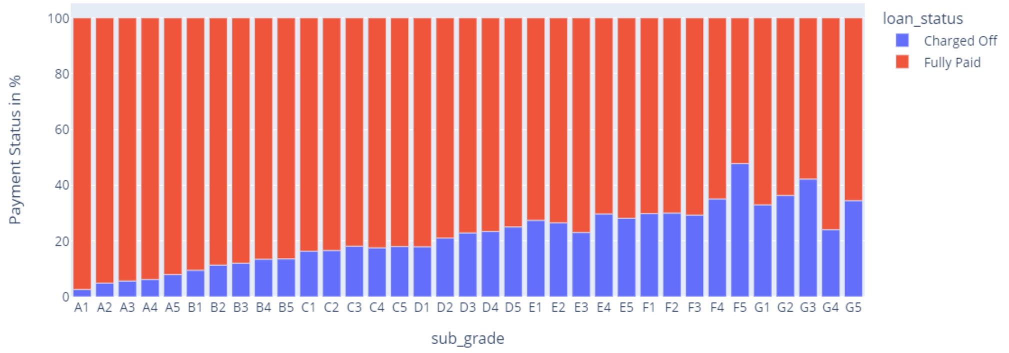
Grade VS Loan_Status

```
In [70]:  import plotly.express as px

fig = px.bar(df7, x='grade', y='loan_status%',hover_data=['count'],
               color='loan_status',
              labels={'loan_status%':'Payment Status in %'}, height=400)
fig.show()
```



Sub_Grade VS Loan_Status

```
In [74]:  import plotly.express as px

fig = px.bar(df10, x='sub_grade', y='loan_status%',hover_data=['count'],
               color='loan_status',
              labels={'loan_status%':'Payment Status in %'}, height=400)
fig.show()
```



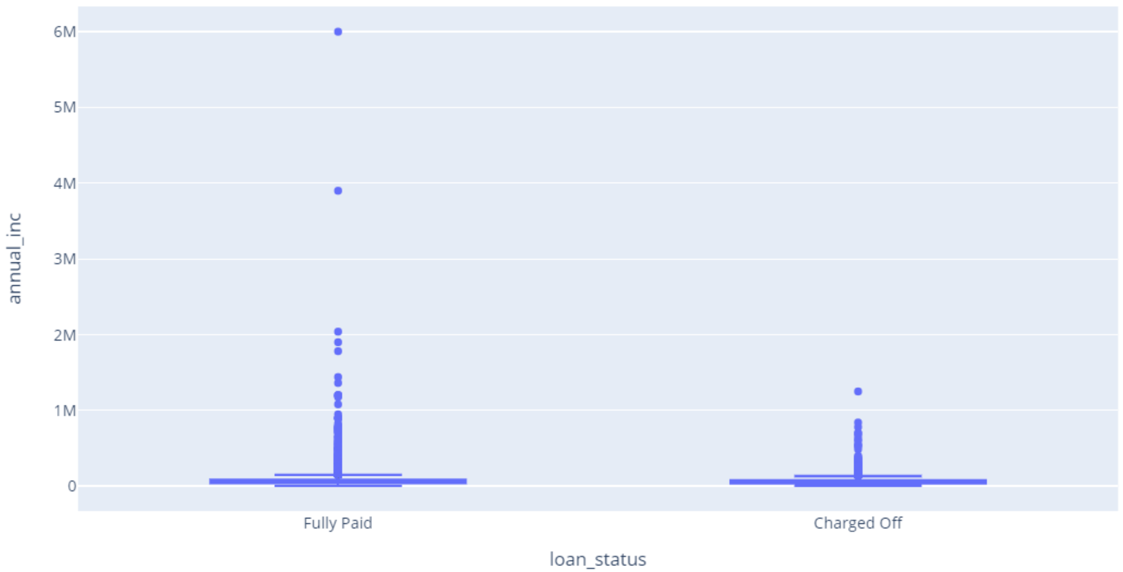
Observation: From above grade and sub_grade graph the grade E,F,G and associated subgrade have higher chance to become charged off(defaulter).

Recommendation: If the loan applications comes under grade E,F, G and associated subgrages more security is required.

BIVARIATE ANALYSIS

Income VS Loan_Status

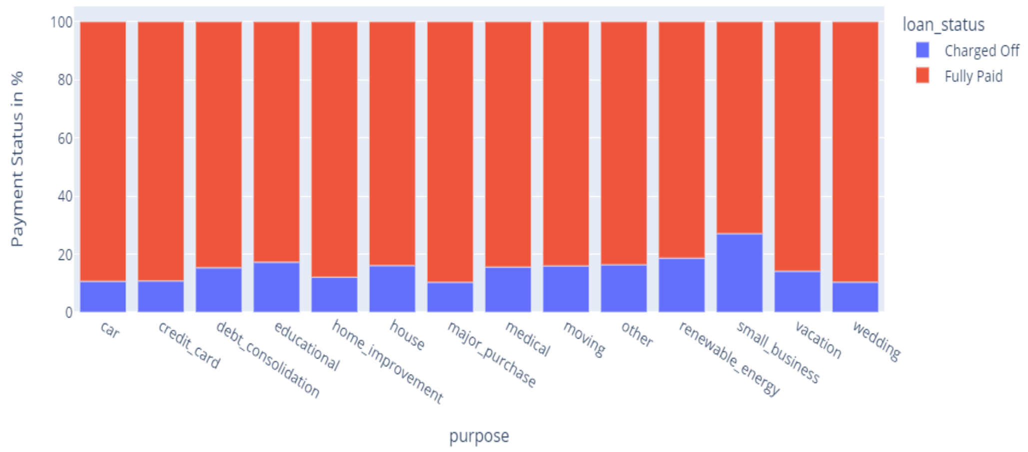
```
In [75]: import plotly.express as px
         fig = px.box(df, x='loan_status', y='annual_inc')
         fig.show()
```



Observation: The median annual for fully paid category is 60k while for charged off it is 53k. That means the people having low income have higher tendency of getting charged off.

Purpose VS Loan_Status

```
In [83]: import plotly.express as px

         fig = px.bar(df17, x='purpose', y='loan_status%',hover_data=['count'],
                      color='loan_status',
                     labels={'loan_status%':'Payment Status in %'}, height=400)
         fig.show()
```

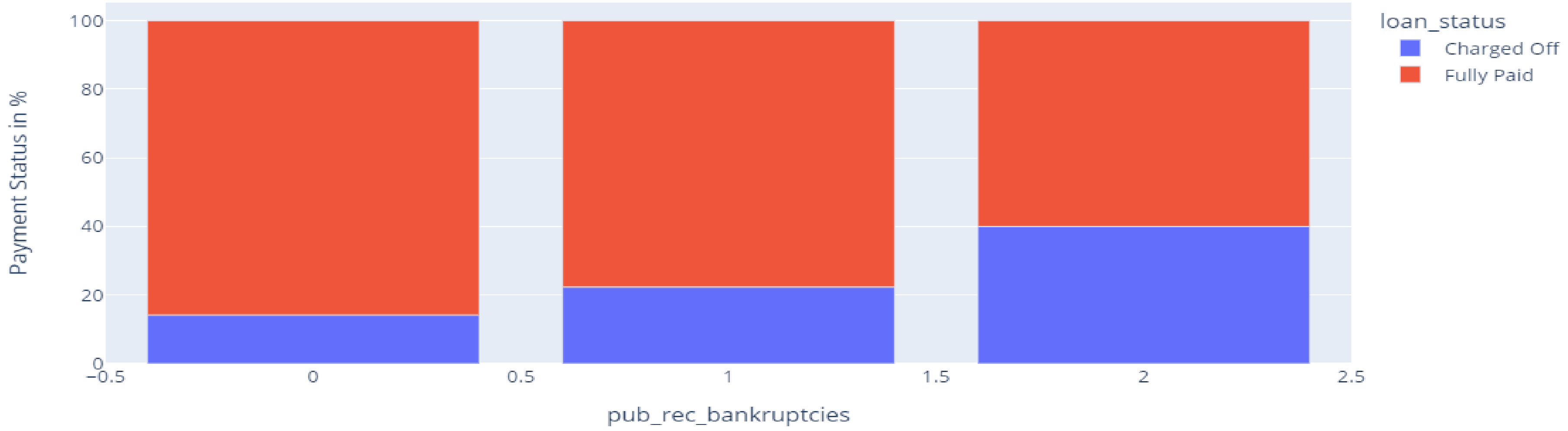


Observation: The chances of getting charged off is 27% for the purpose “small_business”.
Recommendation: More scrutiny is required for risk loan purpose.

BIVARIATE ANALYSIS

Income VS Loan_Status

```
In [87]:
import plotly.express as px

fig = px.bar(df20, x='pub_rec_bankruptcies', y='loan_status%',hover_data=['count'],
             color='loan_status',
            labels={'loan_status%':'Payment Status in %'}, height=400)
fig.show()
```



Observation: The chances of getting charged is very high for the applicants having at least one public bankruptcies records.

Recommendation: Do not offer loan to applicants having public bankruptcies record more than or eual to 2, as the % of getting charged off is around 40%.

CONCLUSION & RECOMMENDATIONS:

CONCLUSION & RECOMMENDATIONS:

After Analyzing the lending club dataset, we can inferred below insights.

- ❖ Below features are clearly driving factors that impact loan repayment certainly.
- ✓ Loan term
- ✓ Grade & sub-Grade
- ✓ Rate of interest
- ✓ Income & purpose of loan
- ✓ Public bankruptcies records

- ❖ Below features have visible indication of associated risks so must be utilized for risk scoring, if already happening then logic to be reviewed and updated accordingly.
- ✓ Grade & sub-grade
- ✓ Public bankruptcies records

CONCLUSION & RECOMMENDATIONS:

After Analyzing the lending club dataset, we can inferred below insights.

Observation :

Looks like the chances of going defaulters is high if the borrowers have atleast one public record of bankrupties.

Recommandation

Lending money to borrower having atleast 1 public record of bankrupties is risky.


Thank you!!! ...