SPADE DS Internship Assignment-2

In [1]:

```
##Importing Libraries
```

In [2]:

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline
```

In [3]:

```
df = pd.read_csv("loan.csv")
```

```
C:\Users\rohith\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:
3063: DtypeWarning: Columns (47) have mixed types.Specify dtype option on imp
ort or set low_memory=False.
  interactivity=interactivity, compiler=compiler, result=result)
```

In [4]:

```
df.head(10)
```

Out[4]:

	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment
0	1077501	1296599	5000	5000	4975.0	36 months	10.65%	162.87
1	1077430	1314167	2500	2500	2500.0	60 months	15.27%	59.83
2	1077175	1313524	2400	2400	2400.0	36 months	15.96%	84.33
3	1076863	1277178	10000	10000	10000.0	36 months	13.49%	339.31
4	1075358	1311748	3000	3000	3000.0	60 months	12.69%	67.79
5	1075269	1311441	5000	5000	5000.0	36 months	7.90%	156.46
6	1069639	1304742	7000	7000	7000.0	60 months	15.96%	170.08
7	1072053	1288686	3000	3000	3000.0	36 months	18.64%	109.43
8	1071795	1306957	5600	5600	5600.0	60 months	21.28%	152.39
9	1071570	1306721	5375	5375	5350.0	60 months	12.69%	121.45

10 rows × 111 columns

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39717 entries, 0 to 39716
Columns: 111 entries, id to total_il_high_credit_limit
dtypes: float64(74), int64(13), object(24)
memory usage: 33.6+ MB
```

In [6]:
```
df.describe(include='all')
```

Out[6]:

	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_r
count	3.971700e+04	3.971700e+04	39717.000000	39717.000000	39717.000000	39717	39
unique	NaN	NaN	NaN	NaN	NaN	2	
top	NaN	NaN	NaN	NaN	NaN	36 months	10.9
freq	NaN	NaN	NaN	NaN	NaN	29096	
mean	6.831319e+05	8.504636e+05	11219.443815	10947.713196	10397.448868	NaN	N
std	2.106941e+05	2.656783e+05	7456.670694	7187.238670	7128.450439	NaN	N
min	5.473400e+04	7.069900e+04	500.000000	500.000000	0.000000	NaN	N
25%	5.162210e+05	6.667800e+05	5500.000000	5400.000000	5000.000000	NaN	N
50%	6.656650e+05	8.508120e+05	10000.000000	9600.000000	8975.000000	NaN	N
75%	8.377550e+05	1.047339e+06	15000.000000	15000.000000	14400.000000	NaN	N
max	1.077501e+06	1.314167e+06	35000.000000	35000.000000	35000.000000	NaN	N

11 rows × 111 columns

In [7]:
```
df.columns
```

Out[7]:
```
Index(['id', 'member_id', 'loan_amnt', 'funded_amnt', 'funded_amnt_inv',
       'term', 'int_rate', 'installment', 'grade', 'sub_grade',
       ...
       'num_tl_90g_dpd_24m', 'num_tl_op_past_12m', 'pct_tl_nvr_dlq',
       'percent_bc_gt_75', 'pub_rec_bankruptcies', 'tax_liens',
       'tot_hi_cred_lim', 'total_bal_ex_mort', 'total_bc_limit',
       'total_il_high_credit_limit'],
      dtype='object', length=111)
```

In [8]:
```
df.dropna(axis=1, how="all",inplace=True)
```

In [9]:
```
df.head(10)
```

Out[9]:

	id	member_id	loan_amnt	funded_amnt	funded_amnt_inv	term	int_rate	installment
0	1077501	1296599	5000	5000	4975.0	36 months	10.65%	162.87
1	1077430	1314167	2500	2500	2500.0	60 months	15.27%	59.83
2	1077175	1313524	2400	2400	2400.0	36 months	15.96%	84.33
3	1076863	1277178	10000	10000	10000.0	36 months	13.49%	339.31
4	1075358	1311748	3000	3000	3000.0	60 months	12.69%	67.79
5	1075269	1311441	5000	5000	5000.0	36 months	7.90%	156.46
6	1069639	1304742	7000	7000	7000.0	60 months	15.96%	170.08
7	1072053	1288686	3000	3000	3000.0	36 months	18.64%	109.43
8	1071795	1306957	5600	5600	5600.0	60 months	21.28%	152.39
9	1071570	1306721	5375	5375	5350.0	60 months	12.69%	121.45

10 rows × 57 columns

In [10]:
```
df.columns
```

Out[10]:
```
Index(['id', 'member_id', 'loan_amnt', 'funded_amnt', 'funded_amnt_inv',
       'term', 'int_rate', 'installment', 'grade', 'sub_grade', 'emp_title',
       'emp_length', 'home_ownership', 'annual_inc', 'verification_status',
       'issue_d', 'loan_status', 'pymnt_plan', 'url', 'desc', 'purpose',
       'title', 'zip_code', 'addr_state', 'dti', 'delinq_2yrs',
       'earliest_cr_line', 'inq_last_6mths', 'mths_since_last_delinq',
       'mths_since_last_record', 'open_acc', 'pub_rec', 'revol_bal',
       'revol_util', 'total_acc', 'initial_list_status', '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',
       'next_pymnt_d', 'last_credit_pull_d', 'collections_12_mths_ex_med',
       'policy_code', 'application_type', 'acc_now_delinq',
       'chargeoff_within_12_mths', 'delinq_amnt', 'pub_rec_bankruptcies',
       'tax_liens'],
      dtype='object')
```

In [11]:
```
df = df[['loan_amnt', 'funded_amnt_inv', 'term', 'int_rate', 'installment', 'g
rade', 'sub_grade', 'emp_length', 'home_ownership', 'annual_inc', 'verificatio
n_status', 'issue_d', 'loan_status', 'purpose', 'dti', 'earliest_cr_line', 'in
q_last_6mths', 'open_acc', 'pub_rec', 'revol_util', 'total_acc']]
```

In [12]:
```
df.head(10)
```

Out[12]:

	loan_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	emp_length	hc
0	5000	4975.0	36 months	10.65%	162.87	B	B2	10+ years	
1	2500	2500.0	60 months	15.27%	59.83	C	C4	< 1 year	
2	2400	2400.0	36 months	15.96%	84.33	C	C5	10+ years	
3	10000	10000.0	36 months	13.49%	339.31	C	C1	10+ years	
4	3000	3000.0	60 months	12.69%	67.79	B	B5	1 year	
5	5000	5000.0	36 months	7.90%	156.46	A	A4	3 years	
6	7000	7000.0	60 months	15.96%	170.08	C	C5	8 years	
7	3000	3000.0	36 months	18.64%	109.43	E	E1	9 years	
8	5600	5600.0	60 months	21.28%	152.39	F	F2	4 years	
9	5375	5350.0	60 months	12.69%	121.45	B	B5	< 1 year	

10 rows × 21 columns

In [13]:
```
df.dtypes
```

Out[13]:
```
loan_amnt                 int64
funded_amnt_inv         float64
term                     object
int_rate                 object
installment             float64
grade                    object
sub_grade                object
emp_length               object
home_ownership           object
annual_inc              float64
verification_status      object
issue_d                  object
loan_status              object
purpose                  object
dti                     float64
earliest_cr_line         object
inq_last_6mths            int64
open_acc                  int64
pub_rec                   int64
revol_util               object
total_acc                 int64
dtype: object
```

In [14]:
```
df.describe()
```

Out[14]:

	loan_amnt	funded_amnt_inv	installment	annual_inc	dti	inq_last_6mths
count	39717.000000	39717.000000	39717.000000	3.971700e+04	39717.000000	39717.00000
mean	11219.443815	10397.448868	324.561922	6.896893e+04	13.315130	0.86920
std	7456.670694	7128.450439	208.874874	6.379377e+04	6.678594	1.07021
min	500.000000	0.000000	15.690000	4.000000e+03	0.000000	0.00000
25%	5500.000000	5000.000000	167.020000	4.040400e+04	8.170000	0.00000
50%	10000.000000	8975.000000	280.220000	5.900000e+04	13.400000	1.00000
75%	15000.000000	14400.000000	430.780000	8.230000e+04	18.600000	1.00000
max	35000.000000	35000.000000	1305.190000	6.000000e+06	29.990000	8.00000

In [15]:
```
df.describe(include='all')
```

Out[15]:

	loan_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	emp_
count	39717.000000	39717.000000	39717	39717	39717.000000	39717	39717	
unique	NaN	NaN	2	371	NaN	7	35	
top	NaN	NaN	36 months	10.99%	NaN	B	B3	10
freq	NaN	NaN	29096	956	NaN	12020	2917	
mean	11219.443815	10397.448868	NaN	NaN	324.561922	NaN	NaN	
std	7456.670694	7128.450439	NaN	NaN	208.874874	NaN	NaN	
min	500.000000	0.000000	NaN	NaN	15.690000	NaN	NaN	
25%	5500.000000	5000.000000	NaN	NaN	167.020000	NaN	NaN	
50%	10000.000000	8975.000000	NaN	NaN	280.220000	NaN	NaN	
75%	15000.000000	14400.000000	NaN	NaN	430.780000	NaN	NaN	
max	35000.000000	35000.000000	NaN	NaN	1305.190000	NaN	NaN	

11 rows × 21 columns

In [18]:
```
df.isnull().sum()
```

Out[18]:
```
loan_amnt                 0
funded_amnt_inv           0
term                      0
int_rate                  0
installment               0
grade                     0
sub_grade                 0
emp_length             1075
home_ownership            0
annual_inc                0
verification_status       0
issue_d                   0
loan_status               0
purpose                   0
dti                       0
earliest_cr_line          0
inq_last_6mths            0
open_acc                  0
pub_rec                   0
revol_util               50
total_acc                 0
dtype: int64
```

In [19]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 39717 entries, 0 to 39716
Data columns (total 21 columns):
 #   Column               Non-Null Count  Dtype  
---  ------               --------------  -----  
 0   loan_amnt            39717 non-null  int64  
 1   funded_amnt_inv      39717 non-null  float64
 2   term                 39717 non-null  object 
 3   int_rate             39717 non-null  object 
 4   installment          39717 non-null  float64
 5   grade                39717 non-null  object 
 6   sub_grade            39717 non-null  object 
 7   emp_length           38642 non-null  object 
 8   home_ownership       39717 non-null  object 
 9   annual_inc           39717 non-null  float64
 10  verification_status  39717 non-null  object 
 11  issue_d              39717 non-null  object 
 12  loan_status          39717 non-null  object 
 13  purpose              39717 non-null  object 
 14  dti                  39717 non-null  float64
 15  earliest_cr_line     39717 non-null  object 
 16  inq_last_6mths       39717 non-null  int64  
 17  open_acc             39717 non-null  int64  
 18  pub_rec              39717 non-null  int64  
 19  revol_util           39667 non-null  object 
 20  total_acc            39717 non-null  int64  
dtypes: float64(4), int64(5), object(12)
memory usage: 6.4+ MB
```

In [20]:
```
df.dropna(inplace=True)
```

In [21]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
Int64Index: 38595 entries, 0 to 39716
Data columns (total 21 columns):
 #   Column                Non-Null Count  Dtype  
---  ------                --------------  -----  
 0   loan_amnt             38595 non-null  int64  
 1   funded_amnt_inv       38595 non-null  float64
 2   term                  38595 non-null  object 
 3   int_rate              38595 non-null  object 
 4   installment           38595 non-null  float64
 5   grade                 38595 non-null  object 
 6   sub_grade             38595 non-null  object 
 7   emp_length            38595 non-null  object 
 8   home_ownership        38595 non-null  object 
 9   annual_inc            38595 non-null  float64
 10  verification_status   38595 non-null  object 
 11  issue_d               38595 non-null  object 
 12  loan_status           38595 non-null  object 
 13  purpose               38595 non-null  object 
 14  dti                   38595 non-null  float64
 15  earliest_cr_line      38595 non-null  object 
 16  inq_last_6mths        38595 non-null  int64  
 17  open_acc              38595 non-null  int64  
 18  pub_rec               38595 non-null  int64  
 19  revol_util            38595 non-null  object 
 20  total_acc             38595 non-null  int64  
dtypes: float64(4), int64(5), object(12)
memory usage: 6.5+ MB
```

In [22]:
```
df.isnull().sum()
```

Out[22]:
```
loan_amnt              0
funded_amnt_inv        0
term                   0
int_rate               0
installment            0
grade                  0
sub_grade              0
emp_length             0
home_ownership         0
annual_inc             0
verification_status    0
issue_d                0
loan_status            0
purpose                0
dti                    0
earliest_cr_line       0
inq_last_6mths         0
open_acc               0
pub_rec                0
revol_util             0
total_acc              0
dtype: int64
```

In [23]:
```
df.head(10)
```

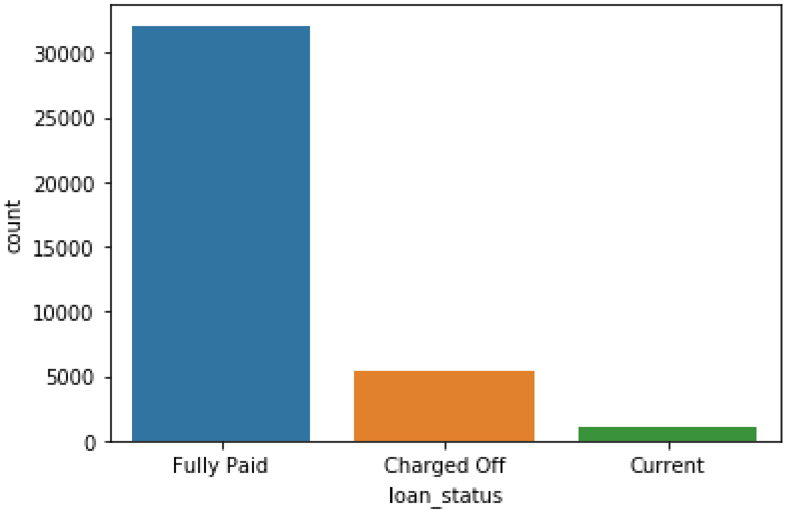
Out[23]:

	loan_amnt	funded_amnt_inv	term	int_rate	installment	grade	sub_grade	emp_length	hc
0	5000	4975.0	36 months	10.65%	162.87	B	B2	10+ years	
1	2500	2500.0	60 months	15.27%	59.83	C	C4	< 1 year	
2	2400	2400.0	36 months	15.96%	84.33	C	C5	10+ years	
3	10000	10000.0	36 months	13.49%	339.31	C	C1	10+ years	
4	3000	3000.0	60 months	12.69%	67.79	B	B5	1 year	
5	5000	5000.0	36 months	7.90%	156.46	A	A4	3 years	
6	7000	7000.0	60 months	15.96%	170.08	C	C5	8 years	
7	3000	3000.0	36 months	18.64%	109.43	E	E1	9 years	
8	5600	5600.0	60 months	21.28%	152.39	F	F2	4 years	
9	5375	5350.0	60 months	12.69%	121.45	B	B5	< 1 year	

10 rows × 21 columns

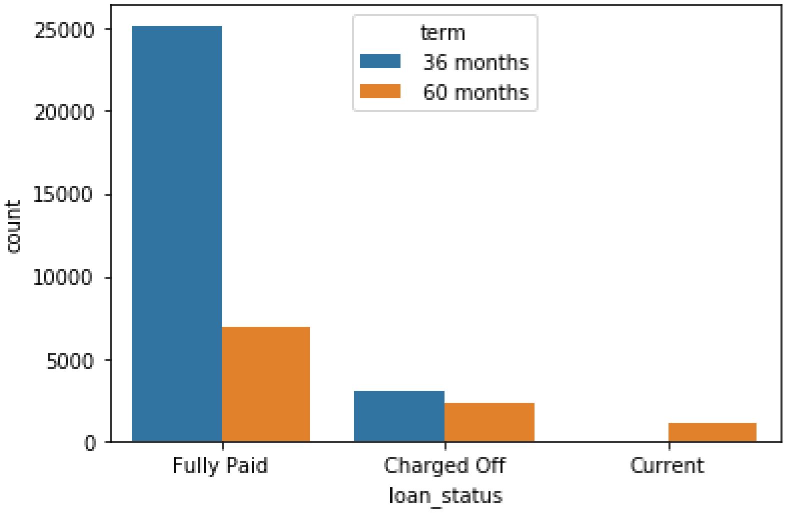
In [26]:
```
sns.countplot(x='loan_status', data=df)
```

Out[26]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb068ca2288>
```



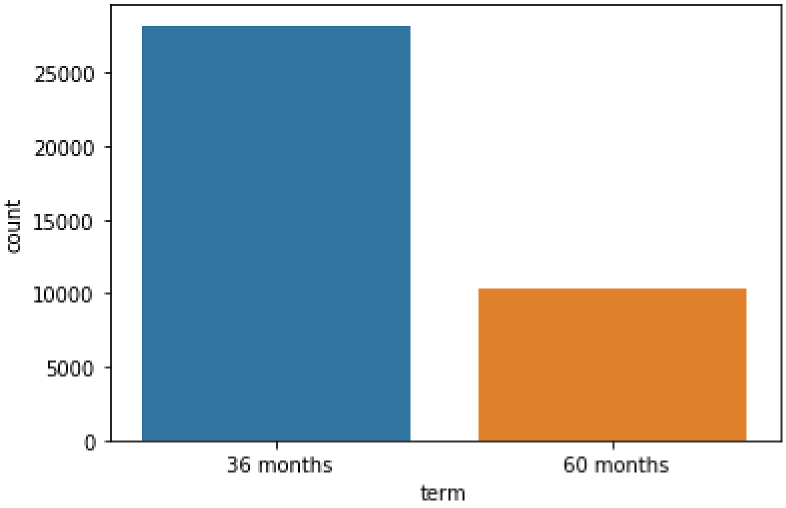
In [27]:
```
sns.countplot(x='loan_status', hue='term',data=df)
```

Out[27]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb068d007c8>
```



In [28]:
```
sns.countplot(x='term', data=df)
```

Out[28]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb068d6b348>
```



In [29]:
```
df.corr()
```

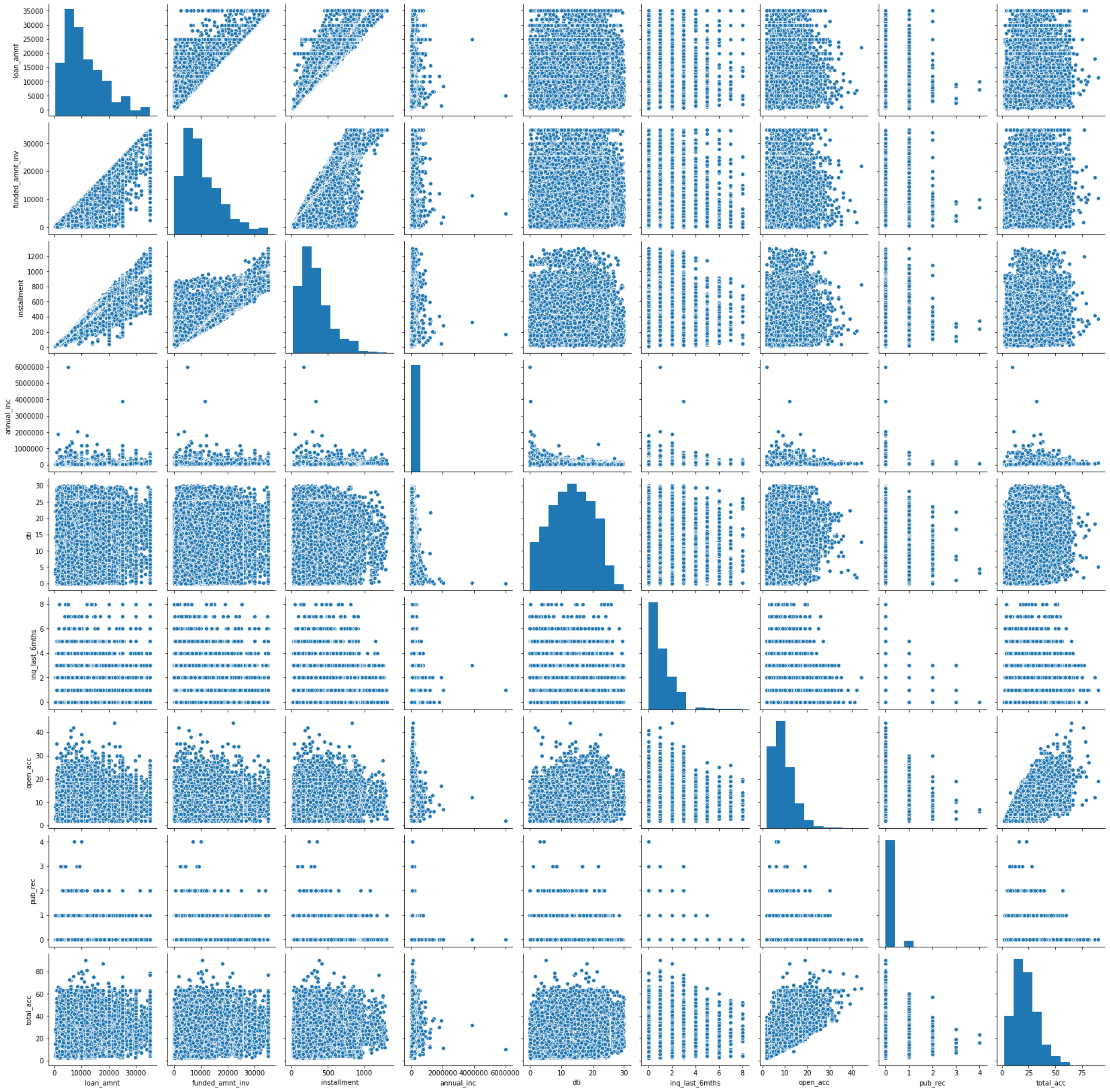
Out[29]:

	loan_amnt	funded_amnt_inv	installment	annual_inc	dti	inq_last_6mths
loan_amnt	1.000000	0.939018	0.929670	0.268364	0.064923	0.009472
funded_amnt_inv	0.939018	1.000000	0.903723	0.251601	0.073544	-0.005500
installment	0.929670	0.903723	1.000000	0.267553	0.052500	0.010011
annual_inc	0.268364	0.251601	0.267553	1.000000	-0.124861	0.034411
dti	0.064923	0.073544	0.052500	-0.124861	1.000000	-0.000477
inq_last_6mths	0.009472	-0.005500	0.010011	0.034411	-0.000477	1.000000
open_acc	0.172921	0.158680	0.168826	0.155628	0.289188	0.092278
pub_rec	-0.047936	-0.050470	-0.043268	-0.015238	-0.005077	0.024677
total_acc	0.254899	0.241188	0.229266	0.234488	0.230389	0.112151

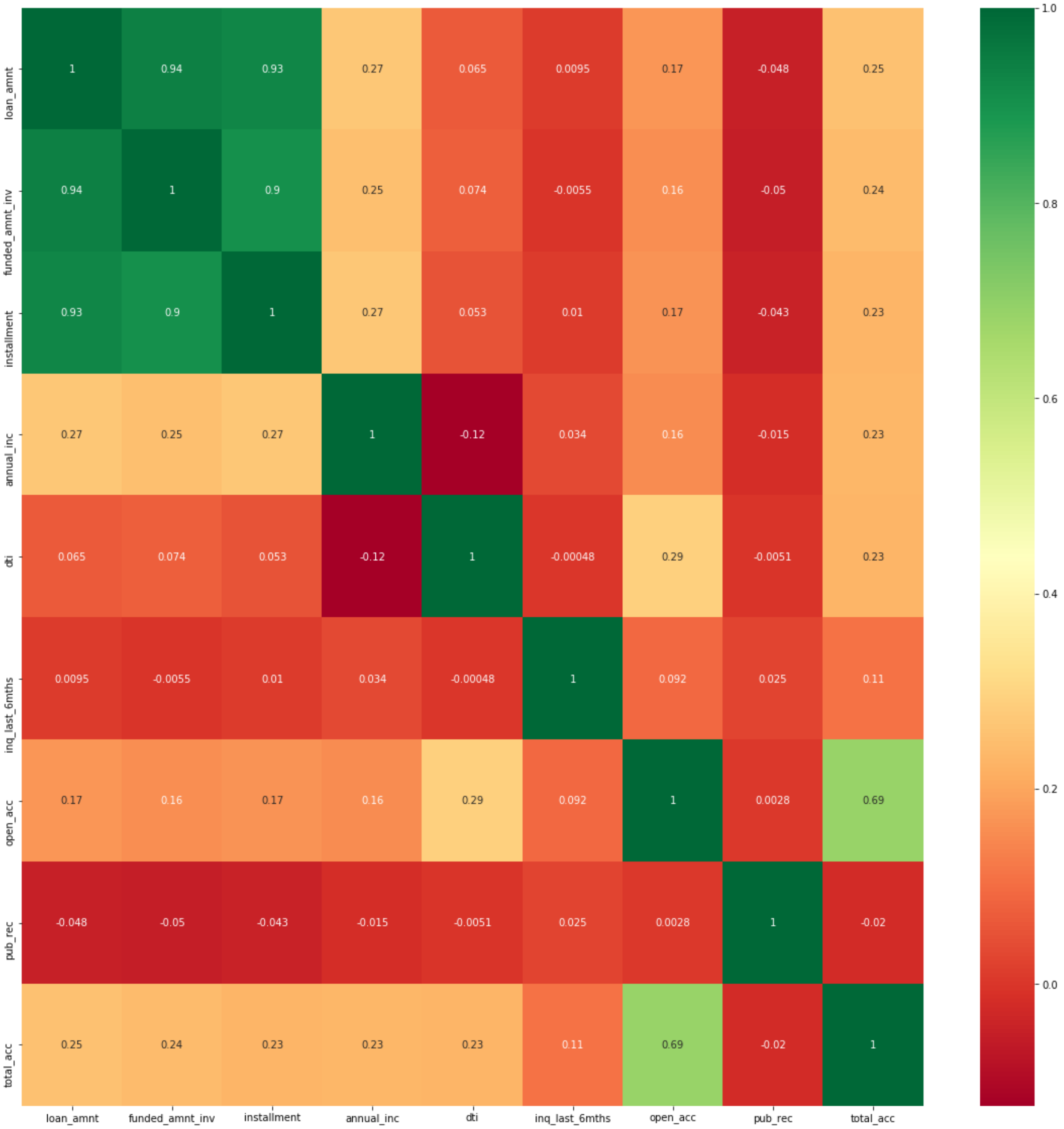
loan_amnt funded_amnt_inv installment annual_inc dti inq_last_6mths open_acc pub_rec total_acc

In [30]:
```
sns.pairplot(df)
```

Out[30]:
```
<seaborn.axisgrid.PairGrid at 0xb068dc1bc8>
```

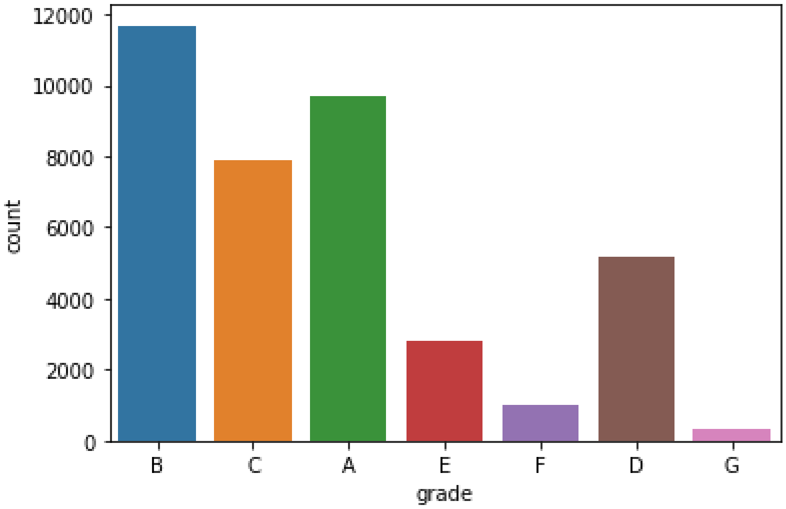


In [32]:
```
corrmat = df.corr()
top_corr_features=corrmat.index
plt.figure(figsize=(20,20))
#plot heat map
g=sns.heatmap(df[top_corr_features].corr(),annot=True,cmap="RdYlGn")
```



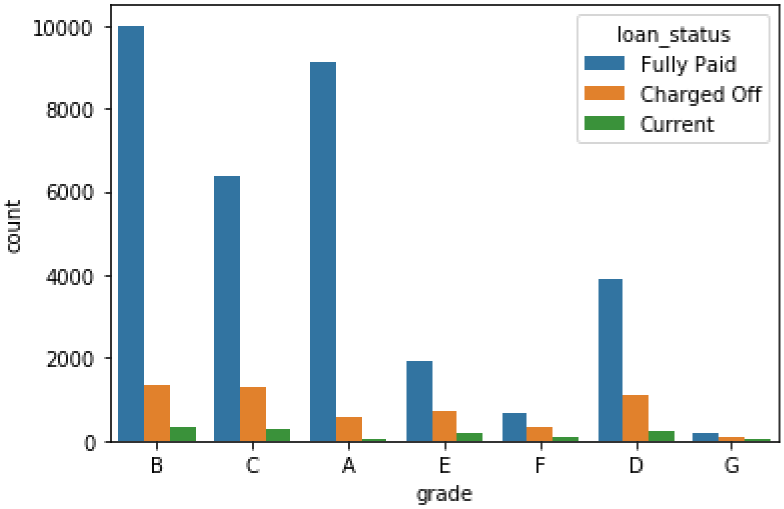
```
In [33]: sns.countplot(x='grade',data=df)
```

```
Out[33]: <matplotlib.axes._subplots.AxesSubplot at 0xb073cd1208>
```



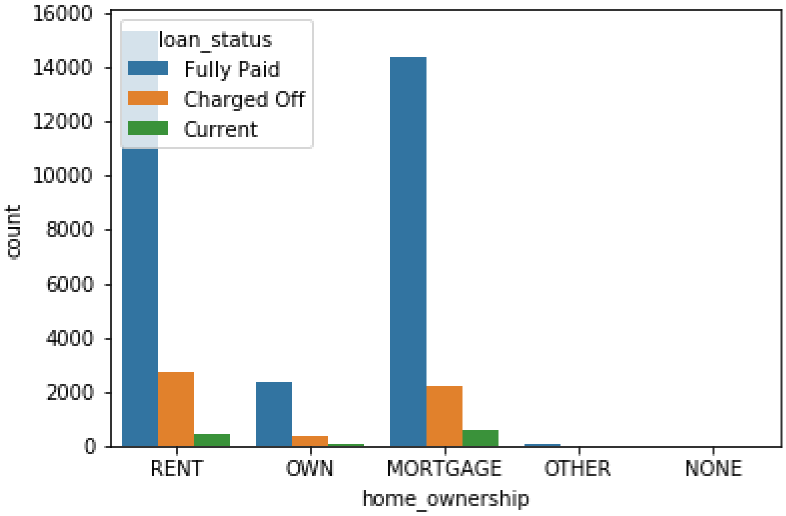
```
In [34]: sns.countplot(x='grade',hue='loan_status',data=df)
```

```
Out[34]: <matplotlib.axes._subplots.AxesSubplot at 0xb0744d6bc8>
```



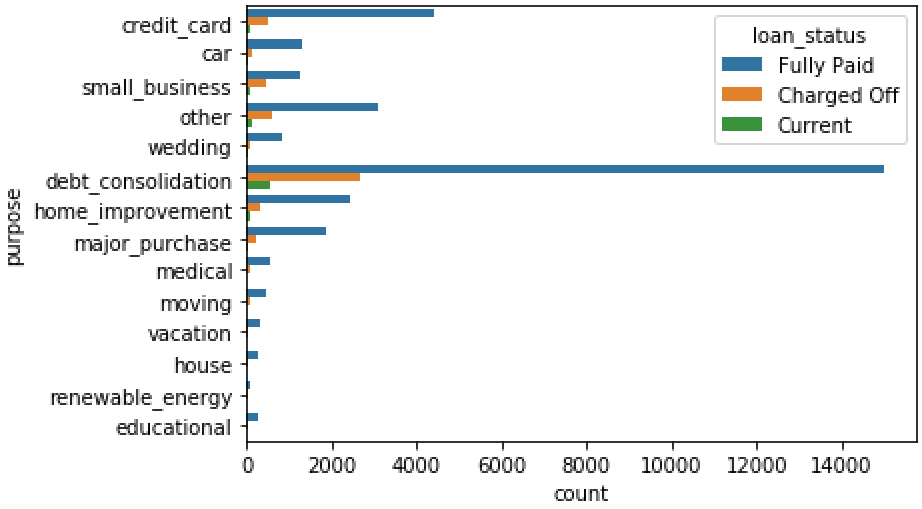
In [35]:
```
sns.countplot(x='home_ownership',hue='loan_status',data=df)
```

Out[35]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb074562e88>
```



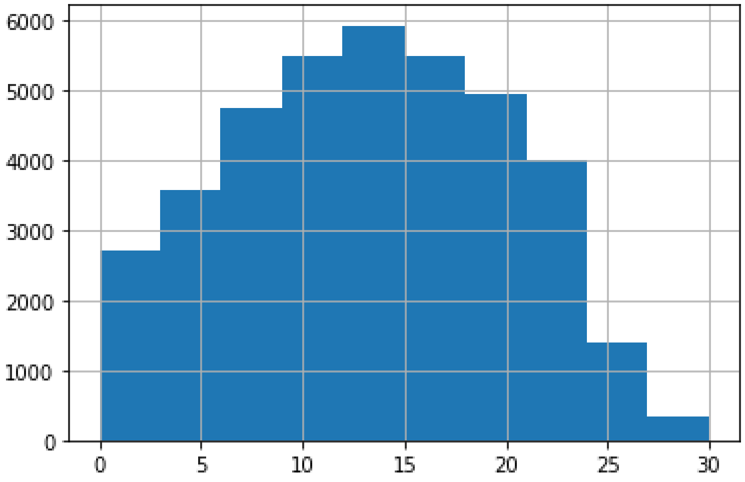
In [37]:
```
sns.countplot(y='purpose',hue='loan_status',data=df)
```

Out[37]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb074576f08>
```



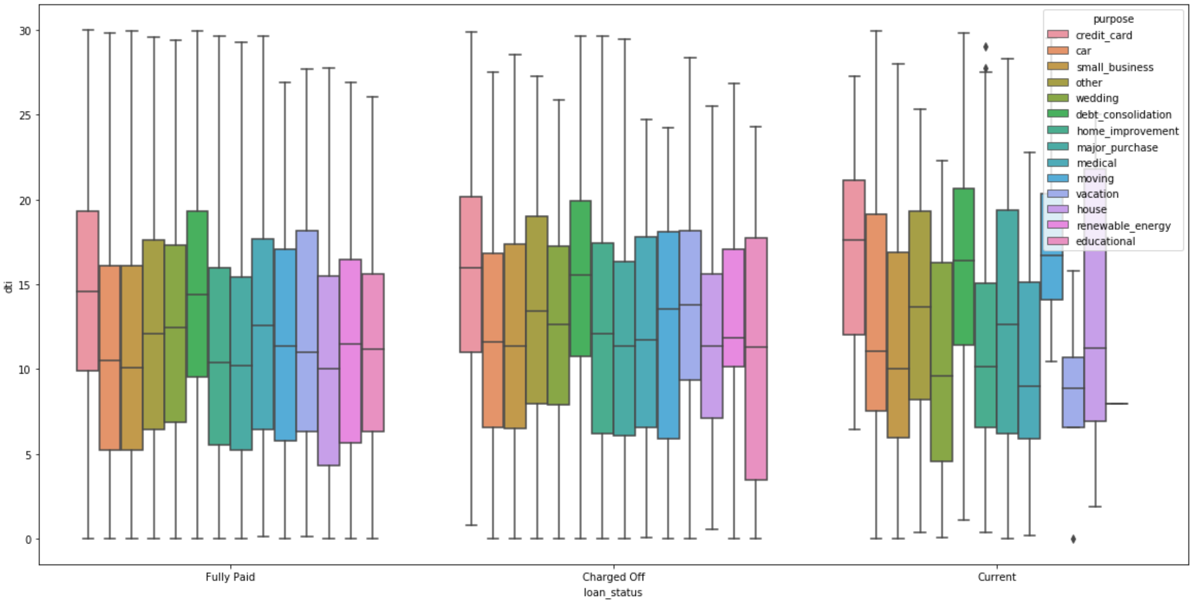
In [40]:
```
df['dti'].hist()
```

Out[40]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb06fea9908>
```



In [46]:
```
plt.figure(figsize=(20,10))
sns.boxplot(x='loan_status',y='dti',hue='purpose',data=df)
```

Out[46]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb015048b08>
```

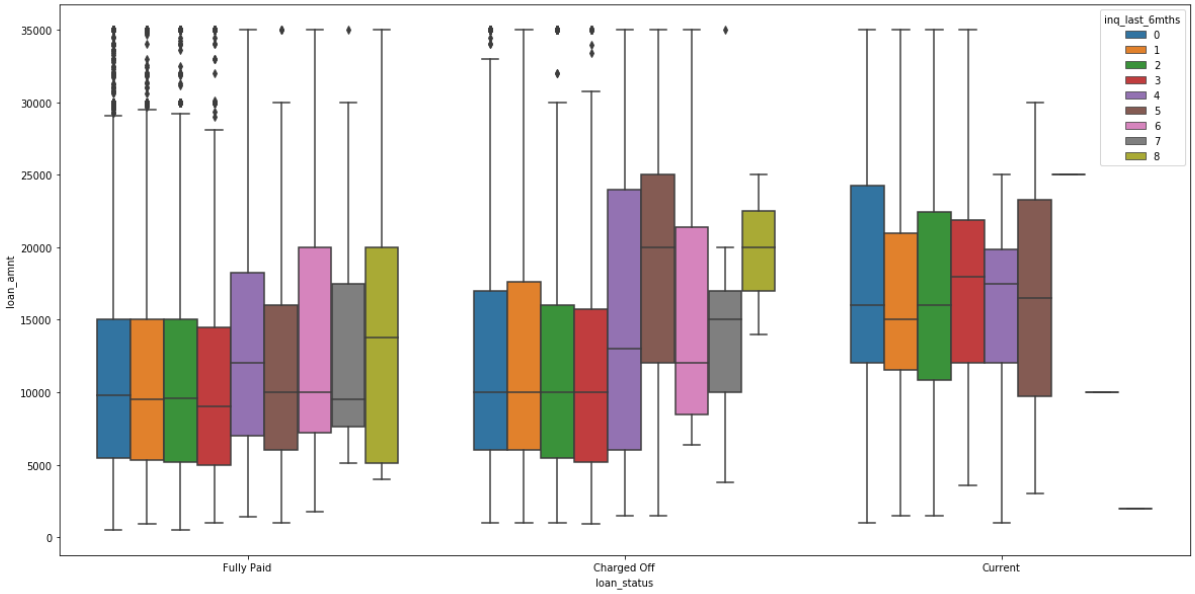


In [49]:

```
plt.figure(figsize=(20,10))
sns.boxplot(x='loan_status',y='loan_amnt',hue='inq_last_6mths',data=df)
```

Out[49]:

```
<matplotlib.axes._subplots.AxesSubplot at 0xb0166b2c88>
```

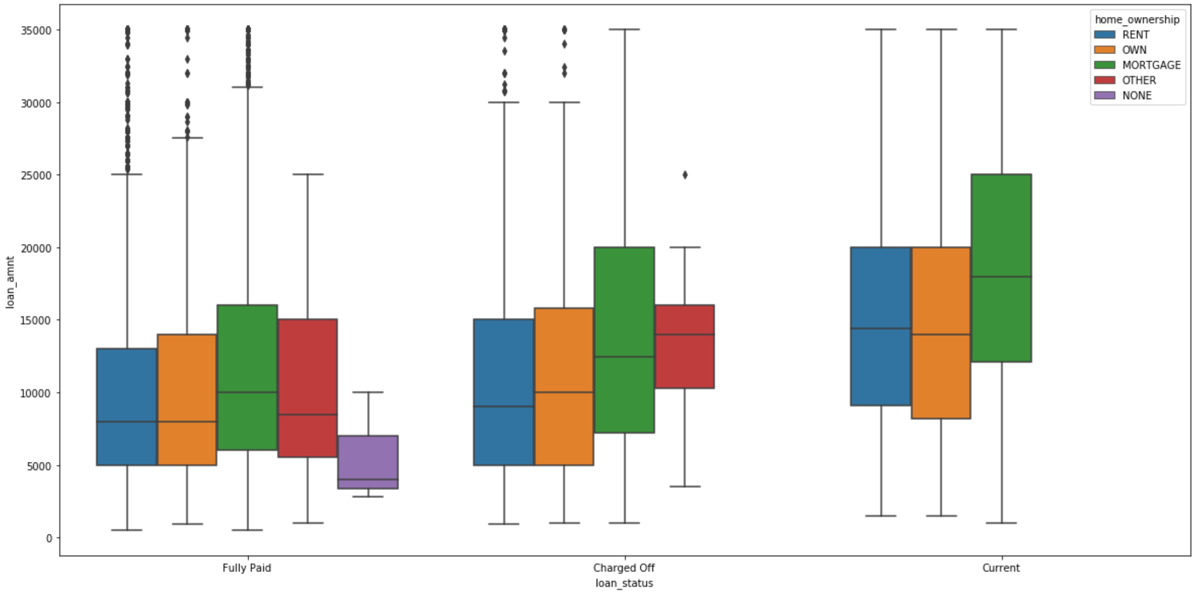


In [50]:

```
plt.figure(figsize=(20,10))
sns.boxplot(x='loan_status',y='loan_amnt',hue='home_ownership',data=df)
```

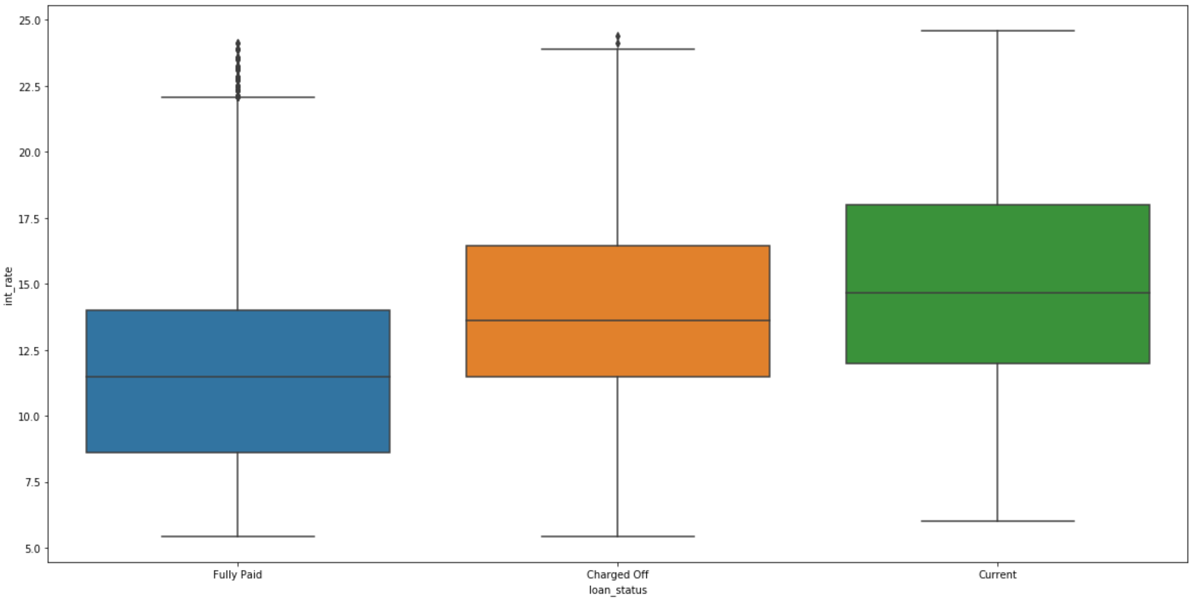
Out[50]:

```
<matplotlib.axes._subplots.AxesSubplot at 0xb0169bcf88>
```



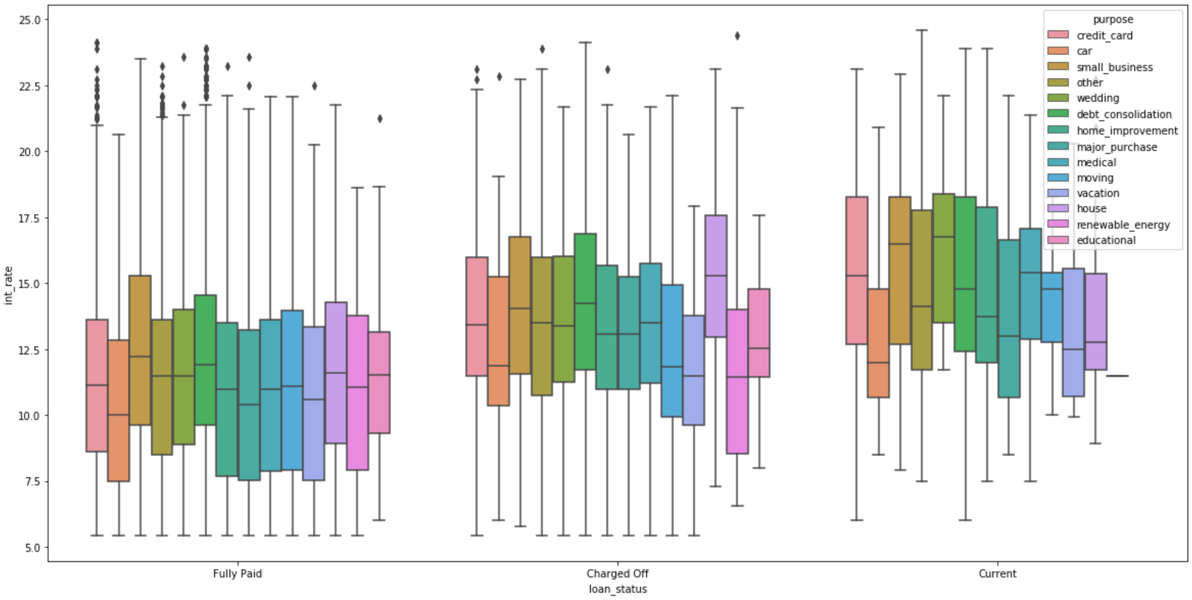
In [55]:
```
df["int_rate"] = pd.to_numeric(df["int_rate"].apply(lambda x:x.split('%')[0]))
plt.figure(figsize=(20,10))
sns.boxplot(x='loan_status',y='int_rate',data=df)
```

Out[55]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb0667c4a48>
```



In [56]:
```
plt.figure(figsize=(20,10))
sns.boxplot(x='loan_status',y='int_rate',hue='purpose',data=df)
```

Out[56]:
```
<matplotlib.axes._subplots.AxesSubplot at 0xb017e88108>
```

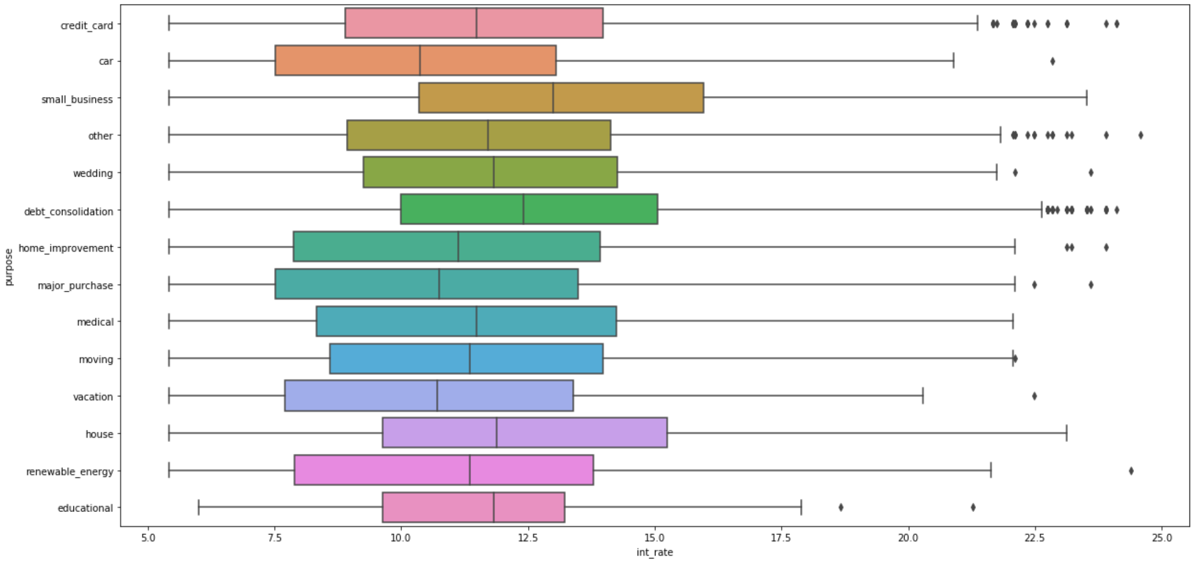


In [57]:

```
plt.figure(figsize=(20,10))
sns.boxplot(x='int_rate',y='purpose',data=df)
```

Out[57]:

```
<matplotlib.axes._subplots.AxesSubplot at 0xb0181371c8>
```



In [ ]: