Univariate Analysis
The dataset is already download in .csv format

IMPORTING THE PACKAGE

In [1]:
```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
import seaborn as sns

import warnings
warnings.filterwarnings('ignore')
```

Load the dataset

In [2]:
```
df=pd.read_csv("C:\loan_prediction.csv")
```

In [3]:
```
df.head()
```

O...

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	Loan
0	LP001002	Male	No	0	Graduate	No	5849	0.0	
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	
4	LP001008	Male	No	0	Graduate	No	6000	0.0	

In [4]:
```
numerical_features = df.select_dtypes(include = [np.number]).columns
categorical_features = df.select_dtypes(include = [np.object]).columns
numerical_features
```

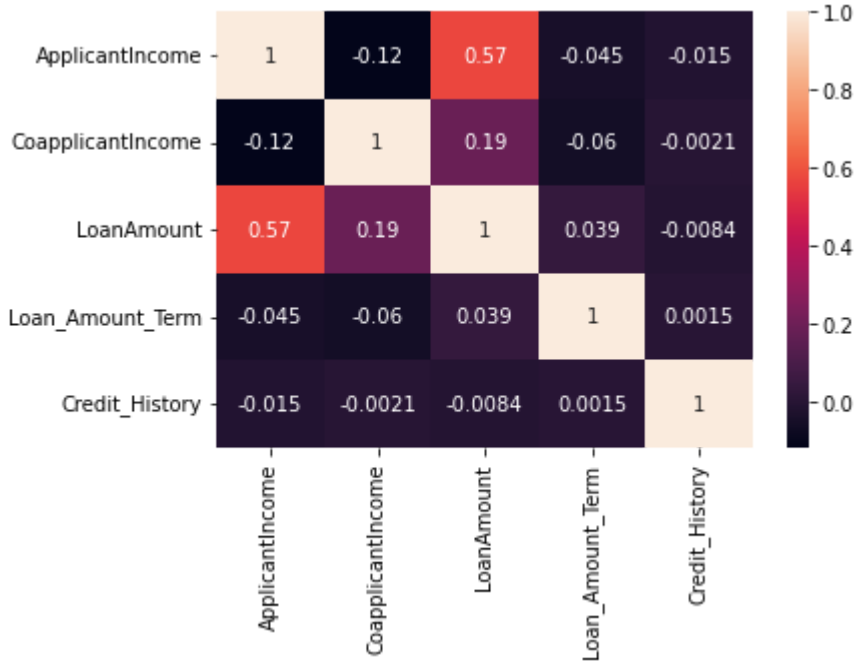
Out[4]:
```
Index(['ApplicantIncome', 'CoapplicantIncome', 'LoanAmount',
       'Loan_Amount_Term', 'Credit_History'],
      dtype='object')
```

In [5]:
```
categorical_features
```

Out[5]:
```
Index(['Loan_ID', 'Gender', 'Married', 'Dependents', 'Education',
       'Self_Employed', 'Property_Area', 'Loan_Status'],
      dtype='object')
```

In [6]:
```
df.corr()
#plotting the correlation
plt.figure(1)
sns.heatmap(df.corr(), annot = True)
```

Out[6]:



In [7]:
```
data=df
data = np.random.randint(low=1,high=100,size=(10,10))
print(data)
```

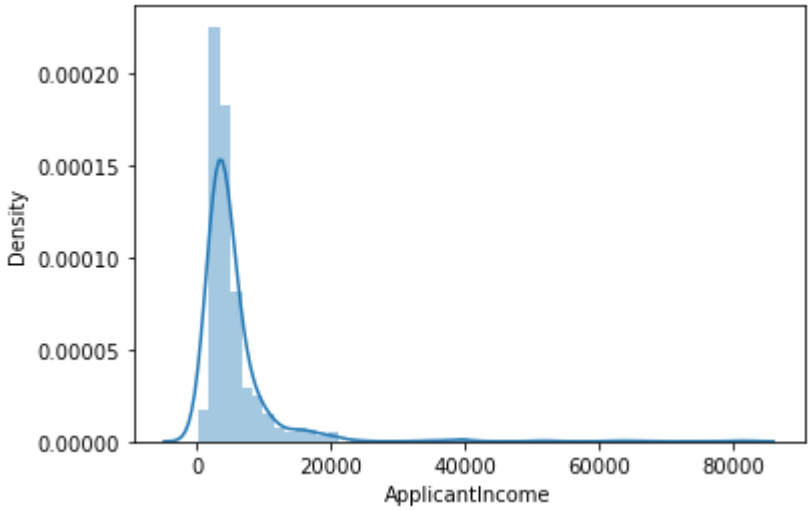
```
[[65 32  2 48 51 46 23 48 99 47]
 [78 59 71  4 68 18 93 25 58 75]
 [20 21 92 96 29 62 24 97 85  3]
 [ 5 55 40 40 48 89 40 40 23 99]
 [21 10 37 53 38 19 38 87 63 47]
 [47 45 23 31 42  4 14 96 65 88]
 [71 85  8 42 25 72 42 69 24 39]
 [79 50 98 57 96 23 27 77 22 52]
 [92  9 70 12 13 15  7 46 36 62]
 [ 7 24 26 33 51 99 33 48  3 89]]
```

In [ ]:

Perform Visualizations

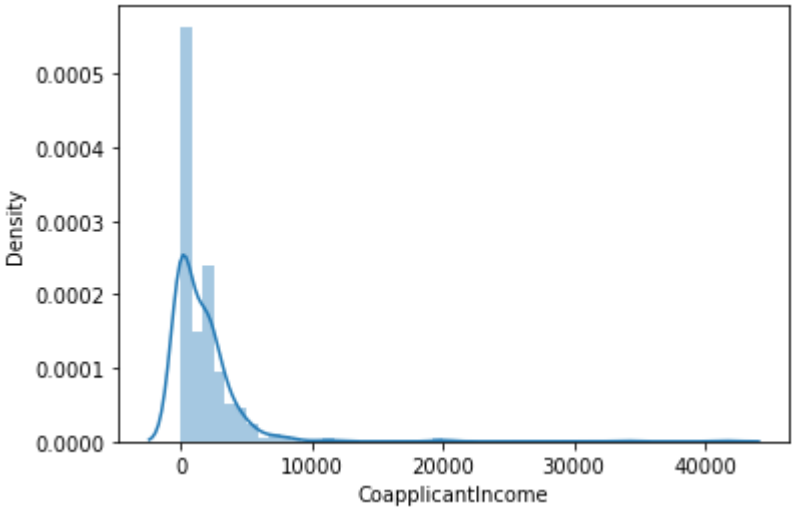
In [8]:
```
sns.distplot(df['ApplicantIncome'])
```

Out[8]:



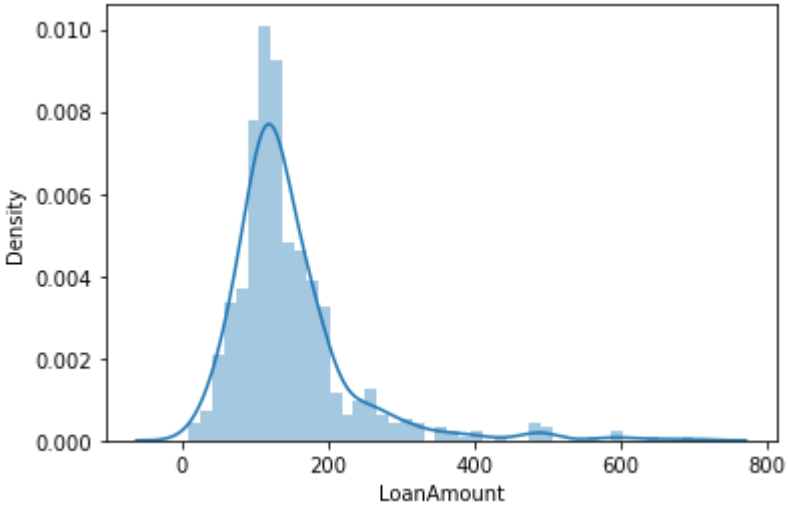
In [9]:
```
sns.distplot(df['CoapplicantIncome'])
```

Out[9]:



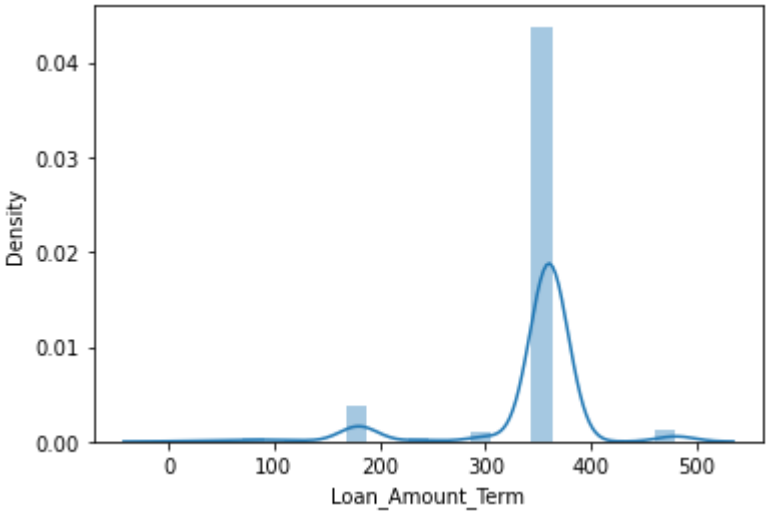
In [10]:
```
sns.distplot(df['LoanAmount'])
```

Out[10]:



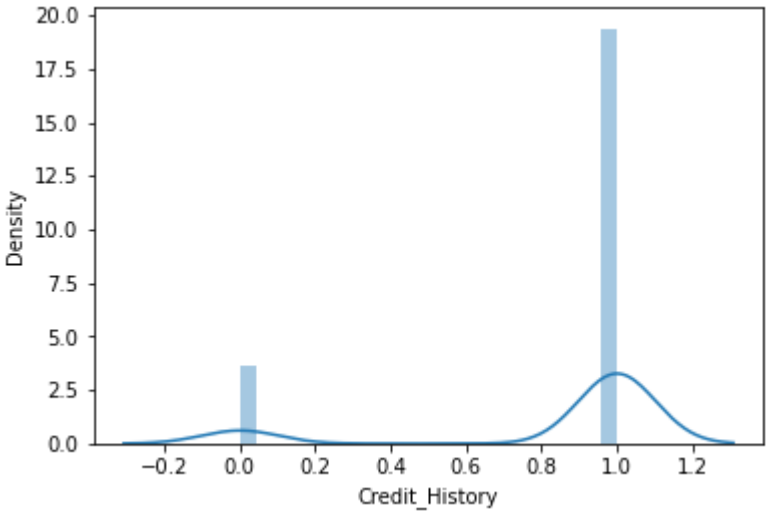
In [11]:
```
sns.distplot(df['Loan_Amount_Term'])
```

Out[11]:



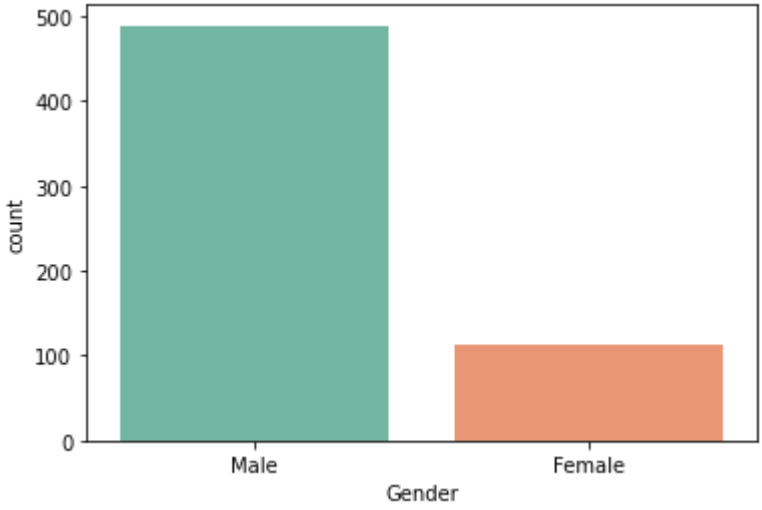
In [12]:
```
sns.distplot(df['Credit_History'])
```

Out[12]:



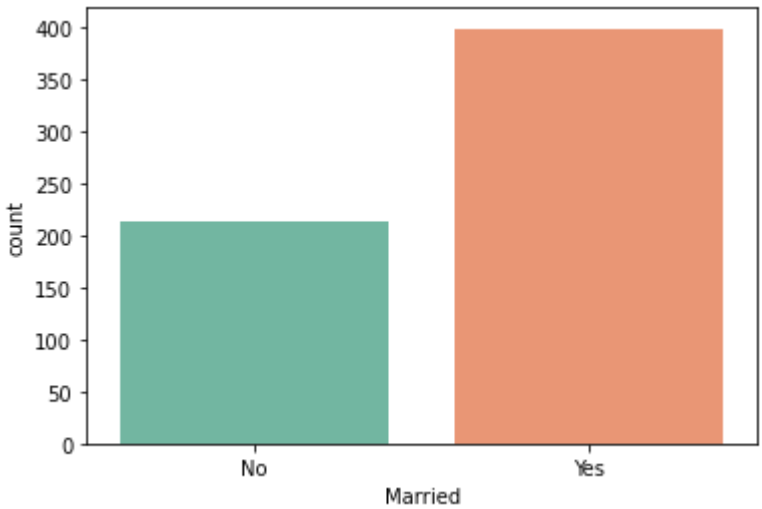
In [13]:
```
sns.countplot(df.Gender , data = df, palette = 'Set2')
```

Out[13]:



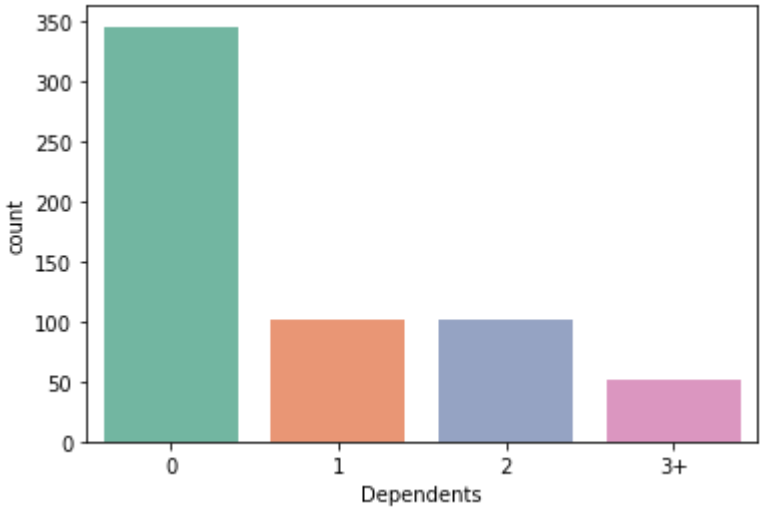
In [14]:
```
sns.countplot(df.Married , data = df, palette = 'Set2')
```

Out[14]:



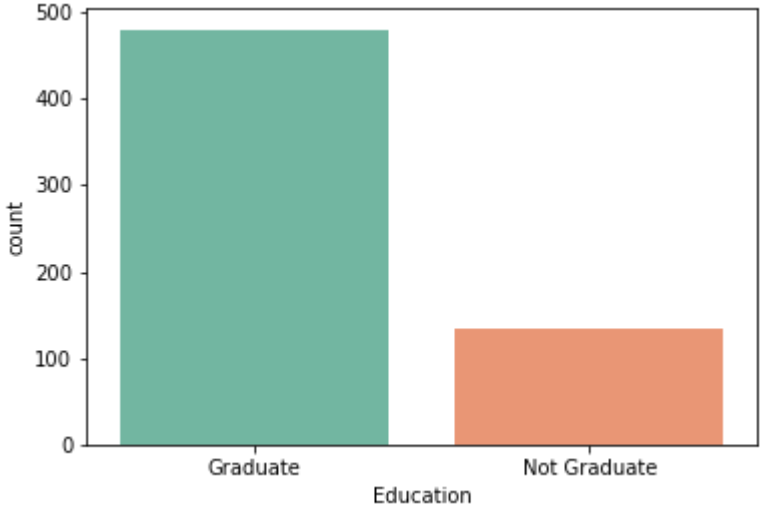
In [15]:
```
sns.countplot(df.Dependents , data = df, palette = 'Set2')
```

Out[15]:



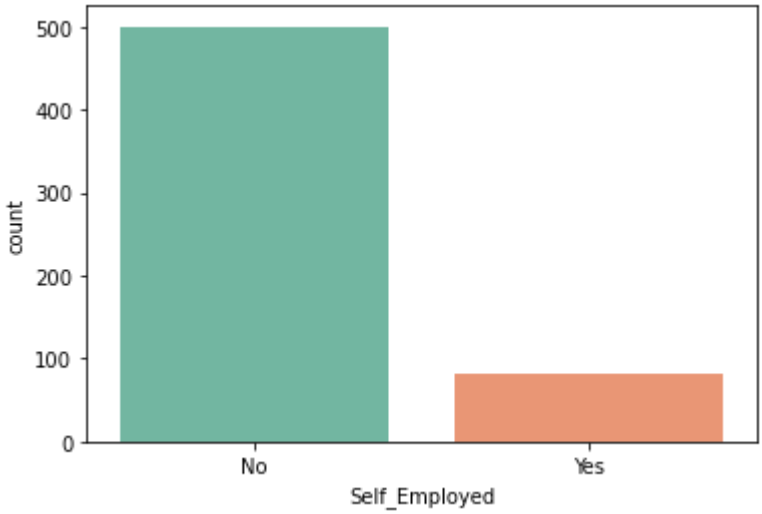
In [16]:
```
sns.countplot(df.Education , data = df, palette = 'Set2')
```

Out[16]:



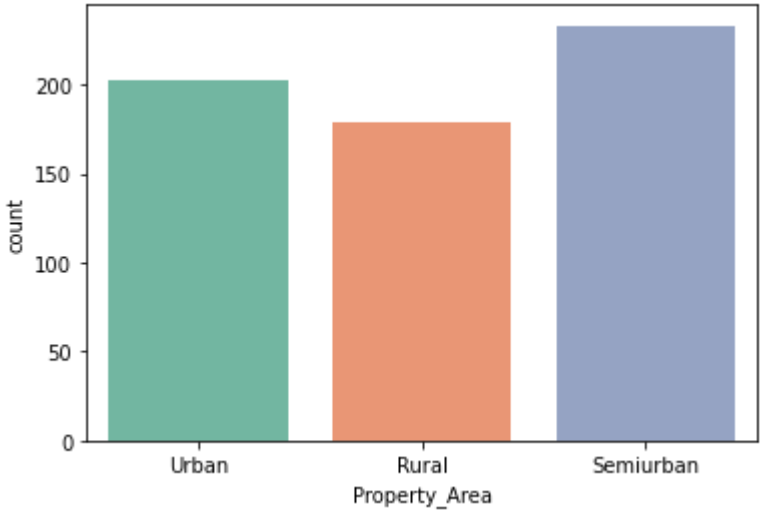
In [17]:
```
sns.countplot(df.Self_Employed , data = df, palette = 'Set2')
```

Out[17]:



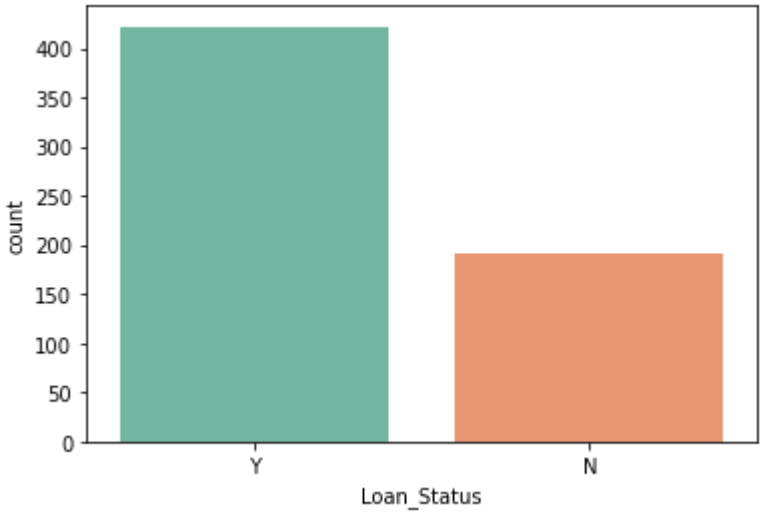
In [18]:
```
sns.countplot(df.Property_Area , data = df, palette = 'Set2')
```

Out[18]:



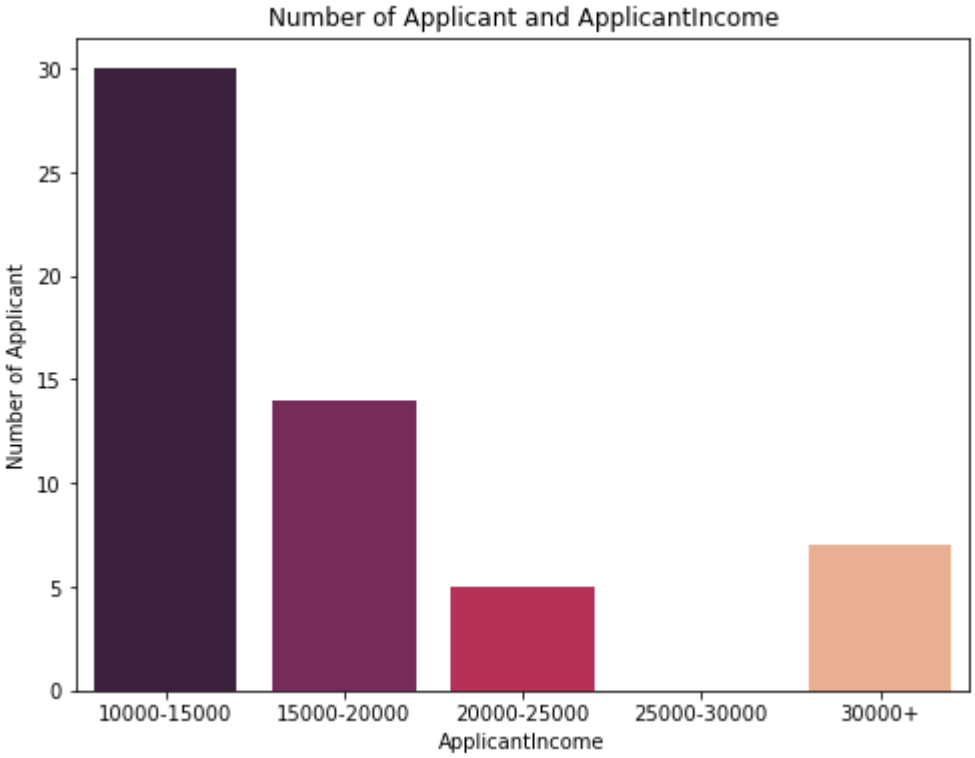
In [19]:
```
sns.countplot(df.Loan_Status , data = df, palette = 'Set2')
```

Out[19]:



I...
```
#Plotting the features of the dataset to see the correlation between Number of Applicant an
ApplicantIncome15000_10000 = df.ApplicantIncome[(df.ApplicantIncome <= 15000) & (df.Applica
```

```
ApplicantIncome20000_15000 = df.ApplicantIncome[(df.ApplicantIncome <= 20000) & (df.Applica
ApplicantIncome25000_20000 = df.ApplicantIncome[(df.ApplicantIncome <= 25000) & (df.Applica
ApplicantIncome30000_25000 = df.ApplicantIncome[(df.ApplicantIncome <= 30000) & (df.Applica
ApplicantIncome30000above = df.ApplicantIncome[df.ApplicantIncome >= 30000]
x = ["10000-15000","15000-20000","20000-25000","25000-30000","30000+"]
y = [len(ApplicantIncome15000_10000.values),len(ApplicantIncome20000_15000.values),len(Appl
plt.figure(figsize=(8,6))
sns.barplot(x=x, y=y, palette="rocket")
plt.title("Number of Applicant and ApplicantIncome")
plt.xlabel("ApplicantIncome")
plt.ylabel("Number of Applicant")
plt.show()
```



In [ ]: