

Name: **Vidul Bhosale**

Div: **A**

Roll No: **15**

Moodle ID: **20102152**

Machine Learning Experiment No: **4**

```
from google.colab import drive
drive.mount('/content/drive')
```

Mounted at /content/drive

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

import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

df = pd.read_csv('/content/BankDataset.csv')

df.head()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome
0	LP001002	Male	No	0	Graduate	No	5849
1	LP001003	Male	Yes	1	Graduate	No	4583
2	LP001005	Male	Yes	0	Graduate	Yes	3000
3	LP001006	Male	Yes	0	Not Graduate	No	2583

```
df.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Loan_ID                614 non-null   object
1   Gender                 601 non-null   object
2   Married                611 non-null   object
3   Dependents             599 non-null   object
4   Education              614 non-null   object
5   Self_Employed          582 non-null   object
6   ApplicantIncome        614 non-null   int64
7   CoapplicantIncome      614 non-null   float64
8   LoanAmount             592 non-null   float64
9   Loan_Amount_Term       600 non-null   float64
10  Credit_History         564 non-null   float64
11  Property_Area          614 non-null   object
12  Loan_Status            614 non-null   object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.5+ KB
```

```
df.shape

(614, 13)
```

```
df.isnull().sum()

Loan_ID                0
Gender                 13
Married                3
Dependents            15
Education              0
Self_Employed         32
ApplicantIncome        0
CoapplicantIncome      0
LoanAmount            22
Loan_Amount_Term       14
```

```

Credit_History      50
Property_Area        0
Loan_Status          0
dtype: int64 df['LoanAmount']
=
df['LoanAmount'].fillna(df['L
oanAmount'].mean())
df['Credit_History'] =
df['Credit_History'].fillna(d
f['Credit_History'].median())

df.isnull().sum()

```

```

Loan_ID      0
Gender       13
Married      3
Dependents   15
Education    0
Self_Employed 32
ApplicantIncome 0
CoapplicantIncome 0
LoanAmount   0
Loan_Amount_Term 14
Credit_History 0
Property_Area 0
Loan_Status  0
dtype: int64

```

```
df.dropna(inplace=True)
```

```
df.isnull().sum()
```

```

Loan_ID      0
Gender       0
Married      0
Dependents   0
Education    0
Self_Employed 0
ApplicantIncome 0
CoapplicantIncome 0
LoanAmount   0
Loan_Amount_Term 0
Credit_History 0
Property_Area 0
Loan_Status  0
dtype: int64

```

```
df.shape
```

```
(542, 13)
```

```
import matplotlib.pyplot as plt
import seaborn as sns
```

```
plt.figure(figsize=(20, 15))
sns.set(font_scale=1.5)
```

```
plt.subplot(3,3,1) sns.countplot(x=df['Gender'],
hue=df['Loan_Status'])
```

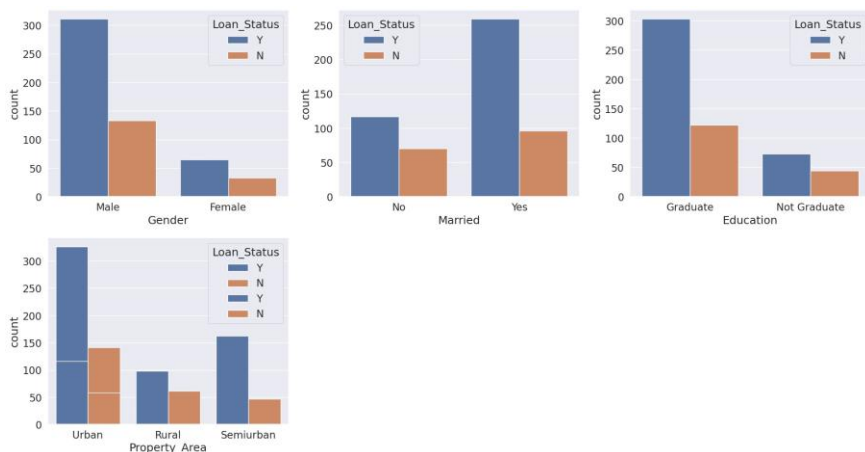
```
plt.subplot(3,3,2) sns.countplot(x=df['Married'],
hue=df['Loan_Status'])
```

```
plt.subplot(3,3,3) sns.countplot(x=df['Education'],
hue=df['Loan_Status'])
```

```
plt.subplot(3,3,4) sns.countplot(x=df['Self_Employed'],
hue=df['Loan_Status'])
```

```
plt.subplot(3,3,4) sns.countplot(x=df['Property_Area'],
hue=df['Loan_Status'])
```

```
plt.tight_layout()
plt.show()
```



```
df['Loan_Status'].replace('Y', 1, inplace=True)
```

```
df['Loan_Status'].replace('N', 0, inplace=True)
```

```
df['Loan_Status'].value_counts()
```

```
1    376
0    166
Name: Loan_Status, dtype: int64
```

```
df.Gender=df.Gender.map({'Male':1, 'Female':0})
```

```
df['Gender'].value_counts()
```

```
1    444
0     98
Name: Gender, dtype: int64
```

```
df.Married=df.Married.map({'Yes':1, 'No':0})
```

```
df['Married'].value_counts()
```

```
1    355
0    187
Name: Married, dtype: int64
```

```
df.Dependents=df.Dependents.map({'0':0, '1':1, '2':2, '3+':3})
```

```
df['Dependents'].value_counts()
```

```
0    309
1     94
2     94
3     45
Name: Dependents, dtype: int64
```

```
df.Education=df.Education.map({'Graduate':1, 'Not Graduate':0})
```

```
df['Education'].value_counts()
```

```
1    425
0    117
Name: Education, dtype: int64
```

```
df.Self_Employed=df.Self_Employed.map({'Yes':1, 'No':0})
```

```
df['Self_Employed'].value_counts()
```

```
0    467
1     75
Name: Self_Employed, dtype: int64
```

```
df.Property_Area=df.Property_Area.map({'Urban':2, 'Rural':0, 'Semiurban':1 })
```

```
df['Property_Area'].value_counts()
```

```
1    209
2    174
0    159
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome
0	LP001002	1	0	0	1	0	5849
1	LP001003	1	1	1	1	0	4583
2	LP001005	1	1	0	1	1	3000
3	LP001006	1	1	0	0	0	2583
4	LP001008	1	0	0	1	0	6000

[illegible]

```
178    1
151    1
518    0
101    1
..
585    0
200    1
404    0
484    1
304    1
```

```
Name: Loan_Status, Length: 163, dtype: int64
```

```
accuracy = metrics.accuracy_score(y_test, lr_prediction)
print('Logistic Regression accuracy =', accuracy)
```

```
Logistic Regression accuracy = 0.8098159509202454
```