

# Data Pre - Processing

## Checking For Null Values

<b>Date</b>	<b>6 Nov 2022</b>
<b>Team Id</b>	<b>PNT2022TMID41056</b>
<b>Project Name</b>	<b>Smart Lender- Applicant Credibility Prediction for Loan Approval</b>

- Let's find the shape of our dataset first, To find the shape of our data, df.shape method is used. To find the data type, df.info() function is used.

```
In [10]: data.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 614 entries, 0 to 613
Data columns (total 13 columns):
Loan_ID           614 non-null object
Gender            601 non-null object
Married           611 non-null object
Dependents        599 non-null object
Education         614 non-null object
Self_Employed     582 non-null object
ApplicantIncome   614 non-null int64
CoapplicantIncome 614 non-null float64
LoanAmount        592 non-null float64
Loan_Amount_Term  600 non-null float64
Credit_History    564 non-null float64
Property_Area     614 non-null object
Loan_Status       614 non-null object
dtypes: float64(4), int64(1), object(8)
memory usage: 62.4+ KB
```

- For checking the null values, df.isnull() function is used. To sum those null values we use .sum() function to it. From the below image we found that there are no null values present in our dataset. So we can skip the handling of the missing values step.

```
In [9]: import pandas as pd
data = pd.read_csv(r"C:\Users\ELCOT\Downloads\Dataset\loan_prediction.csv")
data.isnull().any()
```

```
Out[9]: Loan_ID           False
Gender              True
Married             True
Dependents          True
Education           False
Self_Employed       True
ApplicantIncome     False
CoapplicantIncome   False
LoanAmount          True
Loan_Amount_Term    True
Credit_History      True
Property_Area       False
Loan_Status         False
dtype: bool
```

From the above code of analysis, we can infer that columns such as gender, married, dependents, self-employed, loan amount, loan amount term, and credit history are having the missing values, we need to treat them in a required way.

```
In [16]: data['Gender']=data['Gender'].fillna(data['Gender'].mode()[0])
```

```
In [11]: data['Married']=data['Married'].fillna(data['Married'].mode()[0])
```

```
In [12]: data['Dependents']=data['Dependents'].fillna(data['Dependents'].mode()[0])
```

```
In [13]: data['Self_Employed']=data['Self_Employed'].fillna(data['Self_Employed'].mode()[0])
```

```
In [14]: data['LoanAmount']=data['LoanAmount'].fillna(data['LoanAmount'].mode()[0])
```

```
In [15]: data['Loan_Amount_Term']=data['Loan_Amount_Term'].fillna(data['Loan_Amount_Term'].mode()[0])
```

```
In [17]: data['Credit_History']=data['Credit_History'].fillna(data['Credit_History'].mode()[0])
```

We will fill the missing values in numeric data type using the mean value of that particular column and categorical data type using the most repeated value.