

```
!pip install kaggle
```


```
Requirement already satisfied: kaggle in /usr/local/lib/python3.10/dist-packages (1.6.14)  
Requirement already satisfied: six>=1.10 in /usr/local/lib/python3.10/dist-packages (from kaggle) (1.16.0)  
Requirement already satisfied: certifi>=2023.7.22 in /usr/local/lib/python3.10/dist-packages (from kaggle) (2024.6.2)  
Requirement already satisfied: python-dateutil in /usr/local/lib/python3.10/dist-packages (from kaggle) (2.8.2)  
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from kaggle) (2.31.0)  
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from kaggle) (4.66.4)  
Requirement already satisfied: python-slugify in /usr/local/lib/python3.10/dist-packages (from kaggle) (8.0.4)  
Requirement already satisfied: urllib3 in /usr/local/lib/python3.10/dist-packages (from kaggle) (2.0.7)  
Requirement already satisfied: bleach in /usr/local/lib/python3.10/dist-packages (from kaggle) (6.1.0)  
Requirement already satisfied: webencodings in /usr/local/lib/python3.10/dist-packages (from bleach->kaggle) (0.5.1)  
Requirement already satisfied: text-unidecode>=1.3 in /usr/local/lib/python3.10/dist-packages (from python-slugify->kaggle) (1.3)  
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->kaggle) (3.3.2)  
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->kaggle) (3.7)
```

```
from google.colab import files  
files.upload()
```

Graduate, , 6794, 528, 139, 360, 0, Urban\r\nLP001791, Male, Yes, 0, Graduate, Yes, 32000, 0, 550, 36
Graduate, No, 3276, 0, 90, 360, 1, Semiurban\r\nLP001817, Male, No, 0, Not
Graduate, Yes, 8703, 0, 199, 360, 0, Rural\r\nLP001818, Male, Yes, 1, Graduate, No, 4742, 717, 139, 3
Graduate, No, 2463, 2360, 117, 360, 0, Urban\r\nLP001855, Male, Yes, 2, Graduate, No, 4855, 0, 72, 36
Graduate, Yes, 1599, 2474, 125, 300, 1, Semiurban\r\nLP001862, Male, Yes, 2, Graduate, Yes, 4246, 4
Graduate, No, 7895, 0, 143, 360, 1, Rural\r\nLP001886, Male, No, 0, Graduate, No, 4150, 4256, 209, 36
Graduate, No, 2268, 0, 170, 360, 0, Semiurban\r\nLP001933, Male, No, 2, Not
Graduate, No, 1141, 2017, 120, 360, 0, Urban\r\nLP001943, Male, Yes, 0, Graduate, No, 3042, 3167, 13
Graduate, No, 4483, 0, 130, 360, 1, Rural\r\nLP001975, Male, Yes, 0, Graduate, No, 5225, 0, 143, 360,
Graduate, No, 2431, 1820, 110, 360, 0, Rural\r\nLP001999, Male, Yes, 2, Graduate, , 4912, 4614, 160,
Graduate, No, 2500, 3333, 131, 360, 1, Urban\r\nLP002009, Female, No, 0, Graduate, No, 2918, 0, 65, 3
Graduate, No, 4483, 0, 135, 360, , Semiurban\r\nLP002047, Male, Yes, 2, Not
Graduate, No, 4521, 1184, 150, 360, 1, Semiurban\r\nLP002056, Male, Yes, 2, Graduate, No, 9167, 0, 2
Graduate, No, 13083, 0, , 360, 1, Rural\r\nLP002059, Male, Yes, 2, Graduate, No, 7874, 3967, 336, 366
Graduate, , 3785, 2912, 180, 360, 0, Rural\r\nLP002070, Male, Yes, 3+, Not
Graduate, No, 2654, 1998, 128, 360, 0, Rural\r\nLP002077, Male, Yes, 1, Graduate, No, 10000, 2690, 4
Graduate, No, 2000, 1600, 115, 360, 1, Rural\r\nLP002099, Male, Yes, 2, Graduate, No, 2540, 700, 104
Graduate, No, 2855, 542, 90, 360, 1, Urban\r\nLP002111, Male, Yes, , Graduate, No, 3016, 1300, 100, 3
Graduate, No, 2038, 4027, 100, 360, 1, Rural\r\nLP002167, Female, No, 0, Graduate, No, 2362, 0, 55, 3
Graduate, No, 3754, 3719, 118, , 1, Rural\r\nLP002184, Male, Yes, 0, Not
Graduate, No, 2914, 2130, 150, 300, 1, Urban\r\nLP002186, Male, Yes, 0, Not
Graduate, No, 2747, 2458, 118, 36, 1, Semiurban\r\nLP002192, Male, Yes, 0, Graduate, No, 7830, 2183
Graduate, No, 3500, 2168, 149, 360, 1, Rural\r\nLP002245, Male, Yes, 2, Not
Graduate, No, 2896, 0, 80, 480, 1, Urban\r\nLP002253, Female, No, 1, Graduate, No, 5062, 0, 152, 300,
Graduate, No, 5180, 0, 125, 360, 0, Urban\r\nLP002294, Male, No, 0, Graduate, No, 14911, 14507, 130,
Graduate, No, 6166, 13983, 102, 360, 1, Rural\r\nLP002326, Male, Yes, 2, Not
Graduate, No, 2513, 1110, 107, 360, 1, Semiurban\r\nLP002329, Male, No, 0, Graduate, No, 4333, 0, 66
Graduate, No, 3844, 0, 105, 360, 1, Urban\r\nLP002339, Male, Yes, 0, Graduate, No, 3887, 1517, 105, 3
Graduate, No, 2107, 0, 64, 360, 1, Semiurban\r\nLP002355, , Yes, 0, Graduate, No, 3186, 3145, 150, 18
Graduate, Yes, 3943, 0, 64, 360, 1, Semiurban\r\nLP002376, Male, No, 0, Graduate, No, 2925, 0, 40, 18
Graduate, No, 2792, 2619, 171, 360, 1, Semiurban\r\nLP002420, Male, Yes, 0, Graduate, No, 2982, 155
Graduate, No, 3835, 1400, 112, 480, 0, Urban\r\nLP002445, Female, No, 1, Not
Graduate, No, 3854, 3575, 117, 360, 1, Rural\r\nLP002450, Male, Yes, 2, Graduate, No, 5833, 750, 49,
Graduate, No, 1635, 2444, 99, 360, 1, Urban\r\nLP002482, Female, No, 0, Graduate, Yes, 3333, 3916, 2
Graduate, , 3634, 910, 176, 360, 0, Semiurban\r\nLP002553, , No, 0, Graduate, No, 29167, 0, 185, 360,
Graduate, No, 9000, 0, 122, 360, 1, Rural\r\nLP002570, Female, Yes, 2, Graduate, No, 10000, 11666, 4
Graduate, No, 2157, 2730, 140, 360, , Rural\r\nLP002584, Male, No, 0, Graduate, , 1972, 4347, 106, 36
Graduate, No, 3271, 0, 90, 360, 1, Rural\r\nLP002609, Female, Yes, 0, Graduate, No, 2241, 2000, 88, 3
Graduate, , 1792, 2565, 128, 360, 1, Urban\r\nLP002612, Female, Yes, 0, Graduate, No, 2666, 0, 84, 48
Graduate, , 3808, 0, 83, 360, 1, Rural\r\nLP002635, Female, Yes, 2, Not
Graduate, No, 3729, 0, 117, 360, 1, Semiurban\r\nLP002639, Male, Yes, 2, Graduate, No, 4120, 0, 128,
Graduate, Yes, 570, 2125, 68, 360, 1, Rural\r\nLP002711, Male, Yes, 0, Graduate, No, 2600, 700, 96, 3
Graduate, No, 2733, 1083, 180, 360, , Semiurban\r\nLP002721, Male, Yes, 2, Graduate, Yes, 7500, 0, 1
Graduate, No, 3859, 0, 121, 360, 1, Rural\r\nLP002744, Male, Yes, 1, Graduate, No, 6825, 0, 162, 360,
Graduate, No, 1700, 2900, 67, 360, 0, Urban\r\nLP002775, , No, 0, Not
Graduate, No, 4768, 0, 125, 360, 1, Rural\r\nLP002781, Male, No, 0, Graduate, No, 3083, 2738, 120, 36
Graduate, No, 1647, 1762, 181, 360, 1, Urban\r\nLP002790, Male, Yes, 3+, Graduate, No, 3400, 0, 80, 1
Graduate, , 2600, 618, 122, 360, 1, Semiurban\r\nLP002805, Male, Yes, 2, Graduate, No, 5041, 700, 15
Graduate, No, 3621, 2717, 171, 360, 1, Urban\r\nLP002843, Female, Yes, 0, Graduate, No, 4709, 0, 113
Graduate, No, 3015, 2000, 145, 360, , Urban\r\nLP002856, Male, Yes, 0, Graduate, No, 2292, 1558, 115
Graduate, No, 3522, 0, 81, 180, 1, Rural\r\nLP002870, Male, Yes, 1, Graduate, No, 4700, 0, 80, 360, 1,
Graduate, No, 2868, 0, 70, 360, 1, Urban\r\nLP002890, Male, Yes, 2, Not
Graduate, No, 3418, 1380, 135, 360, 1, Urban\r\nLP002891, Male, Yes, 0, Graduate, Yes, 2500, 296, 13
Graduate, No, 5316, 187, 158, 180, 0, Semiurban\r\nLP002932, Male, Yes, 3+, Graduate, No, 7603, 121
Graduate, No, 3132, 0, 76, 360, , Rural\r\nLP002962, Male, No, 0, Graduate, No, 4000, 2667, 152, 360,
Graduate, Yes, 4009, 1777, 113, 360, 1, Urban\r\nLP002975, Male, Yes, 0, Graduate, No, 4158, 709, 11

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

```
df = pd.read_csv('loan prediction.csv')
df.head()
```




	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome
0	LP001015	Male	Yes	0	Graduate	No	5720	
1	LP001022	Male	Yes	1	Graduate	No	3076	
2	LP001031	Male	Yes	2	Graduate	No	5000	
3	LP001035	Male	Yes	2	Graduate	No	2340	
4	LP001051	Male	No	0	Not Graduate	No	3276	

Next steps:

Generate code with df

 View recommended plots

```
#drop the coloumn Loan_ID
df.drop('Loan_ID',axis=1,inplace=True)
df.head()
```



	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome
0	Male	Yes	0	Graduate	No	5720	
1	Male	Yes	1	Graduate	No	3076	
2	Male	Yes	2	Graduate	No	5000	
3	Male	Yes	2	Graduate	No	2340	
4	Male	No	0	Not Graduate	No	3276	


Next steps:

Generate code with df

 View recommended plots

```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.linear_model import Ridge,Lasso
from sklearn.preprocessing import StandardScaler
```

```
#check for null values
df.isnull().sum()
```




Gender	11
Married	0
Dependents	10
Education	0
Self_Employed	23
ApplicantIncome	0
CoapplicantIncome	0

```
LoanAmount      5
Loan_Amount_Term 6
Credit_History  29
Property_Area    0
dtype: int64
```


```
#drop the null values
df.dropna(inplace=True)
```

```
#check for null values
df.isnull().sum()
```



```
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
dtype: int64
```

```
df.head()
```




	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome
0	Male	Yes	0	Graduate	No	5720	
1	Male	Yes	1	Graduate	No	3076	
2	Male	Yes	2	Graduate	No	5000	
4	Male	No	0	Not Graduate	No	3276	
5	Male	Yes	0	Not Graduate	Yes	2165	

Next steps:

[Generate code with df](#)

 [View recommended plots](#)

```
#drop the coloumn Loan_ID
df.drop('Gender',axis=1,inplace=True)
df.head()
```



	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	Loan_Amount_Term
0	Yes	0	Graduate	No	5720	0	
1	Yes	1	Graduate	No	3076	1500	
2	Yes	2	Graduate	No	5000	1800	
4	No	0	Not Graduate	No	3276	0	
5	Yes	0	Not Graduate	Yes	2165	3422	

Next steps:

[Generate code with df](#)

 [View recommended plots](#)

```
#drop all the categorical data columns
df.drop('Married',axis=1,inplace=True)
df.head()
```



	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount
0	0	Graduate	No	5720	0	110.0
1	1	Graduate	No	3076	1500	126.0
2	2	Graduate	No	5000	1800	208.0
4	0	Not Graduate	No	3276	0	78.0
5	0	Not Graduate	Yes	2165	3422	152.0

Next steps:

Generate code with df

View recommended plots

```
df.drop('Education',axis=1,inplace=True)
df.head()
```



	Dependents	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amou
0	0	No	5720	0	110.0	
1	1	No	3076	1500	126.0	
2	2	No	5000	1800	208.0	
4	0	No	3276	0	78.0	
5	0	Yes	2165	3422	152.0	

Next steps:

Generate code with df

View recommended plots

```
df.drop('Property_Area',axis=1,inplace=True)
df.head()
```



	Dependents	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amou
0	0	No	5720	0	110.0	
1	1	No	3076	1500	126.0	
2	2	No	5000	1800	208.0	
4	0	No	3276	0	78.0	
5	0	Yes	2165	3422	152.0	

Next steps:

Generate code with df

View recommended plots

```
df.drop('Self_Employed',axis=1,inplace=True)
df.head()
```



	Dependents	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit
0	0	5720	0	110.0	360.0	
1	1	3076	1500	126.0	360.0	
2	2	5000	1800	208.0	360.0	
4	0	3276	0	78.0	360.0	
5	0	2165	3422	152.0	360.0	

Next steps:

Generate code with df

View recommended plots

```
#drop the coloumn Loan_ID
df.drop('Dependents',axis=1,inplace=True)
df.head()
```

	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History
0	5720	0	110.0	360.0	1.0
1	3076	1500	126.0	360.0	1.0
2	5000	1800	208.0	360.0	1.0
4	3276	0	78.0	360.0	1.0
Next 5	2166	2166	2166	360.0	1.0

```
#define features and target
```

```
features = df.drop('LoanAmount',axis=1)
```

```
target = df['LoanAmount']
```

```
# Split the data into training and testing sets
```

```
X_train, X_test, y_train, y_test = train_test_split(features, target, test_size=0.2, random_state=42)
```

```
print(X_train.shape)
```

```
print(X_test.shape)
```

```
print(y_train.shape)
```

```
print(y_test.shape)
```

```
(231, 4)
(58, 4)
(231,)
(58,)
```

```
from sklearn.linear_model import Ridge
```

```
#load the algorithm instance
```

```
ridge = Ridge()
```

```
#fit the data
```

```
ridge.fit(X_train,y_train)
```

```
▼ Ridge
Ridge()
```

```
#predict
```

```
y_pred = ridge.predict(X_test)
```

```
y_pred
```

```
array([149.57030431, 123.0912576 , 129.09344169, 126.94048061,
       116.10056925, 127.09736151, 110.42766465, 125.43187118,
       135.36180606, 122.83552123, 130.82902101, 124.39641315,
       119.89059996, 147.52639328, 156.74686919, 115.8664365 ,
       114.44664339, 128.82905817, 141.79085953, 137.714415 ,
       104.21636944, 128.51768098, 126.02081778, 146.11491646,
       172.99250775, 143.51065317, 128.82984621, 147.67601653,
       131.29301769, 142.5568093 , 146.38318264, 139.53935152,
       171.2564179 , 120.68038684, 144.08922978, 115.15176539,
       136.57190917, 129.4927024 , 143.47223791, 157.80512199,
```