

screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

project

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all

RAM Disk

#Predicting Home Loan Approval

```
[2] ✓ 1s
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import warnings
warnings.filterwarnings('ignore')

[3] ✓ 1s
from sklearn.model_selection import train_test_split
from sklearn.preprocessing import LabelEncoder
from sklearn.linear_model import LogisticRegression
from sklearn.ensemble import RandomForestClassifier, GradientBoostingClassifier
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB
from xgboost import XGBClassifier
from sklearn.metrics import accuracy_score, precision_score, confusion_matrix

[4] ✓ 0s
df = pd.read_csv('/content/loan_sanction_train.csv')
df.head()
```

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	

Variables Terminal 5:43 PM Python 3

Screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

File Edit View Insert Runtime Tools Help

Commands + Code + Text ▶ Run all

RAM Disk

Share M

3 LP001006 Male Yes 0 Not Graduate No 2583 2358.0 120.0 360.0 1.0

4 LP001008 Male No 0 Graduate No 6000 0.0 141.0 360.0 1.0

Next steps: Generate code with df New interactive sheet

[5] df.sample(3)

	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Propo
534	LP002731	Female	No	0	Not Graduate	Yes	18165	0.0	125.0	360.0	1.0	
6	LP001013	Male	Yes	0	Not Graduate	No	2333	1516.0	95.0	360.0	1.0	
382	LP002231	Female	No	0	Graduate	No	6000	0.0	156.0	360.0	1.0	

[6] df.shape  
(614, 13)

[7] df.isnull().sum()

Variables Terminal 5:43 PM Python 3

Top Stories Leaders across p... Search

7:12 PM 12/13/2025

screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

project

File Edit View Insert Runtime Tools Help

Commands + Code + Text ▶ Run all

RAM Disk

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

[8] df.duplicated().sum()  
0

np.int64(0)

{ } Variables

Terminal



✓ 5:43 PM Python 3

6 Top Stories  
Leaders across p...



Search



7:12 PM  
12/13/2025



co project ⭐

File Edit View Insert Runtime Tools Help

Commands + Code ▾ + Text ▾ Run all ▾

RAM Disk

```
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.describe()
```

1 to 8 of 8 entries						<a href="#">Filter</a>	<a href="#">?</a>
index	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History		
count	614.0	614.0	592.0	600.0			564.0
mean	5403.459283387622	1621.2457980271008	146.41216216216216	342.0			0.8421985815602837
std	6109.041673387174	2926.2483692241917	85.58732523570545	65.12040985461256			0.3648783192364049
min	150.0	0.0	9.0	12.0			0.0

{ } Variables



✓ 5:43 PM Python 3



File Edit View Insert Runtime Tools Help

Commands + Code + Text ▶ Run all

RAM Disk

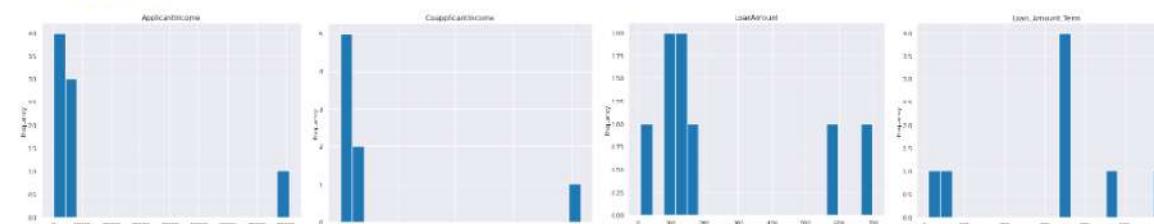
25%	2877.5	0.0	100.0	360.0	1.0
50%	3812.5	1188.5	128.0	360.0	1.0
75%	5795.0	2297.25	168.0	360.0	1.0
max	81000.0	41667.0	700.0	480.0	1.0

Show 25 per page

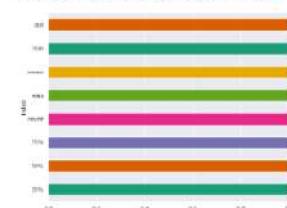


Like what you see? Visit the [data table notebook](#) to learn more about interactive tables.

### Distributions



### Categorical distributions



### 2-d distributions



Variables

Terminal



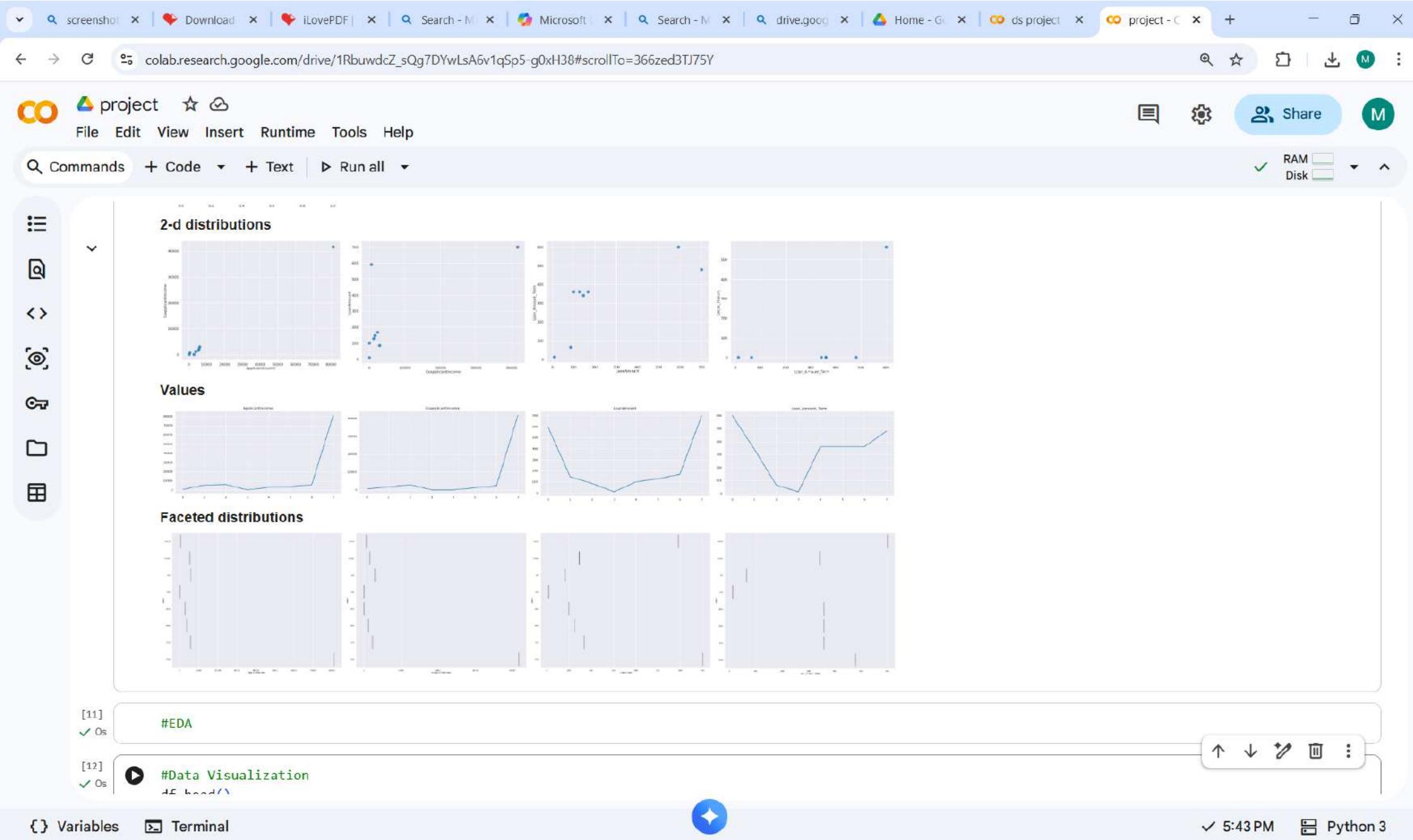
5:43 PM Python 3

28°C  
Partly cloudy

Search



7:13 PM  
12/13/2025



screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all RAM Disk

#EDA

[12] #Data Visualization df.head()

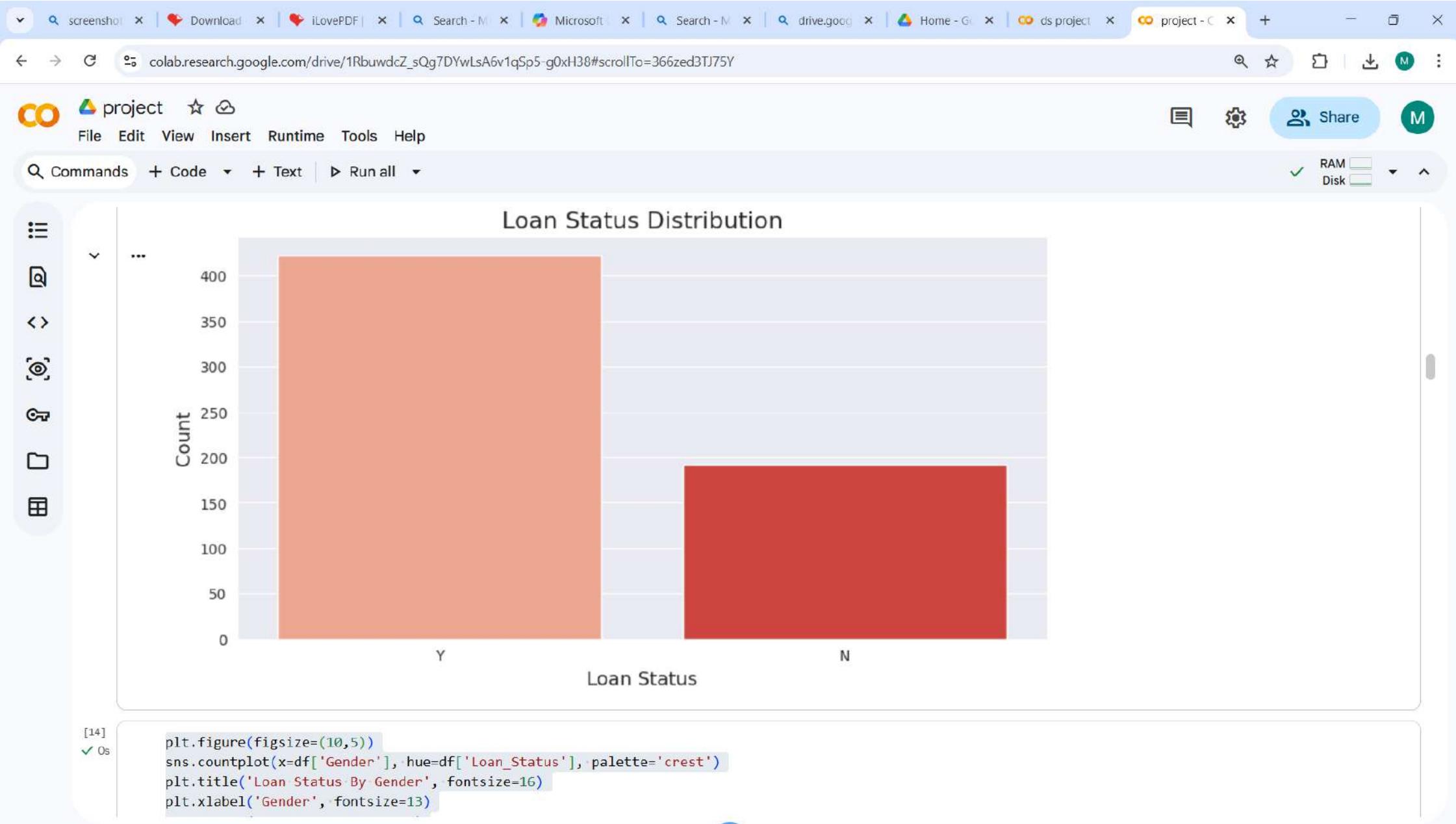
	Loan_ID	Gender	Married	Dependents	Education	Self_Employed	ApplicantIncome	CoapplicantIncome	LoanAmount	Loan_Amount_Term	Credit_History	Property
0	LP001002	Male	No	0	Graduate	No	5849	0.0	NaN	360.0	1.0	
1	LP001003	Male	Yes	1	Graduate	No	4583	1508.0	128.0	360.0	1.0	
2	LP001005	Male	Yes	0	Graduate	Yes	3000	0.0	66.0	360.0	1.0	
3	LP001006	Male	Yes	0	Not Graduate	No	2583	2358.0	120.0	360.0	1.0	
4	LP001008	Male	No	0	Graduate	No	6000	0.0	141.0	360.0	1.0	

Next steps: Generate code with df New interactive sheet

[13] sns.set\_style('darkgrid')  
plt.figure(figsize=(10,5))  
sns.countplot(x=df['Loan\_Status'], palette='Reds')  
plt.title('Loan Status Distribution', fontsize=16)  
plt.xlabel('Loan Status', fontsize=13)  
plt.ylabel('Count', fontsize=13)  
plt.show()

### Loan Status Distribution

Variables Terminal 5:43 PM Python 3



{ } Variables

Terminal



✓ 5:43 PM Python 3

28°C  
Partly cloudy



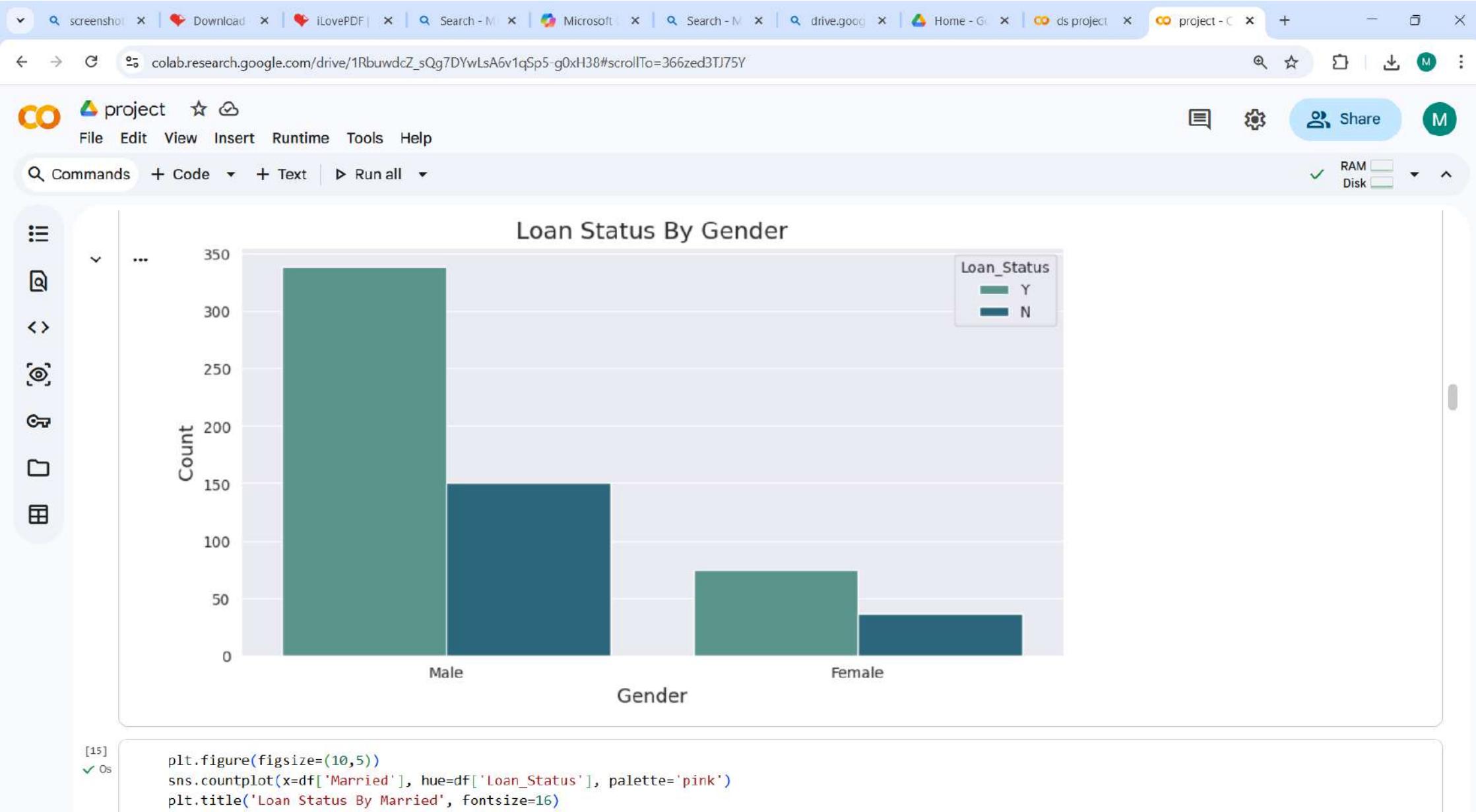
Search



ENG  
IN

7:13 PM

12/13/2025



{ } Variables

Terminal



✓ 5:43 PM Python 3

28°C  
Partly cloudy

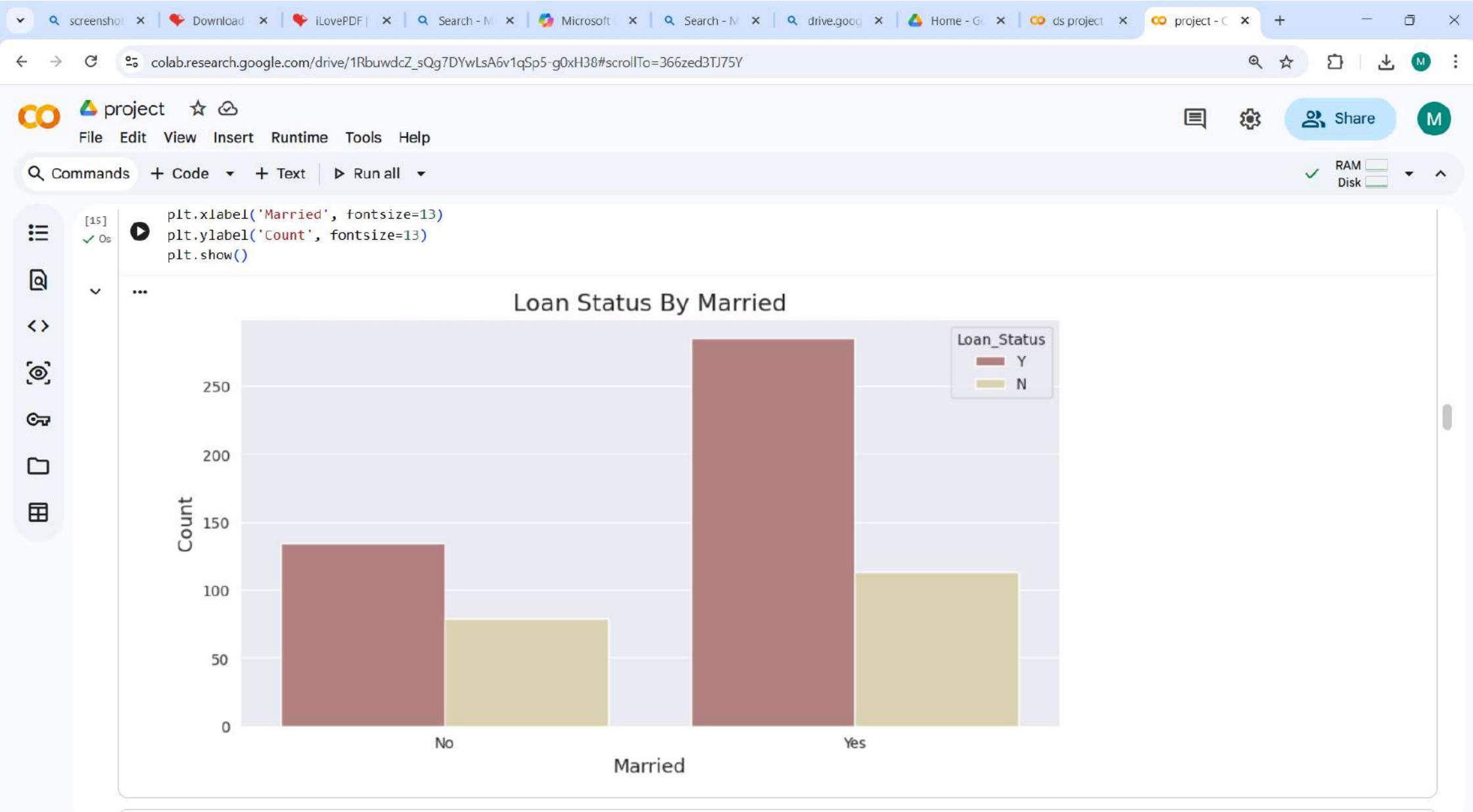
Search



ENG  
IN



7:13 PM  
12/13/2025



[16]

{ } Variables

Terminal



✓ 5:43 PM Python 3

28°C  
Partly cloudy

Search



7:13 PM

12/13/2025

screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

File Edit View Insert Runtime Tools Help

Commands + Code + Text ▶ Run all

✓ RAM Disk

[16] ✓ 0s

```
plt.figure(figsize=(10,5))
sns.countplot(x=df['Property_Area'], hue=df['Loan_Status'], palette='magma')
plt.title('Loan Status By Property Area', fontsize=16)
plt.xlabel('Property Area', fontsize=13)
plt.ylabel('Count', fontsize=13)
plt.show()
```

...

Loan Status By Property Area

The chart displays the count of loans categorized by property area and loan status. The Y-axis represents the count, ranging from 0 to 175. The X-axis categories are Urban, Rural, and Semiurban. For each category, there are two bars: a purple bar for 'Y' and a red bar for 'N'. The legend indicates that purple represents 'Y' and red represents 'N'.

Property Area	Loan_Status	Count
Urban	Y	135
	N	70
Rural	Y	110
	N	70
Semiurban	Y	180
	N	55

Variables Terminal

28°C Partly cloudy

Search

7:13 PM 12/13/2025

5:43 PM Python 3

screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

project

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all

RAM Disk

[17] ✓ 0s

```
plt.figure(figsize=(10,5))
sns.countplot(x=df['Education'], hue=df['Loan_Status'], palette='cividis')
plt.title('Loan Status By Education', fontsize=16)
plt.xlabel('Education', fontsize=13)
plt.ylabel('Count', fontsize=13)
plt.show()
```

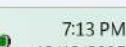
...

Loan Status By Education

The chart displays the count of loans for two education levels: Graduate and Not Graduate. For each level, there are two bars representing the Loan\_Status: Y (dark grey) and N (brown). The y-axis is labeled 'Count' and ranges from 0 to 350. The x-axis categories are 'Graduate' and 'Not Graduate'. The legend indicates that dark grey represents 'Y' and brown represents 'N'.

Education	Loan_Status	Count
Graduate	Y	340
	N	140
Not Graduate	Y	80
	N	55

1



✓ 5:43 PM Python 3

screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

File Edit View Insert Runtime Tools Help

Commands + Code + Text ▶ Run all

✓ RAM Disk

[18] `plt.figure(figsize=(10,5))  
sns.countplot(x=df['Self_Employed'], hue=df['Loan_Status'], palette='viridis_r')  
plt.title('Loan Status By Self Employed', fontsize=16)  
plt.xlabel('Self Employed', fontsize=13)  
plt.ylabel('Count', fontsize=13)  
plt.show()`

...

Loan Status By Self Employed

Self Employed	Loan_Status	Count
No	Y	~345
No	N	~155
Yes	Y	~55
Yes	N	~25

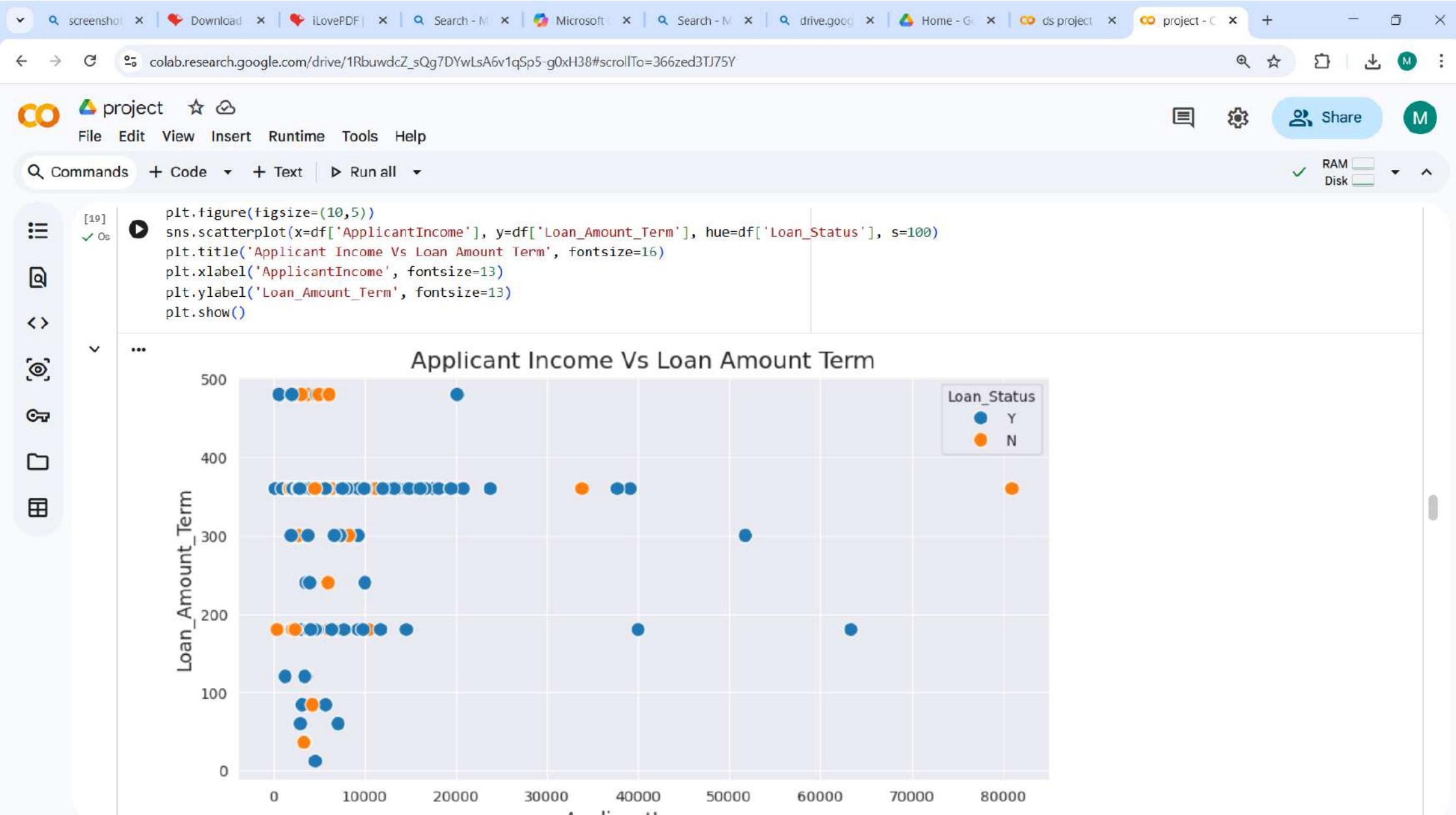
Variables Terminal

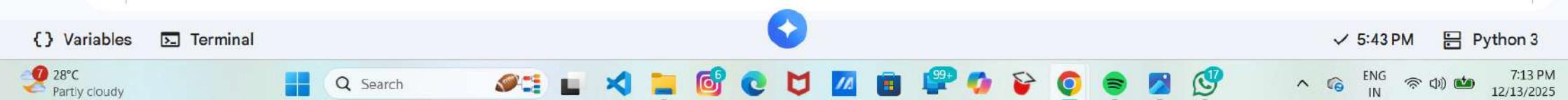
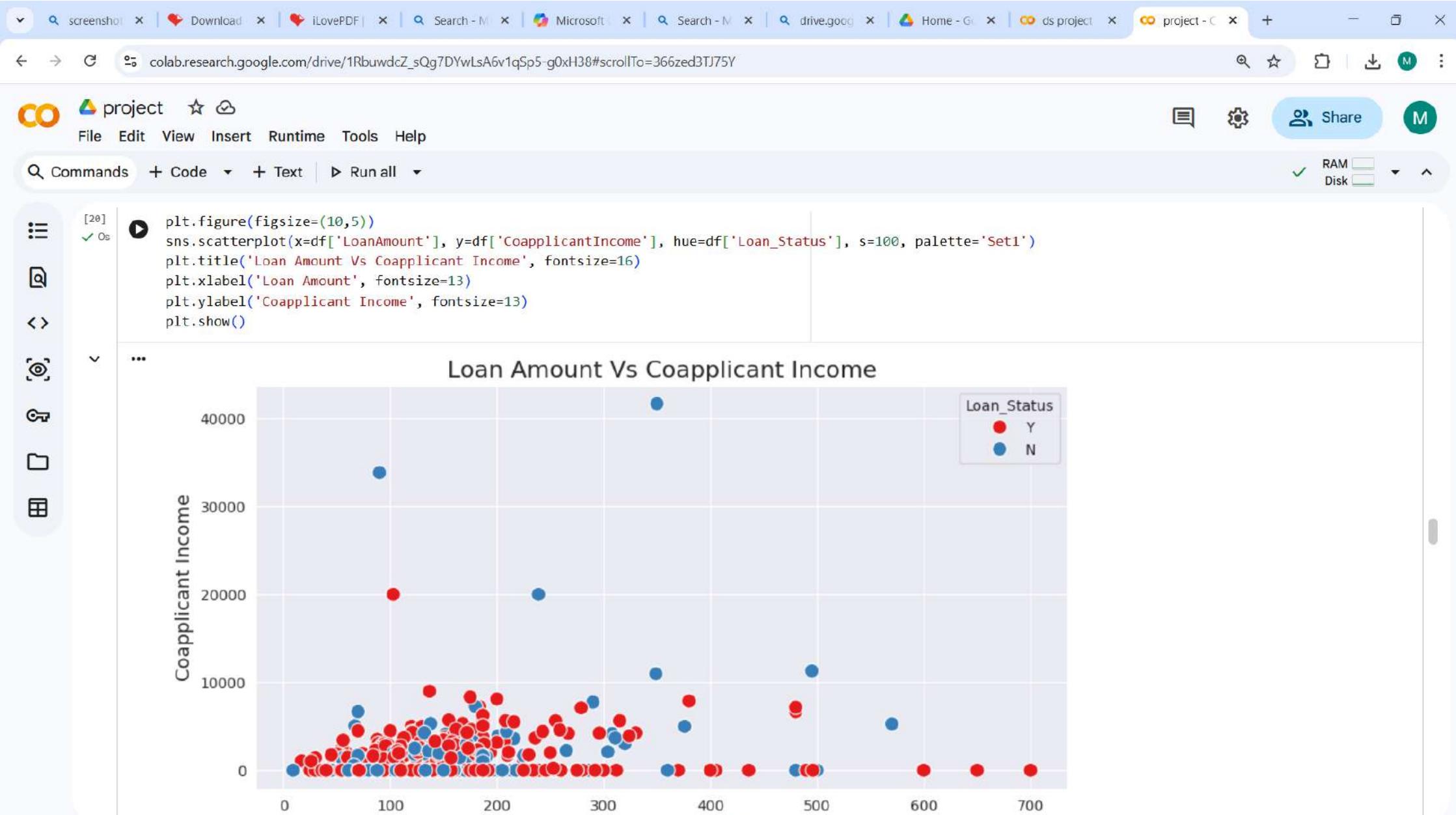
28°C Partly cloudy

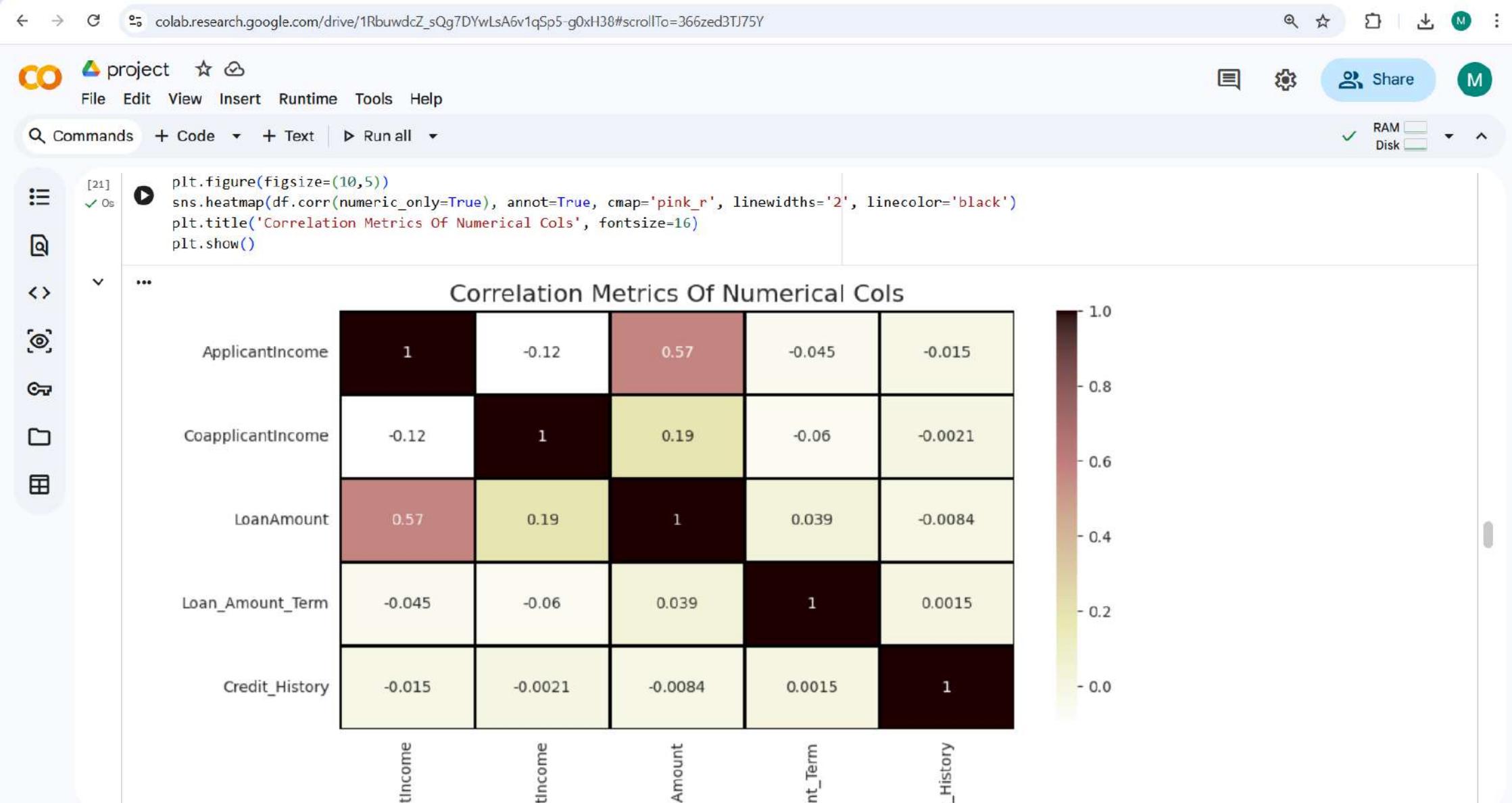
Search

7:13 PM 12/13/2025

5:43 PM Python 3







screenshot | Download | iLovePDF | Search - M | Microsoft | Search - M | drive.goog | Home - G | ds project | project - C

colab.research.google.com/drive/1RbuwdcZ\_sQg7DYwLsA6v1qSp5-g0xH38#scrollTo=366zed3TJ75Y

File Edit View Insert Runtime Tools Help

Commands + Code + Text Run all

RAM Disk

File Edit View Insert Runtime Tools Help

Applicant Coapplicant LoanA Loan\_Amount Credit\_

[22] 1s

```
plt.figure(figsize=(13,15))
for c, cols in enumerate(['ApplicantIncome', 'CoapplicantIncome', 'LoanAmount', 'Loan_Amount_Term',
                           'Credit_History']):
    plt.subplot(3, 2, c+1)
    sns.histplot(x=cols, data=df, color='blue', kde=True)
    plt.title(f"Distribution Of {cols}")
    plt.tight_layout(pad=4.0)

plt.show()
```

Distribution Of ApplicantIncome

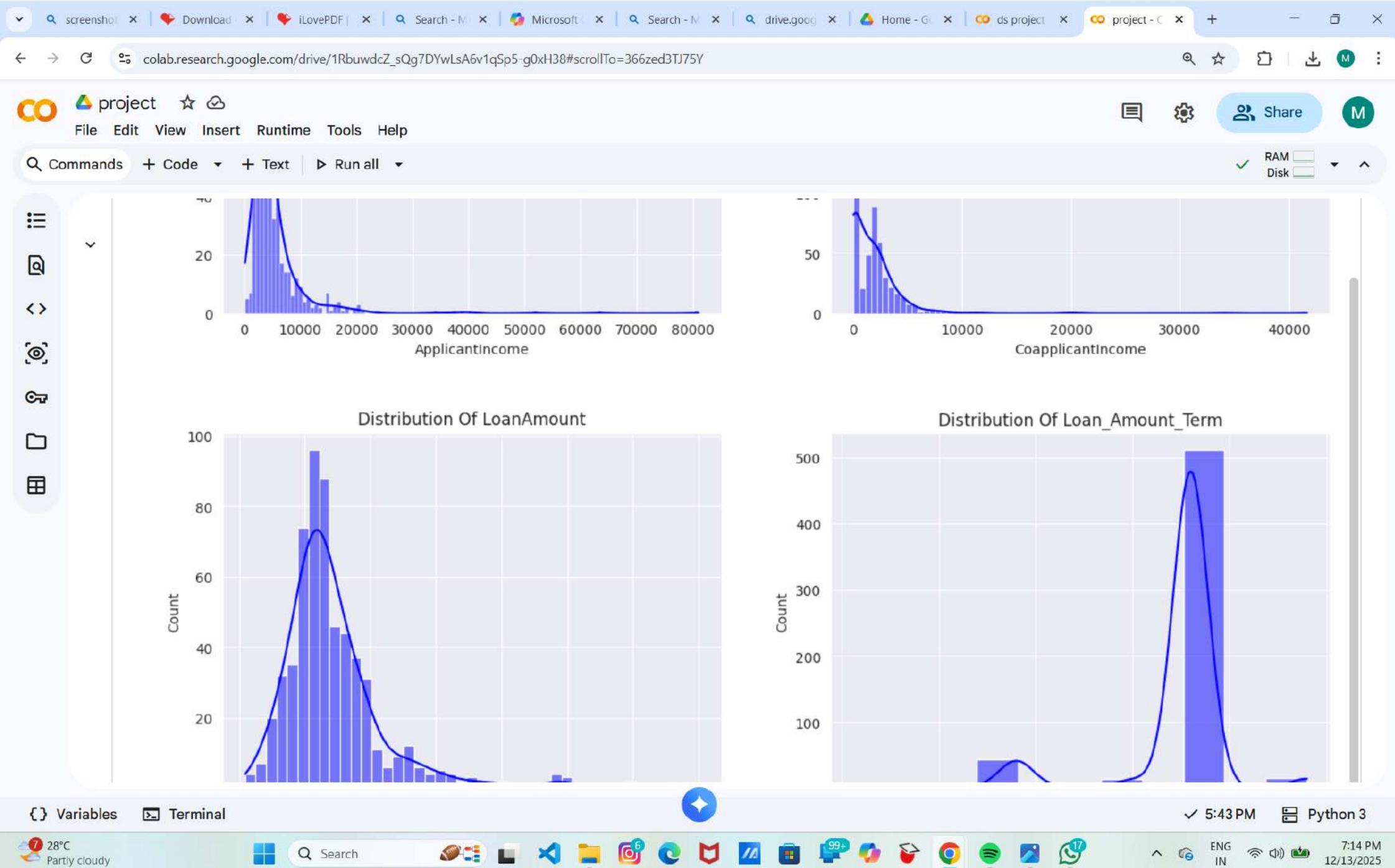
Distribution Of CoapplicantIncome

Variables Terminal

28°C Partly cloudy

Search

7:14 PM 12/13/2025





{ } Variables

Terminal



✓ 5:43 PM Python 3

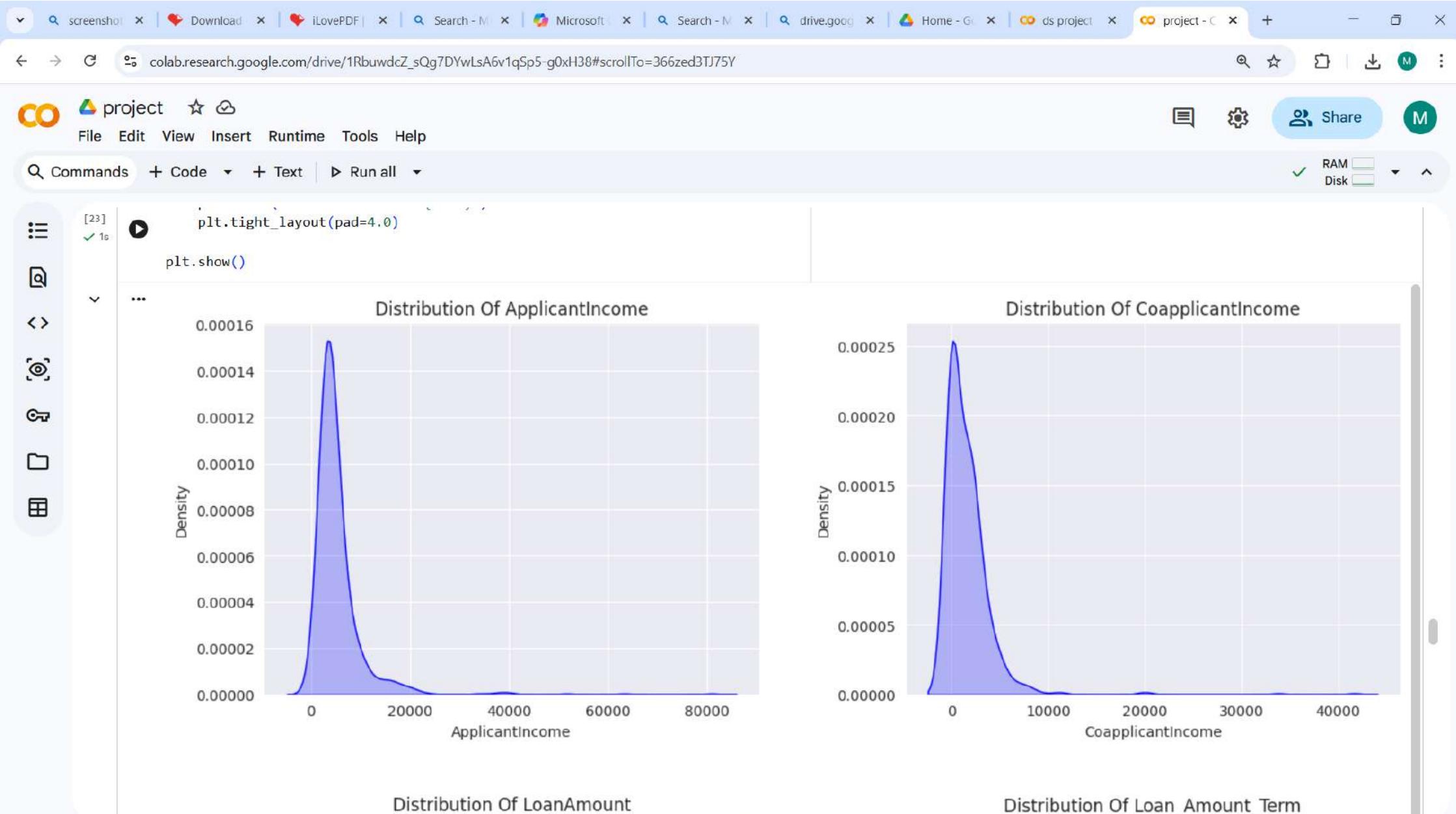
28°C  
Partly cloudy

Search



ENG  
IN

7:14 PM  
12/13/2025



Search



7:14 PM 12/13/2025