requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

src > Customer_Churn_Prediction.ipynb > M↓ Customer Churn Prediction > df.head()

Generate | + Code | + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline | ... | venv (Python 3.13.3)

[8] Python

```python
df.head()
```

[9] Python

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges	Total Long Distance Charges	Total Revenue	Customer Status	Cat
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.6	593.30	0.00	0	381.51	974.81	Stayed	
1	0003-MKNFE	Male	46	No	0	Glendale	91206	34.162515	-118.203869	0	...	Credit Card	-4.0	542.40	38.33	10	96.21	610.28	Stayed	
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.645672	-117.922613	0	...	Bank Withdrawal	73.9	280.85	0.00	0	134.60	415.45	Churned	Comp
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.0	1237.85	0.00	0	361.66	1599.51	Churned	Dissatisf
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.9	267.40	0.00	0	22.14	289.54	Churned	Dissatisf

5 rows × 38 columns

```python
df.head(5)
```

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS | JUPYTER | cmd

```
(c) Microsoft Corporation. All rights reserved.

C:\Users\supri\Downloads\Customer-Churn-Prediction-main1\Customer-Churn-Prediction-main>
```

requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

src > Customer_Churn_Prediction.ipynb > M↓ Customer Churn Prediction > df.head()

Generate + Code + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline | ··· venv (Python 3.13.3)

```
df.head(5)
```

[10] Python

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges	Total Long Distance Charges	Total Revenue	Customer Status	Cat
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.6	593.30	0.00	0	381.51	974.81	Stayed	
1	0003-MKNFE	Male	46	No	0	Glendale	91206	34.162515	-118.203869	0	...	Credit Card	-4.0	542.40	38.33	10	96.21	610.28	Stayed	
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.645672	-117.922613	0	...	Bank Withdrawal	73.9	280.85	0.00	0	134.60	415.45	Churned	Com
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.0	1237.85	0.00	0	361.66	1599.51	Churned	Dissatisf
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.9	267.40	0.00	0	22.14	289.54	Churned	Dissatisf

5 rows × 38 columns

Overviewing all the columns in the dataset

```
df.columns
```

[11] Python

```
Index(['Customer ID', 'Gender', 'Age', 'Married', 'Number of Dependents',
```

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS | JUPYTER | cmd


```
C:\Users\supri\Downloads\Customer-Churn-Prediction-main1\Customer-Churn-Prediction-main>
```

requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

src > Customer_Churn_Prediction.ipynb > Customer Churn Prediction > df.head()

Generate + Code + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline ... | venv (Python 3.13.3)

Creating a copy of the Dataset

[13]
```python
df1.head(7)
```
Python

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges	Total Long Distance Charges	Total Revenue	Customer Status	Cat
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.6	593.30	0.00	0	381.51	974.81	Stayed	
1	0003-MKNFE	Male	46	No	0	Glendale	91206	34.162515	-118.203869	0	...	Credit Card	-4.0	542.40	38.33	10	96.21	610.28	Stayed	
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.645672	-117.922613	0	...	Bank Withdrawal	73.9	280.85	0.00	0	134.60	415.45	Churned	Com
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.0	1237.85	0.00	0	361.66	1599.51	Churned	Dissatisf
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.9	267.40	0.00	0	22.14	289.54	Churned	Dissatisf
5	0013-MHZWF	Female	23	No	3	Midpines	95345	37.581496	-119.972762	0	...	Credit Card	69.4	571.45	0.00	0	150.93	722.38	Stayed	
6	0013-SMEOE	Female	67	Yes	0	Lompoc	93437	34.757477	-120.550507	1	...	Bank Withdrawal	109.7	7904.25	0.00	0	707.16	8611.41	Stayed	

7 rows × 38 columns

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS JUPYTER | cmd

```
(c) Microsoft Corporation. All rights reserved.

C:\Users\supri\Downloads\Customer-Churn-Prediction-main1\Customer-Churn-Prediction-main>
```

requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

src > Customer_Churn_Prediction.ipynb > M↓ Customer Churn Prediction > df.head()

Generate | + Code | + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline | ··· venv (Python 3.13.3)

[22]

```python
df1=df1.dropna()
df.head()
```

Python

	Customer ID	Gender	Age	Married	Number of Dependents	City	Zip Code	Latitude	Longitude	Number of Referrals	...	Payment Method	Monthly Charge	Total Charges	Total Refunds	Total Extra Data Charges	Total Long Distance Charges	Total Revenue	Customer Status	Cat
0	0002-ORFBO	Female	37	Yes	0	Frazier Park	93225	34.827662	-118.999073	2	...	Credit Card	65.6	593.30	0.00	0	381.51	974.81	Stayed	
1	0003-MKNFE	Male	46	No	0	Glendale	91206	34.162515	-118.203869	0	...	Credit Card	-4.0	542.40	38.33	10	96.21	610.28	Stayed	
2	0004-TLHLJ	Male	50	No	0	Costa Mesa	92627	33.645672	-117.922613	0	...	Bank Withdrawal	73.9	280.85	0.00	0	134.60	415.45	Churned	Com
3	0011-IGKFF	Male	78	Yes	0	Martinez	94553	38.014457	-122.115432	1	...	Bank Withdrawal	98.0	1237.85	0.00	0	361.66	1599.51	Churned	Dissatisf
4	0013-EXCHZ	Female	75	Yes	0	Camarillo	93010	34.227846	-119.079903	3	...	Credit Card	83.9	267.40	0.00	0	22.14	289.54	Churned	Dissatisf

5 rows × 38 columns

[23]

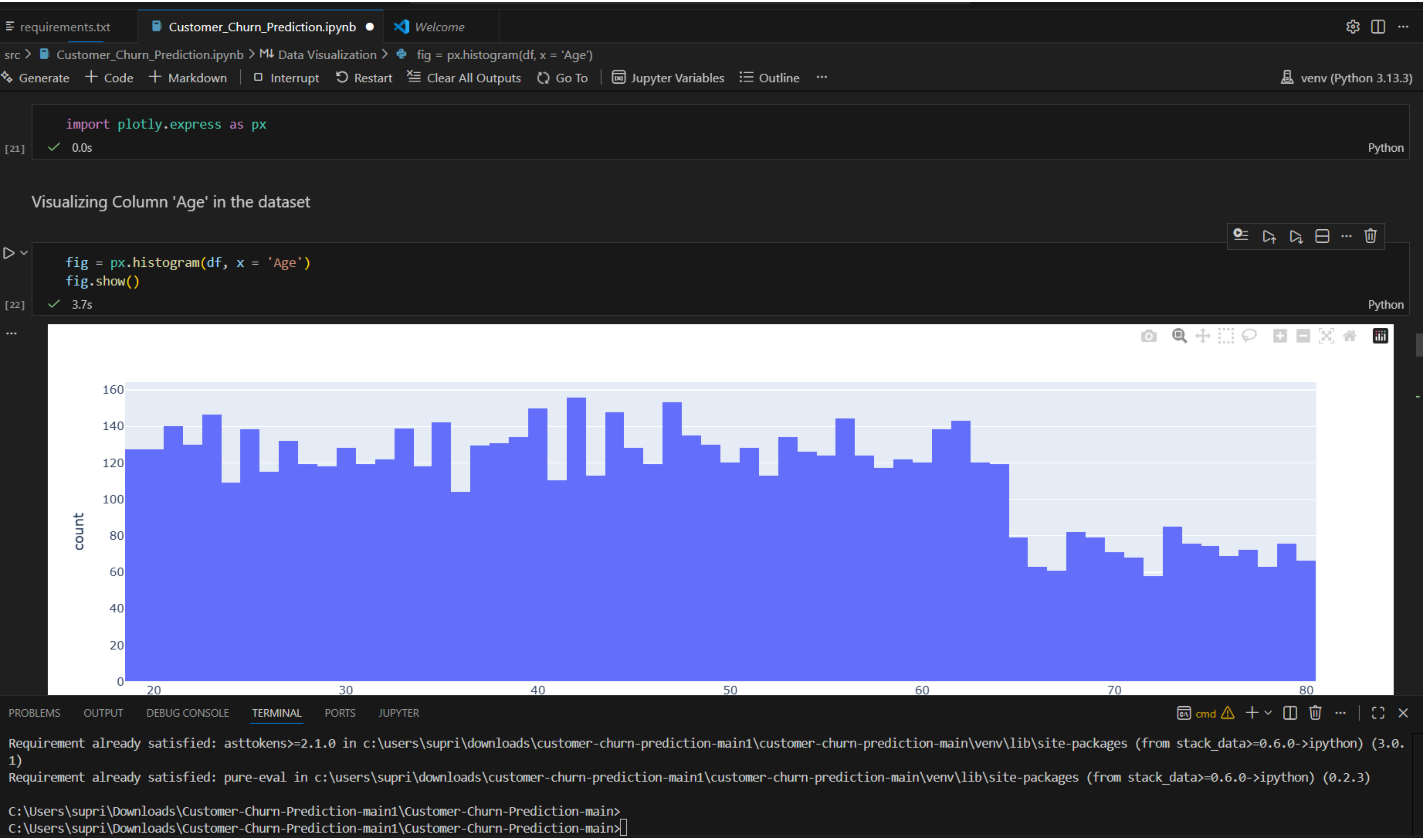
```python
df['Unlimited Data']
```

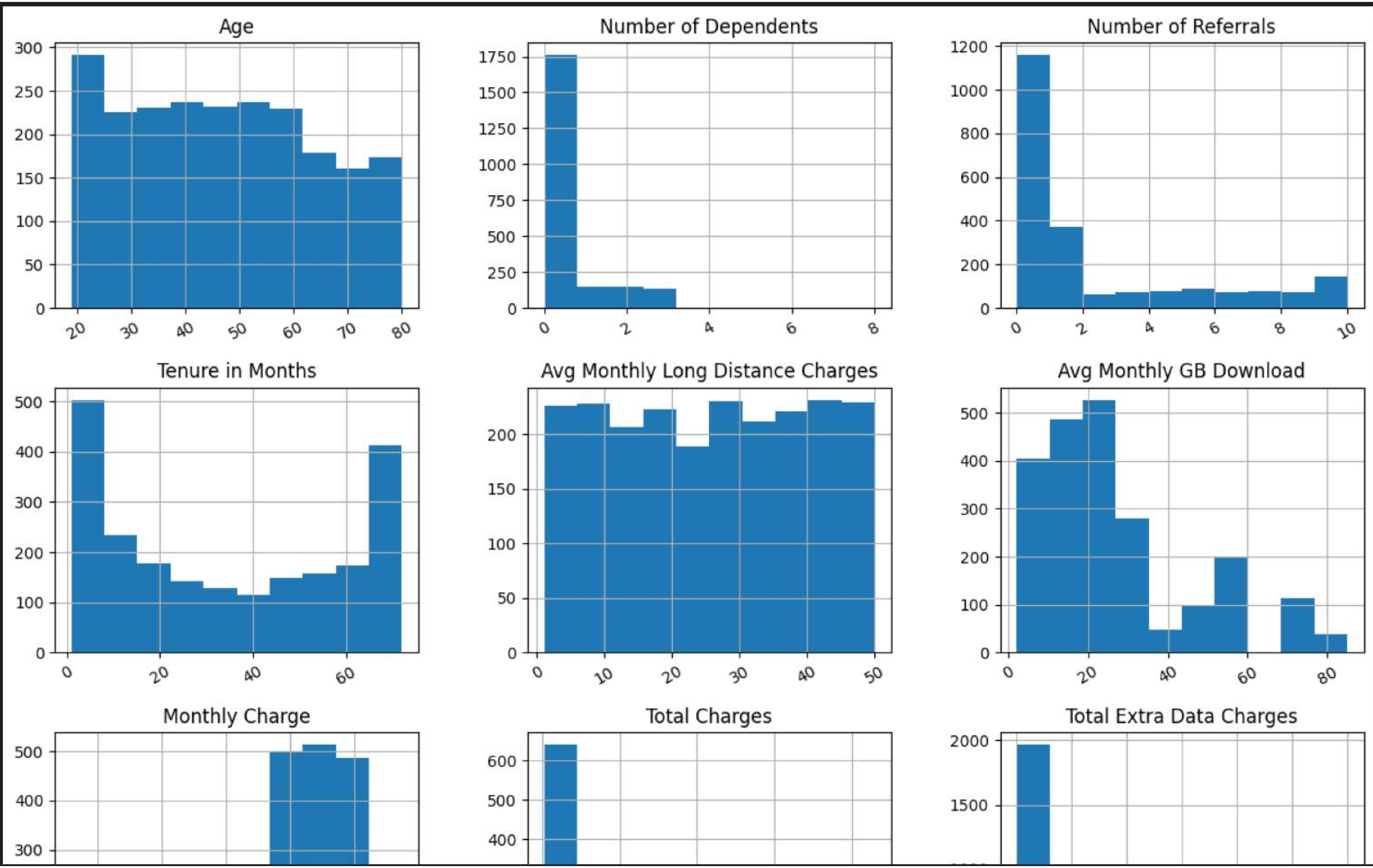
Python

PROBLEMS | OUTPUT | DEBUG CONSOLE | TERMINAL | PORTS | JUPYTER

```
(c) Microsoft Corporation. All rights reserved.

C:\Users\supri\Downloads\Customer-Churn-Prediction-main1\Customer-Churn-Prediction-main>
```


requirements.txt
Customer_Churn_Prediction.ipynb
Welcome
src > Customer_Churn_Prediction.ipynb > M↓ Data Visualization > fig = px.histogram(df, x = 'Age')
Generate
+ Code
+ Markdown
Interrupt
Restart
Clear All Outputs
Go To
Jupyter Variables
Outline
venv (Python 3.13.3)
import plotly.express as px
[21]
0.0s
Python
Visualizing Column 'Age' in the dataset
fig = px.histogram(df, x = 'Age')
fig.show()
[22]
3.7s
Python
count
160
140
120
100
80
60
40
20
0
20
30
40
50
60
70
80
PROBLEMS
OUTPUT
DEBUG CONSOLE
TERMINAL
PORTS
JUPYTER
cmd
Requirement already satisfied: asttokens>=2.1.0 in c:\users\supri\downloads\customer-churn-prediction-main1\customer-churn-prediction-main\venv\lib\site-packages (from stack_data>=0.6.0->ipython) (3.0.1)
Requirement already satisfied: pure-eval in c:\users\supri\downloads\customer-churn-prediction-main1\customer-churn-prediction-main\venv\lib\site-packages (from stack_data>=0.6.0->ipython) (0.2.3)
C:\Users\supri\Downloads\Customer-Churn-Prediction-main1\Customer-Churn-Prediction-main>
C:\Users\supri\Downloads\Customer-Churn-Prediction-main1\Customer-Churn-Prediction-main>


Age
Number of Dependents
Number of Referrals
Tenure in Months
Avg Monthly Long Distance Charges
Avg Monthly GB Download
Monthly Charge
Total Charges
Total Extra Data Charges

requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

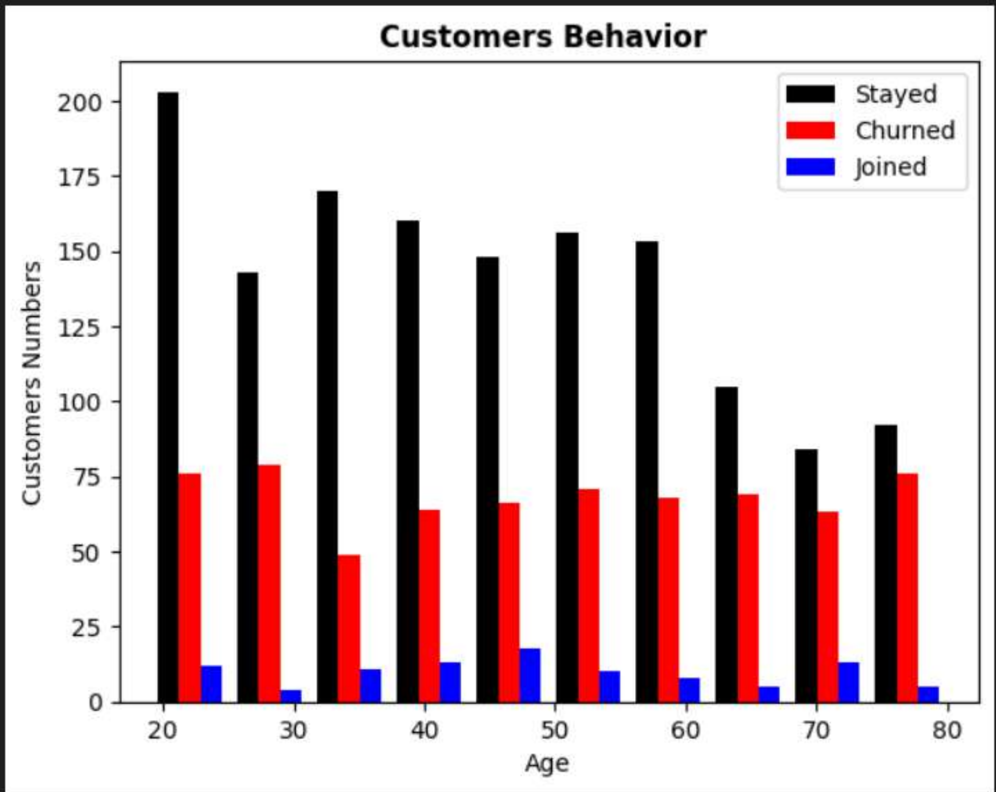
src > Customer_Churn_Prediction.ipynb > M↓ Data Visualization > fig = px.histogram(df, x = 'Age')

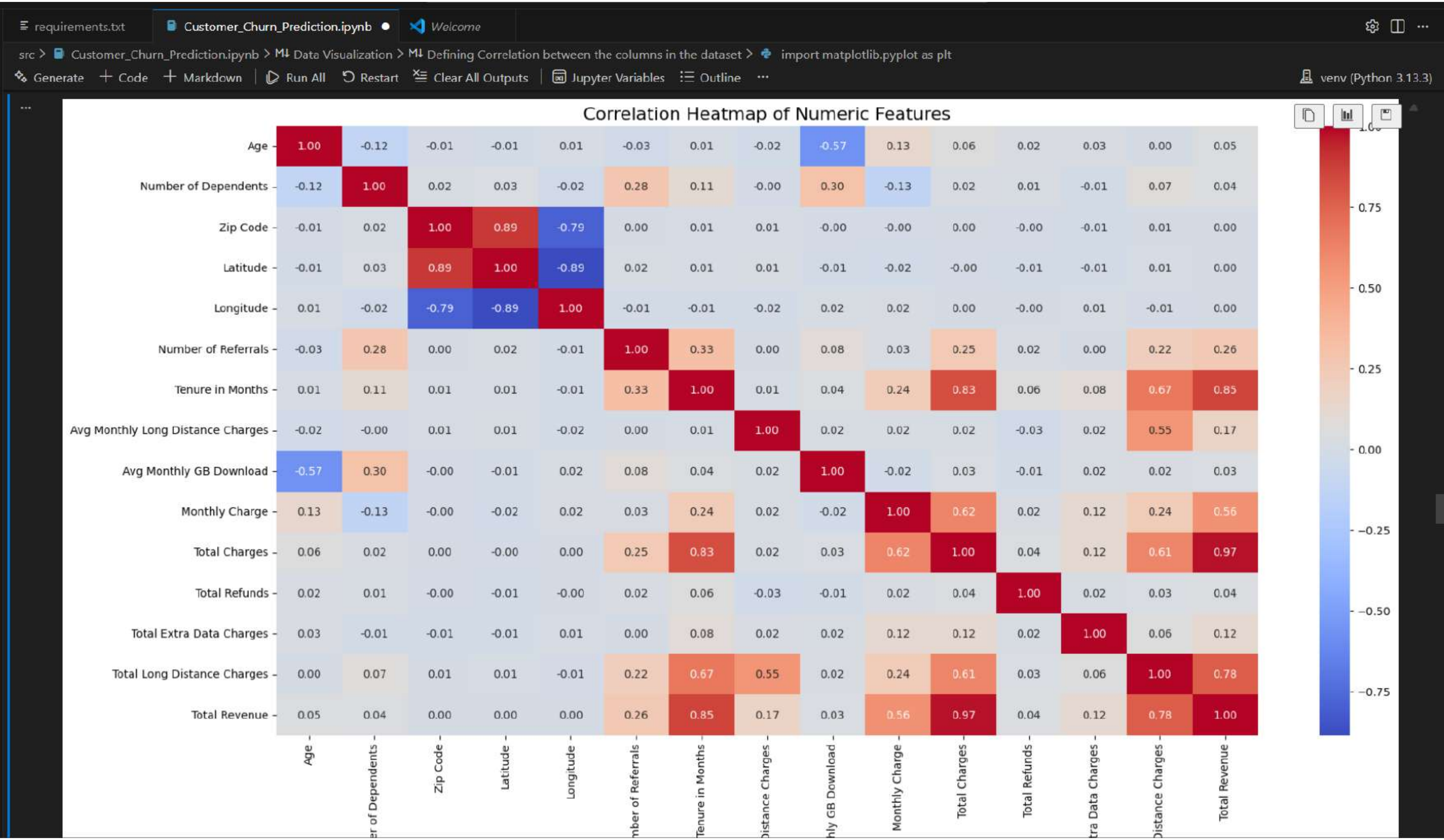
Generate | + Code | + Markdown | Run All | Restart | Clear All Outputs | Go To | Jupyter Variables | Outline | ···

```
    plt.title('Customers Behavior ',fontweight ="bold")
    plt.legend()
```

[26] ✓ 0.2s

```
<matplotlib.legend.Legend at 0x23904604830>
```




requirements.txt
Customer_Churn_Prediction.ipynb
Welcome
src > Customer_Churn_Prediction.ipynb > Data Visualization > Defining Correlation between the columns in the dataset > import matplotlib.pyplot as plt
Generate
Code
Markdown
Run All
Restart
Clear All Outputs
Jupyter Variables
Outline
venv (Python 3.13.3)
Correlation Heatmap of Numeric Features
Age
Number of Dependents
Zip Code
Latitude
Longitude
Number of Referrals
Tenure in Months
Avg Monthly Long Distance Charges
Avg Monthly GB Download
Monthly Charge
Total Charges
Total Refunds
Total Extra Data Charges
Total Long Distance Charges
Total Revenue
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75

Dealing with Imbalance Data

Dropping the Customer_Status

i.e. The column tht we have to predict and set as a dependent variable

```python
X = df1.drop('Customer_Status',axis='columns')
y = df1['Customer_Status']
```

[83] ✓ 0.0s Python

```python
X.head(5)
```

[84] ✓ 0.0s Python

···

	Gender	Age	Married	Number of Dependents	Number of Referrals	Tenure in Months	Phone Service	Avg Monthly Long Distance Charges	Multiple Lines	Internet Service	...	City_Woodland Hills	City_Woody	City_Wrightwood	City_Yermo	City_Yorba Linda	City_Yorkville	Cit
2	1	0.508197	0	0.000	0.0	0.042254	1	0.666462	0	1	...	False	False	False	False	False	False	
3	1	0.967213	1	0.000	0.1	0.169014	1	0.547386	0	1	...	False	False	False	False	False	False	
5	0	0.065574	0	0.375	0.0	0.112676	1	0.321691	0	1	...	False	False	False	False	False	False	
6	0	0.786885	1	0.000	0.1	0.985915	1	0.182598	0	1	...	False	False	False	False	False	False	
7	1	0.540984	1	0.000	0.8	0.873239	1	0.243873	1	1	...	False	False	False	False	False	False	

5 rows × 928 columns

```python
y.head(5)
```

[85] ✓ 0.0s Python

```
2    0
3    0
```

requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

src > Customer_Churn_Prediction.ipynb > M↓ Data Visualization > M↓ Defining Correlation between the columns in the dataset > import matplotlib.pyplot as plt

Generate | + Code | + Markdown | Run All | Restart | Clear All Outputs | Go To | Jupyter Variables | Outline | ... | venv (Python 3.13.3)

```
len(X_train)
```

[87] ✓ 0.0s Python

```
1755
```

```
X_train[:10]
```

[88] ✓ 0.0s Python

	Gender	Age	Married	Number of Dependents	Number of Referrals	Tenure in Months	Phone Service	Avg Monthly Long Distance Charges	Multiple Lines	Internet Service	...	City_Woodland Hills	City_Woody	City_Wrightwood	City_Yermo	City_Yorba Linda	City_Yorkville
5086	1	0.213115	0	0.000	0.0	0.126761	1	0.135417	0	1	...	False	False	False	False	False	False
1686	0	0.000000	0	0.000	0.0	0.140845	1	0.815768	0	1	...	False	False	False	False	False	False
3655	1	0.737705	0	0.000	0.0	0.000000	1	0.154003	1	1	...	False	False	False	False	False	False
697	1	0.983607	1	0.000	0.1	0.408451	1	0.959967	1	1	...	False	False	False	False	False	False
6685	0	0.721311	0	0.000	0.0	0.478873	1	0.247549	1	1	...	False	False	False	False	False	False
5617	0	0.262295	0	0.000	0.0	0.084507	1	0.367034	1	1	...	False	False	False	False	False	False
2938	1	0.098361	0	0.000	0.0	0.873239	1	0.443423	1	1	...	False	False	False	False	False	False
6323	1	0.688525	0	0.000	0.0	0.000000	1	0.257557	0	1	...	False	False	False	False	False	False
3382	1	0.868852	0	0.000	0.0	0.014085	1	0.825368	0	1	...	False	False	False	False	False	False
1184	1	0.377049	1	0.375	0.8	0.774648	1	0.813725	1	1	...	False	False	False	False	False	False

10 rows × 928 columns

Importing the required files for the model that is to applied

1. Random Forest Classifier
2. Logistic Regression

	model	best_score	best_params
0	random_forest	0.798291	{'n_estimators': 10}
1	logistic_regression	0.790313	{'C': 1}
2	naive_bayes_gaussian	0.388604	{}
3	decision_tree	0.792023	{'criterion': 'entropy'}
4	XGB_Classifier	0.828490	{'base_score': 0.5}

requirements.txt | Customer_Churn_Prediction.ipynb | Welcome

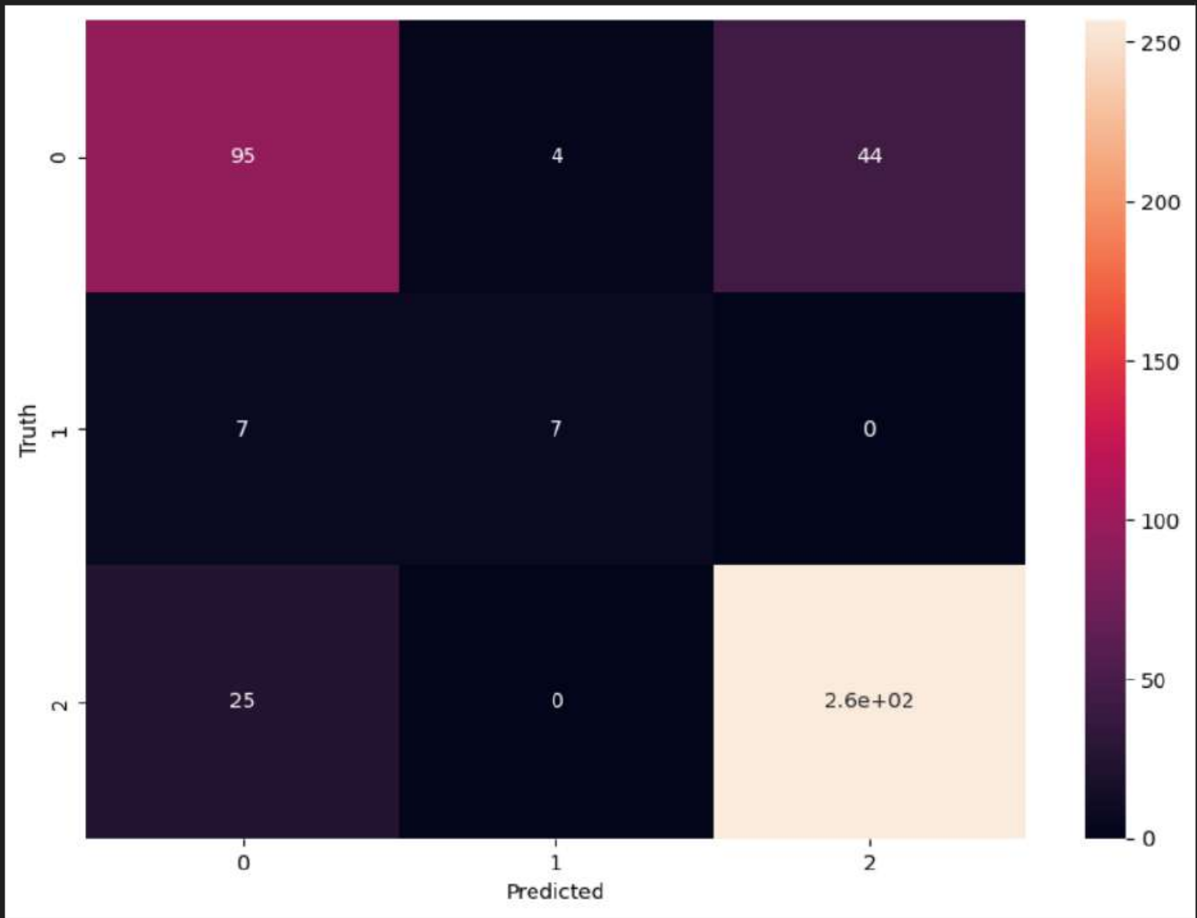
src > Customer_Churn_Prediction.ipynb > Data Model Building > Importing Models > import pandas as pd

Generate | + Code | + Markdown | Run All | Restart | Clear All Outputs | Jupyter Variables | Outline | ··· | venv (Python 3.13.3)

```python
    plt.xlabel('Predicted')
    plt.ylabel('Truth')
```

[228] ✓ 0.1s Python

```
Text(95.72222222222221, 0.5, 'Truth')
```



requirements.txt | Customer_Churn_Prediction.ipynb ● | Welcome

src > Customer_Churn_Prediction.ipynb > M↓ Data Model Building > M↓ Importing Models > import pandas as pd

Generate + Code + Markdown | Run All Restart Clear All Outputs | Jupyter Variables Outline ··· venv (Python 3.13.3)

```python
from sklearn.metrics import classification_report
```

[229] ✓ 0.0s Python

```python
print(classification_report(y_test, y_predicted))
```

[230] ✓ 0.0s Python

```
              precision    recall  f1-score   support

           0       0.75      0.66      0.70       143
           1       0.64      0.50      0.56        14
           2       0.85      0.91      0.88       282

    accuracy                           0.82       439
   macro avg       0.75      0.69      0.72       439
weighted avg       0.81      0.82      0.81       439
```

```python
from sklearn.metrics import accuracy_score
accuracy_score(y_test, y_predicted)
```

[231] ✓ 0.0s Python

```
0.8177676537585421
```

In the end we conclude that the Telecom Customer Churn Prediction was best worked with XGB_Classifier with an accuracy score of 80.86%