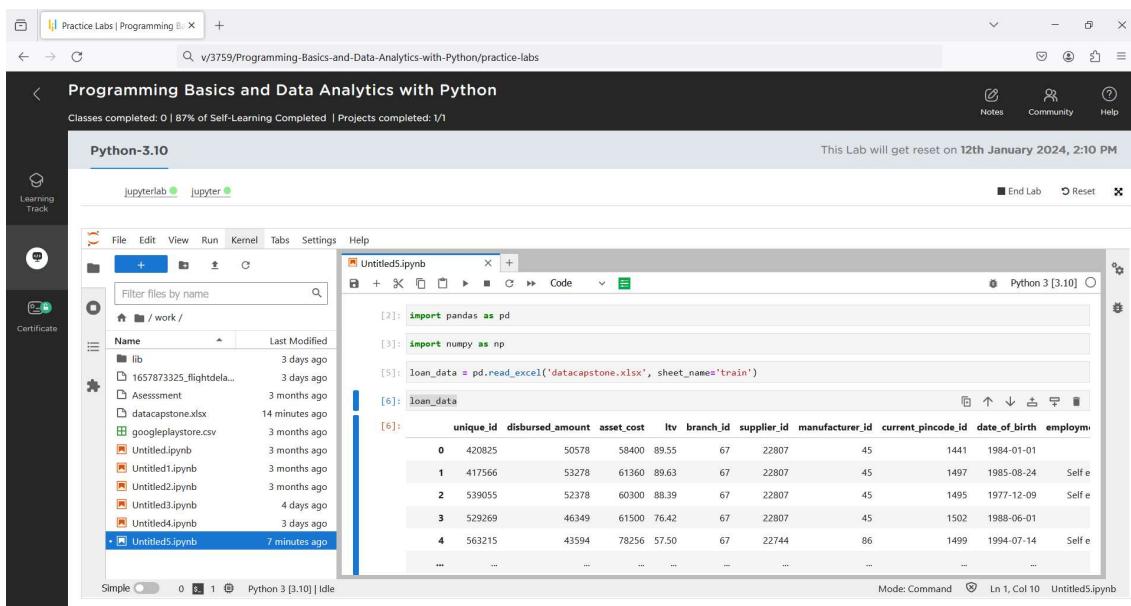


1. Data Preliminary Analysis:

- Perform preliminary data inspection and report the findings like the structure of the data, missing values, duplicates, etc.
 - Variable names in the data may not be in accordance with the identifier naming in Python. Change the variable names accordingly.
 - The presented data might also contain missing values, therefore, exploration will also lead to devising strategies to fill in the missing values. Devise strategies while exploring the data.



Practice Labs | Programming X

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 89% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

Python 3 [3.10]

Mode: Command Ln 1, Col 17 Untitled5.ipynb

Filter files by name

/ work /

Name	Last Modified
lib	3 days ago
1657873325_flightde...	3 days ago
Assessment	3 months ago
datacapstone.xlsx	16 minutes ago
googleplaystore.csv	3 months ago
Untitled.ipynb	3 months ago
Untitled1.ipynb	3 months ago
Untitled2.ipynb	3 months ago
Untitled3.ipynb	4 days ago
Untitled4.ipynb	3 days ago
Untitled5.ipynb	seconds ago

Simple 0 1 Python 3 [3.10] Idle

```
[7]: pd.options.display.max_columns = None
```

```
[8]: loan_data.head()
```

	_amt	new_accts_in_last_six_months	delinquent_accts_in_last_six_months	average_acct_age	credit_history_length	no_of_inquiries	loan_default
0	0	0	0	Oyrs 0mon	Oyrs 0mon	0	0
0	0	0	0	Oyrs 0mon	Oyrs 0mon	0	0
0	0	0	0	Oyrs 0mon	Oyrs 0mon	1	1
0	0	0	0	Oyrs 0mon	Oyrs 0mon	0	0
0	0	0	0	Oyrs 0mon	Oyrs 0mon	0	0

Practice Labs | Programming X

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 92% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

Python 3 [3.10]

Mode: Command Ln 1, Col 17 Untitled5.ipynb

Filter files by name

/ work /

Name	Last Modified
lib	3 days ago
1657873325_flightde...	3 days ago
Assessment	3 months ago
datacapstone.xlsx	18 minutes ago
googleplaystore.csv	3 months ago
Untitled.ipynb	3 months ago
Untitled1.ipynb	3 months ago
Untitled2.ipynb	3 months ago
Untitled3.ipynb	4 days ago
Untitled4.ipynb	3 days ago
Untitled5.ipynb	seconds ago

Simple 0 1 Python 3 [3.10] Idle

```
[9]: loan_data.tail()
```

unique_id	disbursed_amount	asset_cost	ltv	branch_id	supplier_id	manufacturer_id	current_pincode_id	date_of_birth	employment_status	
233149	561031	57759	76350	77.28	5	22289	51	3326	1981-11-10	Self-employed
233150	649600	55009	71200	78.72	138	17408	51	3385	1992-10-15	Self-employed
233151	603445	58513	68000	88.24	135	23313	45	1797	1981-12-19	Self-employed
233152	442948	22824	40458	61.79	160	16212	48	96	1989-07-31	Self-employed
233153	545300	35299	72698	52.27	3	14573	45	17	1968-08-01	Self-employed

Practice Labs | Programming

Programming Basics and Data Analytics with Python

Classes completed: 0 | 94% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[10]: loan_data.columns.values
```

```
[10]: array(['unique_id', 'disbursed_amount', 'asset_cost', 'ltv', 'branch_id', 'supplier_id', 'manufacturer_id', 'current_pincode_id', 'date_of_birth', 'employment_type', 'disbursement_date', 'state_id', 'employee_code_id', 'mobileno_avl_flag', 'aadhar_flag', 'pan_flag', 'voterid_flag', 'driving_flag', 'passport_flag', 'perfrm_cns_score', 'perform_cns_score_desc', 'pri_no_of_accts', 'pri_active_accts', 'pri_overdue_accts', 'pri_current_balance', 'pri_sanctioned_amount', 'pri_disbursed_amount', 'sec_no_of_accts', 'sec_active_accts', 'sec_overdue_accts', 'sec_current_balance', 'sec_sanctioned_amount', 'sec_disbursed_amount', 'primary_instal_amt', 'sec_instal_amt', 'new_accts_in_last_six_months', 'delinquent_accts_in_last_six_months', 'average_acct_age', 'credit_history_length', 'no_of_inquiries', 'loan_default'], dtype=object)
```

Simple 0 1 Python 3 [3.10] Idle Mode: Edit Notes Community Help

Practice Labs | Programming

Programming Basics and Data Analytics with Python

Classes completed: 0 | 95% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[11]: loan_data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 233154 entries, 0 to 233153
Data columns (total 41 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   unique_id        233154 non-null   int64 
 1   disbursed_amount 233154 non-null   int64 
 2   asset_cost       233154 non-null   int64 
 3   ltv              233154 non-null   float64
 4   branch_id        233154 non-null   int64 
 5   supplier_id      233154 non-null   int64 
 6   manufacturer_id 233154 non-null   int64 
 7   current_pincode_id 233154 non-null   int64 
 8   date_of_birth     233154 non-null   datetime64[ns]
 9   employment_type   233154 non-null   object 
 10  disbursal_date    233154 non-null   datetime64[ns]
 11  state_id          233154 non-null   int64 
 12  employee_code_id  233154 non-null   int64 
 13  mobileno_avl_flag 233154 non-null   int64 
 14  aadhar_flag        233154 non-null   int64
```

Simple 0 1 Python 3 [3.10] Idle Mode: Command Notes Community Help

The screenshot shows a Jupyter Notebook interface within a web browser window titled "Practice Labs | Programming". The notebook is titled "Python-3.10" and contains a code cell with the following output:

```
15 pan_flag          233154 non-null int64
16 voterid_flag      233154 non-null int64
17 driving_flag       233154 non-null int64
18 sec_flag           233154 non-null int64
19 perfrm_cns_score   233154 non-null object
20 perfrm_cns_score_desc 233154 non-null object
21 pri_no_of_accts    233154 non-null int64
22 pri_active_accts   233154 non-null int64
23 pri_overdue_accts  233154 non-null int64
24 pri_current_balance 233154 non-null int64
25 pri_sanctioned_amount 233154 non-null int64
26 pri_disbursement_amount 233154 non-null int64
27 sec_no_of_accts    233154 non-null int64
28 sec_active_accts   233154 non-null int64
29 sec_overdue_accts  233154 non-null int64
30 sec_current_balance 233154 non-null int64
31 sec_sanctioned_amount 233154 non-null int64
32 sec_disbursed_amount 233154 non-null int64
33 primary_instal_amt 233154 non-null int64
34 sec_instal_amt     233154 non-null int64
35 new_accts_in_last_six_months 233154 non-null int64
36 delinquent_accts_in_last_six_months 233154 non-null int64
...           ...       ...       ...
```

The screenshot shows a Jupyter Notebook interface within a web browser window titled "Practice Labs | Programming". The notebook is titled "Python-3.10" and contains a code cell with the following output:

```
37 average_acct_age      233154 non-null object
38 credit_history_length   233154 non-null object
39 no_of_inquiries         233154 non-null int64
40 loan_default            233154 non-null int64
dtypes: datetime64[ns](2), float64(1), int64(34), object(4)
memory usage: 72.9+ MB
```

The screenshot shows a web-based Python environment for data analysis. The top navigation bar includes tabs for 'Practice Labs' and 'Programming'. The main title is 'Programming Basics and Data Analytics with Python'. A progress bar indicates 'Classes completed: 0 | 99% of Self-Learning Completed' and 'Projects completed: 1/1'. On the left, there's a sidebar with 'Learning Track' and 'Certificate' sections. The main area is titled 'Python 3.10' and contains a Jupyter notebook interface. The notebook has two cells: Cell 14 displays `pd.options.display.max_rows = None` and Cell 15 displays `loan_data.isnull().sum()`. The output of Cell 15 shows the count of null values for various columns in the 'loan_data' DataFrame. The bottom status bar shows 'Mode: Command' and the file path 'Untitled3.ipynb'.

Name	Last Modified
lib	3 days ago
1657873325_flightdetails.ipynb	4 days ago
Assessment	3 months ago
datacapstone.xlsx	an hour ago
googleplaystore.csv	3 months ago
Untitled1.ipynb	3 months ago
Untitled1.ipynb	3 months ago
Untitled2.ipynb	3 months ago
Untitled3.ipynb	4 days ago
Untitled4.ipynb	3 days ago
Untitled5.ipynb	seconds ago

```
[14]: pd.options.display.max_rows = None
[15]: loan_data.isnull().sum()
unique_id          0
disbursed_amount   0
asset_cost          0
ltv                 0
branch_id           0
supplier_id         0
manufacturer_id     0
current_pincode_id  0
date_of_birth        0
employment_type      7661
disursal_date        0
state                0
employee_code_id     0
mobile_no_avl_flag    0
aadhar_flag           0
pan_flag              0
```

The screenshot shows a Jupyter Notebook interface with the following details:

- Header:** Practice Labs | Programming
- Title Bar:** v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs
- Section Title:** Programming Basics and Data Analytics with Python
- Status Bar:** Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1
- Help Bar:** Notes, Community, Help
- Section:** Python-3.10
- File List:** Filter files by name (work /), showing files like lib, 1657873325_flightdelay..., Assessment, datacapstone.xlsx, googleplaystore.csv, Untitled.ipynb, Untitled1.ipynb, Untitled2.ipynb, Untitled3.ipynb, Untitled4.ipynb, Untitled5.ipynb.
- Code Editor:** Untitled5.ipynb, Python 3 [3.10] mode. The code contains numerous zero values for various financial and behavioral metrics.
- Bottom Bar:** Simple, Python 3 [3.10] | Idle, Mode: Command, Ln 1, Col 1, Untitled5.ipynb

The screenshot shows a Jupyter Notebook interface within a web browser window. The title bar reads "Programming Basics and Data Analytics with Python". The notebook file is titled "Untitled5.ipynb". The code cell displays a list of variables and their corresponding values:

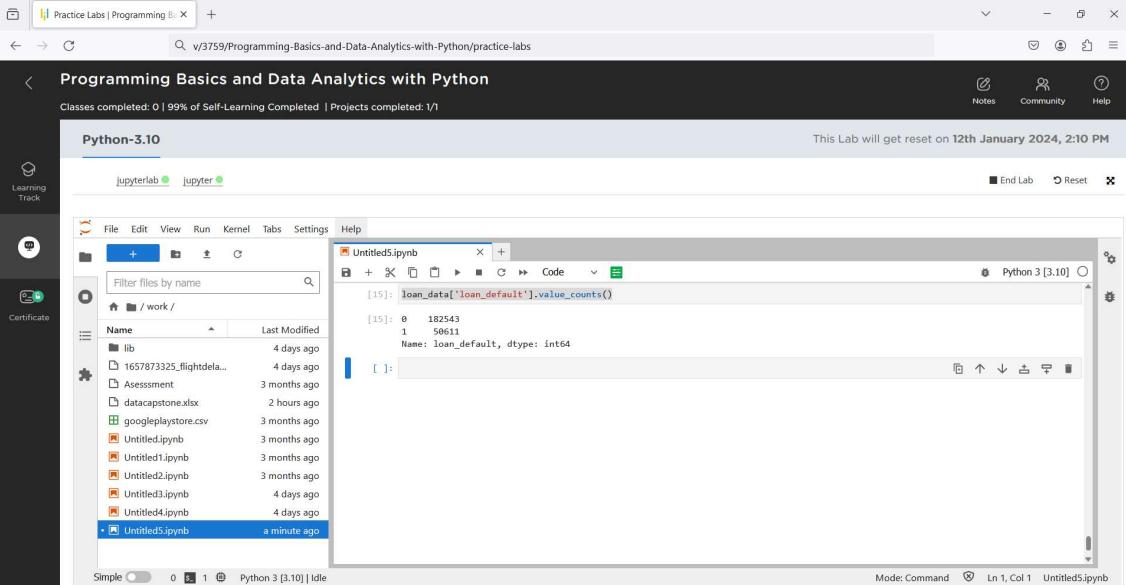
```
pari_mean_cnscore_desc 0
pri_no_of_accts 0
pri_active_accts 0
pri_overdue_accts 0
pri_current_balance 0
pri_sanctioned_amount 0
pri_disbursed_amount 0
sec_mean_cnscore_desc 0
sec_active_accts 0
sec_overdue_accts 0
sec_current_balance 0
sec_sanctioned_amount 0
sec_disbursed_amount 0
primary_instal_amt 0
sec_instal_amt 0
new_accts_in_last_six_months 0
delinquent_accts_in_last_six_months 0
average_acct_ape 0
credit_history_length 0
no_of_inquiries 0
loan_default 0
dtype: int64
```

The screenshot shows a Jupyter Notebook interface within a web browser window. The title bar reads "Programming Basics and Data Analytics with Python". The notebook file is titled "Untitled5.ipynb". The code cells show a sequence of operations on a dataset named "loan_data":

```
[16]: loan_data['employment_type'].unique()
[17]: array(['Salaried', 'Self employed', nan], dtype=object)
[18]: loan_data['employment_type'].fillna('Not Specified', inplace=True)
[19]: loan_data['employment_type'].unique()
[20]: array(['Salaried', 'Self employed', 'Not Specified'], dtype=object)
[21]: loan_data['employment_type'].isnull().sum()
```

2. Performing EDA:

- Provide the statistical description of the quantitative data variables
- How is the target variable distributed overall?
- Study the distribution of the target variable across the various categories like branch, city, state, branch, supplier, manufacturer, etc.
- What are the different employment types given in the data? Can a strategy be developed to fill in the missing values (if any)? Use pie charts to express how different types of employment defines defaulter and non-defaulters.
- Has age got something to do with defaulting? What is the distribution of age w.r.t. to defaulters and non-defaulters?
- What type of ID is presented by most of the customers as proofs?



The screenshot shows a Jupyter Notebook interface within a web browser window. The title bar indicates the URL is [v/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs](#). The main header says "Programming Basics and Data Analytics with Python". On the left, there's a sidebar with "Learning Track" and "Certificate" sections. The central area has a file tree showing several Jupyter notebooks and other files. A code cell titled "Untitled5.ipynb" is active, displaying the following Python code and output:

```
[15]: loan_data['loan_default'].value_counts()
```

```
[15]: 0    182543  
1     59611  
Name: loan_default, dtype: int64
```

The notebook interface includes tabs for "File", "Edit", "View", "Run", "Kernel", "Tabs", "Settings", and "Help". The status bar at the bottom shows "Mode: Command" and "Ln 1, Col 1 Untitled5.ipynb".

Practice Labs | Programming +

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb Python 3 [3.10]

```
[23]: loan_data.groupby(['employment_type', 'loan_default'])['employment_type'].count()
```

employment_type	loan_default	Count
Not Specified	0	6017
	1	1034
Salaried	0	77948
	1	19910
Self employed	0	98578
	1	29957

Name: employment_type, dtype: int64

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Learning Track Certificate

Simple 0 1 Python 3 [3.10] Idle

Practice Labs | Programming +

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb Python 3 [3.10]

```
[16]: loan_data['delinquent_accts_in_last_six_months'].value_counts()
```

Value	Count
0	214959
1	14941
2	2470
3	537
4	138
5	58
6	20
7	13
8	7
11	3
12	3
10	2
9	2
20	1

Name: delinquent_accts_in_last_six_months, dtype: int64

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Learning Track Certificate

Simple 0 1 Python 3 [3.10] Idle

Practice Labs | Programming

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

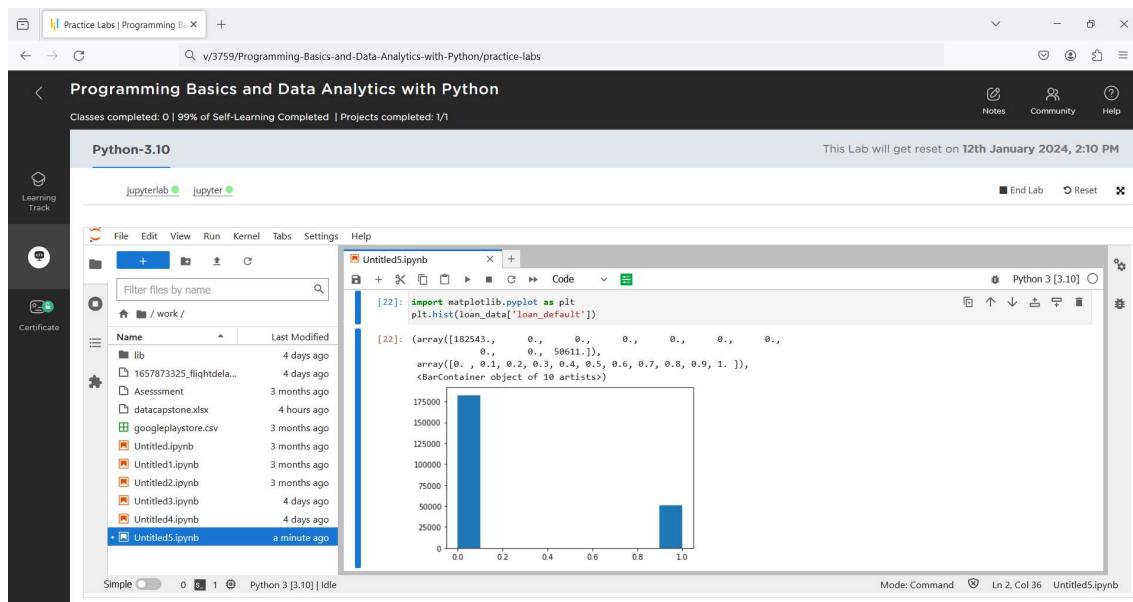
Untitled5.ipynb

```
[18]: bad_loans['employment_type'].value_counts()
[19]: good_loans['employment_type'].value_counts()
```

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Learning Track Certificate

Simple 0 1 Python 3 [3.10] Idle



Practice Labs | Programming

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

untitled5.ipynb

```
[27]: bad_loans.groupby(['employment_type']).sum().plot(kind='pie', y='loan_default', autopct='%1.0f%%')  
/tmp/ipynbkernel_71/147628399.py:1: FutureWarning: The default value of numeric_only in DataFrameGroupBy.sum is deprecated. In a future version, numeric_only will default to False. Either specify numeric_only or select only columns which should be valid for the function.  
bad_loans.groupby(['employment_type']).sum().plot(kind='pie', y='loan_default', autopct='%1.0f%%')
```

Mode: Command

Practice Labs | Programming

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

untitled5.ipynb

```
[34]: bad_loans_manufacturer_ids = bad_loans.groupby(['manufacturer_id'])['loan_default'].count()  
[36]: bad_loans_manufacturer_ids.plot.bar(ylabel='Total Loan Defaulters')
```

Mode: Command

Practice Labs | Programming

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[38]: bad_loans_supplier_ids.plot.bar(ylabel='Total Loan Defaulters')
```

<AxesSubplot: xlabel='supplier_id', ylabel='Total Loan Defaulters'>

Simple 0 1 Python 3 [3.10] | Idle

Mode: Command 2006 28/12/2023

Practice Labs | Programming

v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[48]: bad_loans_branch_ids.plot.bar(ylabel='Total Loan Defaulters')
```

<AxesSubplot: xlabel='branch_id', ylabel='Total Loan Defaulters'>

Simple 0 1 Python 3 [3.10] | Idle

Mode: Command 2006 28/12/2023

The screenshot shows a Jupyter Notebook interface within a web browser window titled "Practice Labs | Programming". The browser address bar shows the URL "v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs". The main content area is titled "Programming Basics and Data Analytics with Python". A sidebar on the left contains "Learning Track" and "Certificate" sections. The notebook interface has tabs for "jupyterlab" and "jupyter". The code cell [44] displays the following Python code and its output:

```
ns_branch_ids.index['defaulter_count':bad_loans_branch_ids.values]).sort_values(by=['defaulter_count'], ascending=False).head(5)
```

branch_id	defaulter_count
20	36
1	2
30	67
3	5
12	16

The status bar at the bottom indicates "Mode: Command" and "Ln 1, Col 1 Untitled5.ipynb".

The screenshot shows a Jupyter Notebook interface within a web browser window titled "Practice Labs | Programming". The browser address bar shows the URL "v3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs". The main content area is titled "Programming Basics and Data Analytics with Python". A sidebar on the left contains "Learning Track" and "Certificate" sections. The notebook interface has tabs for "jupyterlab" and "jupyter". The code cell [53] displays the following Python code:

```
from datetime import datetime, date
def calculate_age(born):
    born = born.date()
    today = date.today()
    return today.year - born.year - ((today.month, today.day) < (born.month, born.day))
```

The code cell [54] displays the following Python code:

```
from datetime import datetime
loan_data['age'] = loan_data['date_of_birth'].apply(calculate_age)
```

The code cell [58] displays the following Python code and its output:

```
loan_data[['date_of_birth', 'age', 'loan_default']].head(5)
```

date_of_birth	age	loan_default
1984-01-01	39	0
1985-08-24	38	0
1977-12-09	46	1
1988-06-01	35	0

The status bar at the bottom indicates "Mode: Command" and "Ln 1, Col 60 Untitled5.ipynb".

Practice Labs | Programming

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[61]: loan_data[['date_of_birth', 'age', 'loan_default']].plot.scatter(x='age', y='loan_default', c='DarkBlue')
```

The scatter plot shows a horizontal distribution of points for each age group. Most points are at y=0 (loan_default=0), with a few scattered at y=1 (loan_default=1). The x-axis ranges from approximately 30 to 70.

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Practice Labs | Programming

Programming Basics and Data Analytics with Python

Classes completed: 0 | 99% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:10 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[62]: loan_data[['date_of_birth', 'age', 'loan_default']].groupby(['age', 'loan_default']).count()
```

age	loan_default	date_of_birth
23	0	470
	1	170
24	0	1229
	1	384
25	0	1637
	1	549
26	0	5602
	1	1940
27	0	7745

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Therefore, the most common ID provided are aadhar and voter, followed by PAN.

Practice Labs | Programming

https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 41% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:13 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

```
[58]: loan_data_flags.value_counts().sort_index().plot.bar(x='Target Value', y='Number of Occurrences')
```

```
[58]: <AxesSubplot: xlabel='mobileno_avl_flag,aadhar_flag,pan_flag,voterid_flag,driving_flag,passport_flag'>
```

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Practice Labs | Programming

https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 41% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:13 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

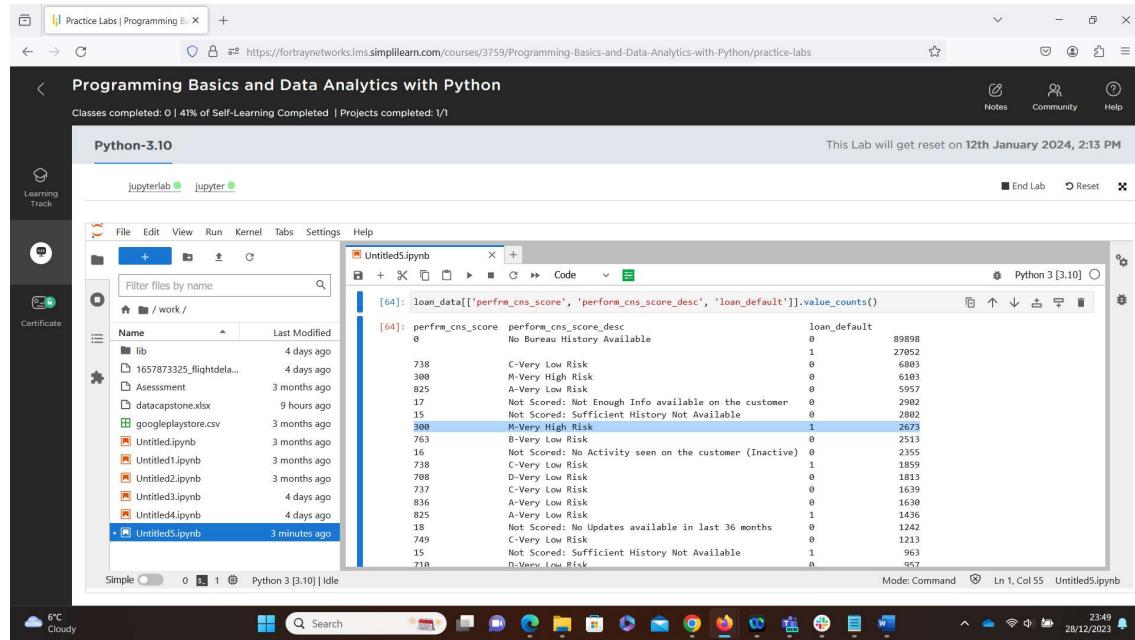
```
[58]: loan_data_flags.value_counts().sort_index().plot.bar(x='Target Value', y='Number of Occurrences')
```

```
[58]: <AxesSubplot: xlabel='mobileno_avl_flag,aadhar_flag,pan_flag,voterid_flag,driving_flag,passport_flag'>
```

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Performing EDA and Modelling:

- Study the credit bureau score distribution. How is the distribution for defaulters vs. non-defaulters? Explore in detail.
- Explore the primary and secondary account details. Is the information in some way related to the loan default probability?
- Is there a difference between the sanctioned and disbursed amount of primary and secondary loans? Study the difference by providing appropriate statistics and graphs.
- Do customers who make higher numbers of inquiries end up being higher risk candidates?
- Is credit history, that is, new loans in the last six months, loans defaulted in the last six months, time since the first loan, etc., a significant factor in estimating the probability of loan defaulters?
- Perform logistic regression modeling, predict the outcome for the test data, and validate the results using the confusion matrix.



The screenshot shows a Jupyter Notebook interface within a web browser window titled "Practice Labs | Programming". The browser URL is <https://fortrainingnetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs>. The notebook title is "Python-3.10". The code cell displays the following Python code and output:

```
[64]: loan_data[['perfrm_cns_score', 'perform_cns_score_desc', 'loan_default']].value_counts()
```

perfrm_cns_score	perform_cns_score_desc	loan_default	Count
0	No Bureau History Available	0	89898
738	C-Very Low Risk	1	27852
300	A-Very High Risk	0	6883
825	A-Very Low Risk	0	5957
17	Not Scored: Not Enough Info available on the customer	0	2902
15	Not Scored: Sufficient History Not Available	0	2892
300	M-Very High Risk	1	2673
763	B-Very Low Risk	0	2513
16	Not Scored: No Activity seen on the customer (Inactive)	0	2355
738	C-Very Low Risk	1	1859
708	D-Very Low Risk	0	1813
737	C-Very Low Risk	0	1639
836	A-Very Low Risk	0	1630
825	A-Very Low Risk	1	1436
18	Not Scored: No Updates available in last 36 months	0	1242
749	C-Very Low Risk	0	1213
15	Not Scored: Sufficient History Not Available	1	963
718	D-Very Low Risk	0	957

Credit Bureau score on its own appear to have mixed outcome. For example, Credit Bureau score of 300 (M – Very High Risk) has got 6103 non-defaulters and 2673 defaulters.

Again, Credit Bureau score of 825 (A – Very Low Risk) has got 1436 defaulters and 5957 non-defaulters.

The screenshot shows a Jupyter Notebook interface within a web browser window. The title bar reads "Practice Labs | Programming" and the URL is "https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs". The main content area is titled "Python-3.10" and displays a code cell output:

```
[65]: loan_data[['no_of_inquiries', 'loan_default']].value_counts()
```

no_of_inquiries	loan_default	Count
0	0	159404
1	0	42557
1	1	16844
2	0	5441
2	1	3110
3	0	1491
3	1	1259
4	0	526
4	1	517
5	0	234
5	1	212
6	0	148
6	1	131
7	0	93
7	1	88
8	0	61
8	1	55
9	0	44
9	1	30

Below the code cell, the status bar shows "Mode: Command" and "Ln 1, Col 1 Untitled5.ipynb".

Number of enquiries appear to be unrelated as there are 42557 defaulters (out of 50611 total defaulters) who have made no inquiries.

Age less than 28 has high probability of defaulters. The ratio between defaulters and non-defaulters in age range less than 28 years is high (almost 25%).

The screenshot shows a Jupyter Notebook interface within a web browser window. The title bar reads "Practice Labs | Programming" and the URL is "https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs". The main content area is titled "Python-3.10" and displays four code cells:

```
[86]: loan_data[loan_data['age'] <= 28].groupby('loan_default')['loan_default'].count()
```

```
[86]: loan_default
0    24718
1    6866
Name: loan_default, dtype: int64
```

```
[84]: loan_data[(loan_data['age'] > 28) & (loan_data['age'] < 55)].groupby('loan_default')['loan_default'].count()
```

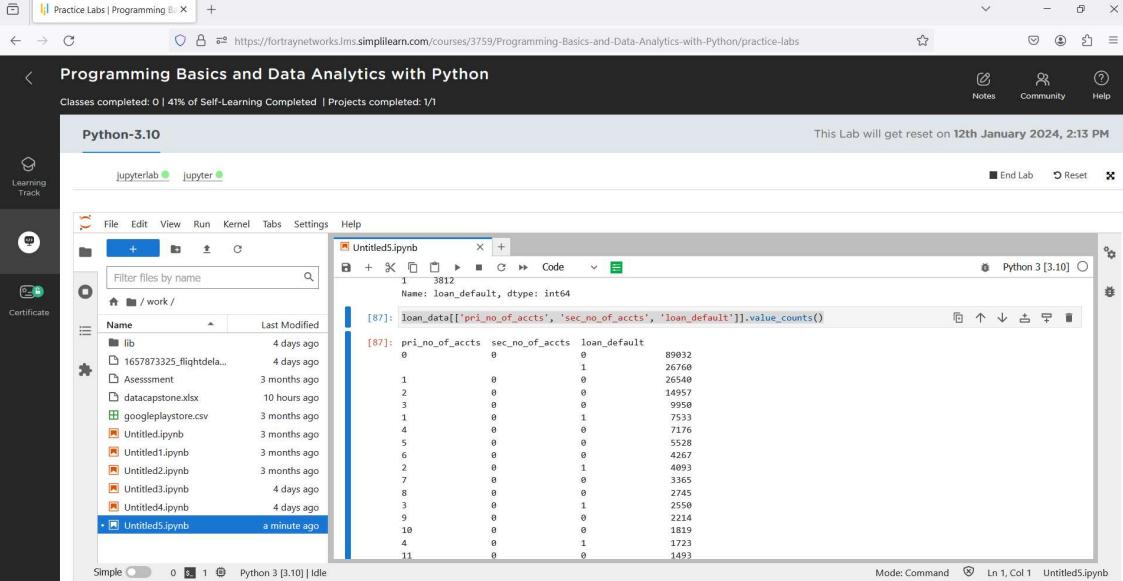
```
[84]: loan_default
0    141792
1    38733
Name: loan_default, dtype: int64
```

```
[85]: loan_data[loan_data['age'] >= 55].groupby('loan_default')['loan_default'].count()
```

```
[85]: loan_default
0    16933
1    3812
Name: loan_default, dtype: int64
```

Below the code cells, the status bar shows "Mode: Command" and "Ln 1, Col 60 Untitled5.ipynb".

There are 26760 out of a total 50611 defaulters, i.e., about 50% that had no primary or secondary accounts. However, around 50% defaulters had one or more primary or secondary accounts. Therefore this along with Age is a good indication to determine potential defaulters.



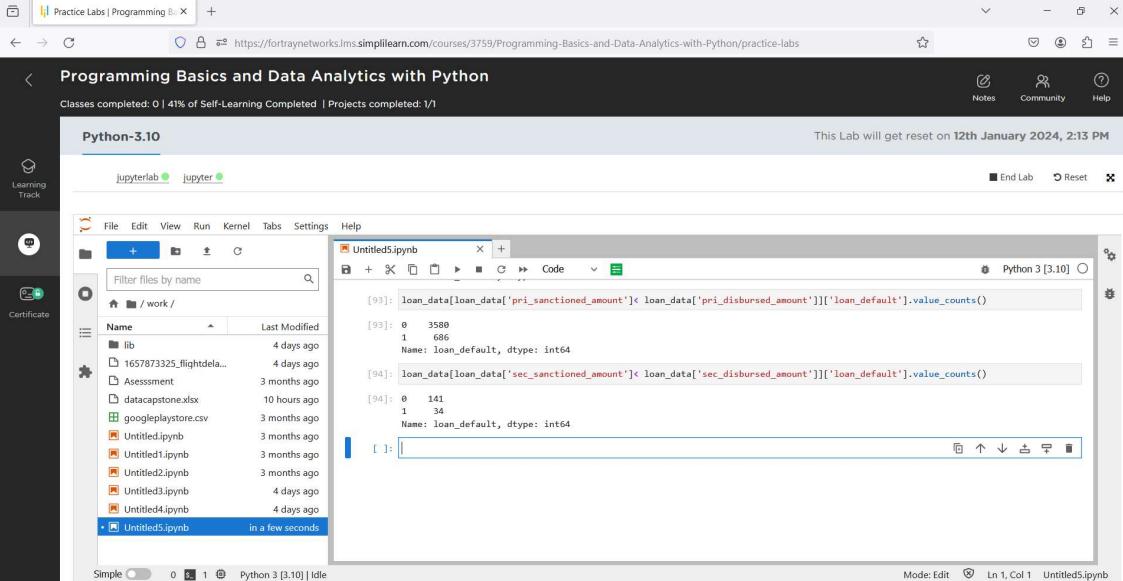
The screenshot shows a Jupyter Notebook interface within a web browser window. The notebook is titled "Untitled5.ipynb" and is running in Python 3 [3.10]. The code cell contains the following command:

```
[87]: loan_data[['pri_no_of_accts', 'sec_no_of_accts', 'loan_default']].value_counts()
```

The output of the command is a table showing the count of values for each combination of primary and secondary account counts and loan default status. The table has columns: pri_no_of_accts, sec_no_of_accts, and loan_default. The rows represent the count of observations for each combination of these three variables. The counts are as follows:

pri_no_of_accts	sec_no_of_accts	loan_default	Count
0	0	0	89032
1	0	1	26760
2	0	0	26540
3	0	0	14957
4	0	1	9958
5	0	0	7533
6	0	0	7176
7	0	0	5528
8	0	0	4267
9	0	1	4093
10	0	0	3365
11	0	0	2745
12	0	1	2559
13	0	0	2214
14	0	0	1819
15	0	1	1723
16	0	0	1493

Significant percentage (around 16%) of cases are defaulters where disbursed amount is greater than sanctioned amount in their primary and secondary loan account.



The screenshot shows a Jupyter Notebook interface within a web browser window. The notebook is titled "Untitled5.ipynb" and is running in Python 3 [3.10]. The code cell contains the following command:

```
[93]: loan_data[loan_data['pri_sanctioned_amount'] < loan_data['pri_disbursed_amount']]['loan_default'].value_counts()
```

The output of the command is a table showing the count of values for each primary loan default status. The table has columns: pri_sanctioned_amount and pri_disbursed_amount, and the loan_default column. The rows represent the count of observations for each primary loan default status. The counts are as follows:

pri_sanctioned_amount	pri_disbursed_amount	loan_default	Count
0	3588	0	3588
1	686	1	686

The second part of the code cell contains the following command:

```
[94]: loan_data[loan_data['sec_sanctioned_amount'] < loan_data['sec_disbursed_amount']]['loan_default'].value_counts()
```

The output of the command is a table showing the count of values for each secondary loan default status. The table has columns: sec_sanctioned_amount and sec_disbursed_amount, and the loan_default column. The rows represent the count of observations for each secondary loan default status. The counts are as follows:

sec_sanctioned_amount	sec_disbursed_amount	loan_default	Count
0	141	0	141
1	34	1	34

Practice Labs | Programming

https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 41% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:13 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

untitled5.ipynb

```
[102]: loan_data[(loan_data['sec_sanctioned_amount'] < loan_data['sec_disbursed_amount']) | (loan_data['pri_sanctioned_amount'] < loan_data['pri_disbursed_amount'])].groupby(['loan_default']).count().plot(kind='pie', y='loan_default', autopct='%1.0f%')

[102]: <AxesSubplot: ylabel='loan_default'>
```

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Simple 0 1 Python 3 [3.10] Idle

Practice Labs | Programming

https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 41% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:13 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

untitled5.ipynb

```
[107]: loan_data[(loan_data['new_accts_in_last_six_months'] > 2) & (loan_data['delinquent_accts_in_last_six_months'] > 2)][['new_accts_in_last_six_months', 'delinquent_accts_in_last_six_months', 'loan_default']]
```

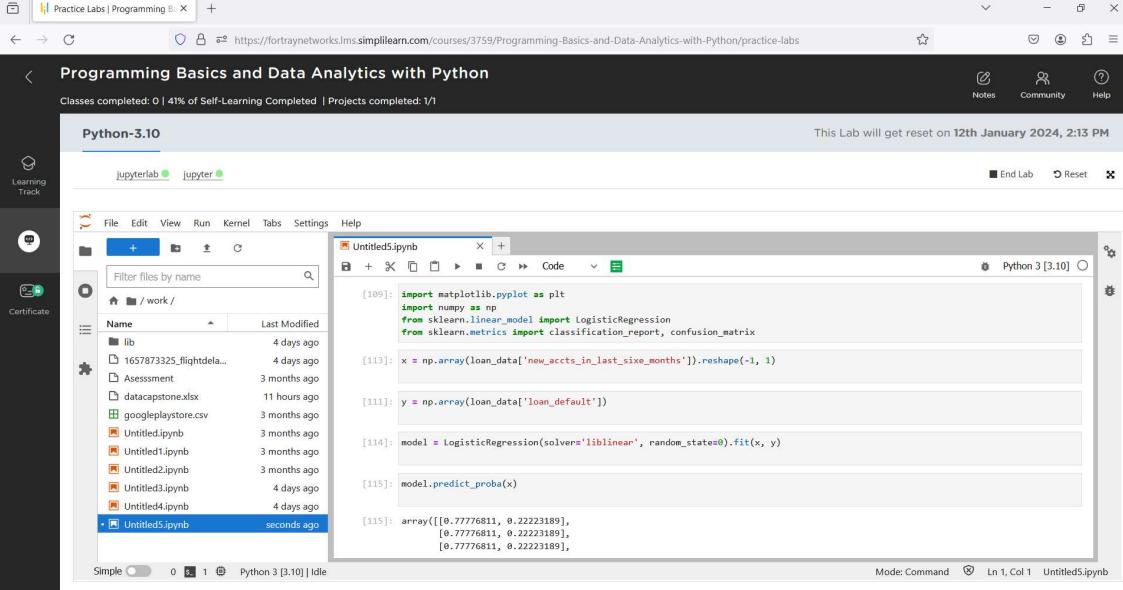
new_accts_in_last_six_months	delinquent_accts_in_last_six_months	loan_default
3	0	32
4	0	28
3	1	15
5	0	13
4	0	8
7	0	7
6	0	6
4	0	5
5	1	4
3	0	4
6	1	3
7	0	3
4	0	3
8	0	2
6	1	2

Mode: Command Ln 1, Col 1 Untitled5.ipynb

Simple 0 1 Python 3 [3.10] Idle

Significant percentage of cases where new/delinquent accounts in last 6 months is greater than 2 are defaulters.

- Perform logistic regression modeling, predict the outcome for the test data, and validate the results using the confusion matrix.



```

import matplotlib.pyplot as plt
import numpy as np
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import classification_report, confusion_matrix

x = np.array(loan_data['new_accts_in_last_six_months'].reshape(-1, 1))

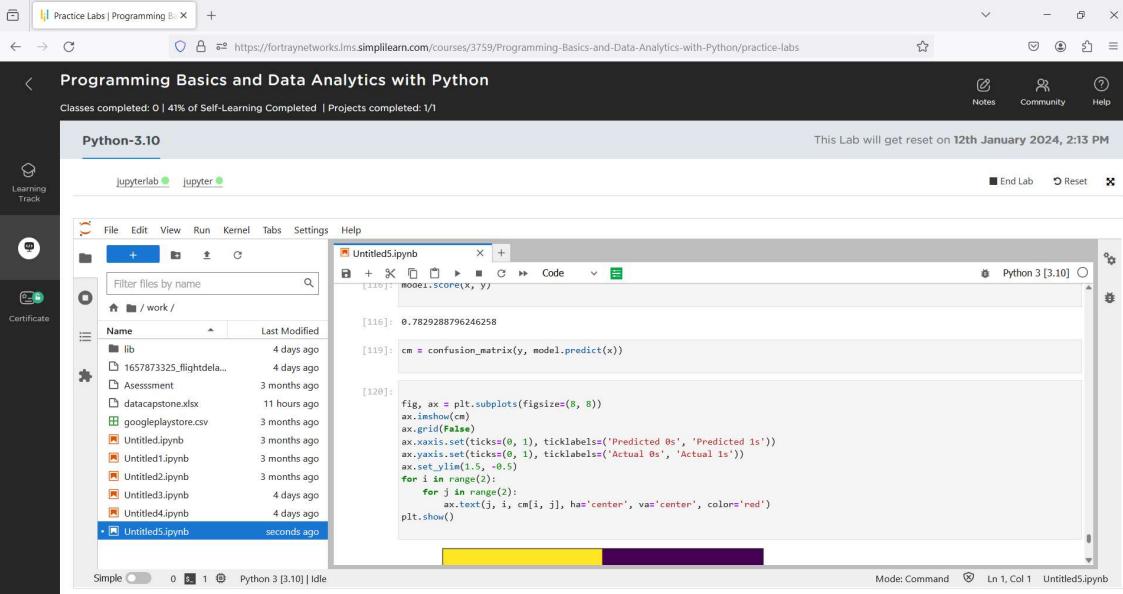
y = np.array(loan_data['loan_default'])

model = LogisticRegression(solver='liblinear', random_state=0).fit(x, y)

model.predict_proba(x)

array([[0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189],
       [0.77776811, 0.22223189]])

```



```

model.score(x, y)

cm = confusion_matrix(y, model.predict(x))

fig, ax = plt.subplots(figsize=(8, 8))
ax.imshow(cm)
ax.grid(False)
ax.xaxis.set(ticks=(0, 1), ticklabels=('Predicted 0', 'Predicted 1'))
ax.yaxis.set(ticks=(0, 1), ticklabels=('Actual 0', 'Actual 1'))
ax.set_ylim(1.5, -0.5)
for i in range(2):
    for j in range(2):
        ax.text(j, i, cm[i, j], ha='center', va='center', color='red')
plt.show()

```

Practice Labs | Programming

https://fortraynetworks.lms.simplilearn.com/courses/3759/Programming-Basics-and-Data-Analytics-with-Python/practice-labs

Programming Basics and Data Analytics with Python

Classes completed: 0 | 41% of Self-Learning Completed | Projects completed: 1/1

This Lab will get reset on 12th January 2024, 2:13 PM

Python-3.10

jupyterlab jupyter

File Edit View Run Kernel Tabs Settings Help

Untitled5.ipynb

Actual 0s: 182543
Actual 1s: 90111
Predicted 0s: 0
Predicted 1s: 0

Python 3 [3.10]

Learning Track Certificate

Simple 0 1 Python 3 [3.10] Idle

Mode: Command Ln 1, Col 1 Untitled5.ipynb

	Predicted 0s	Predicted 1s
Actual 0s	182543	0
Actual 1s	90111	0