

## CYT245 Assignment 1. Learn AlienVault IP Reputation database

### Teamwork policy

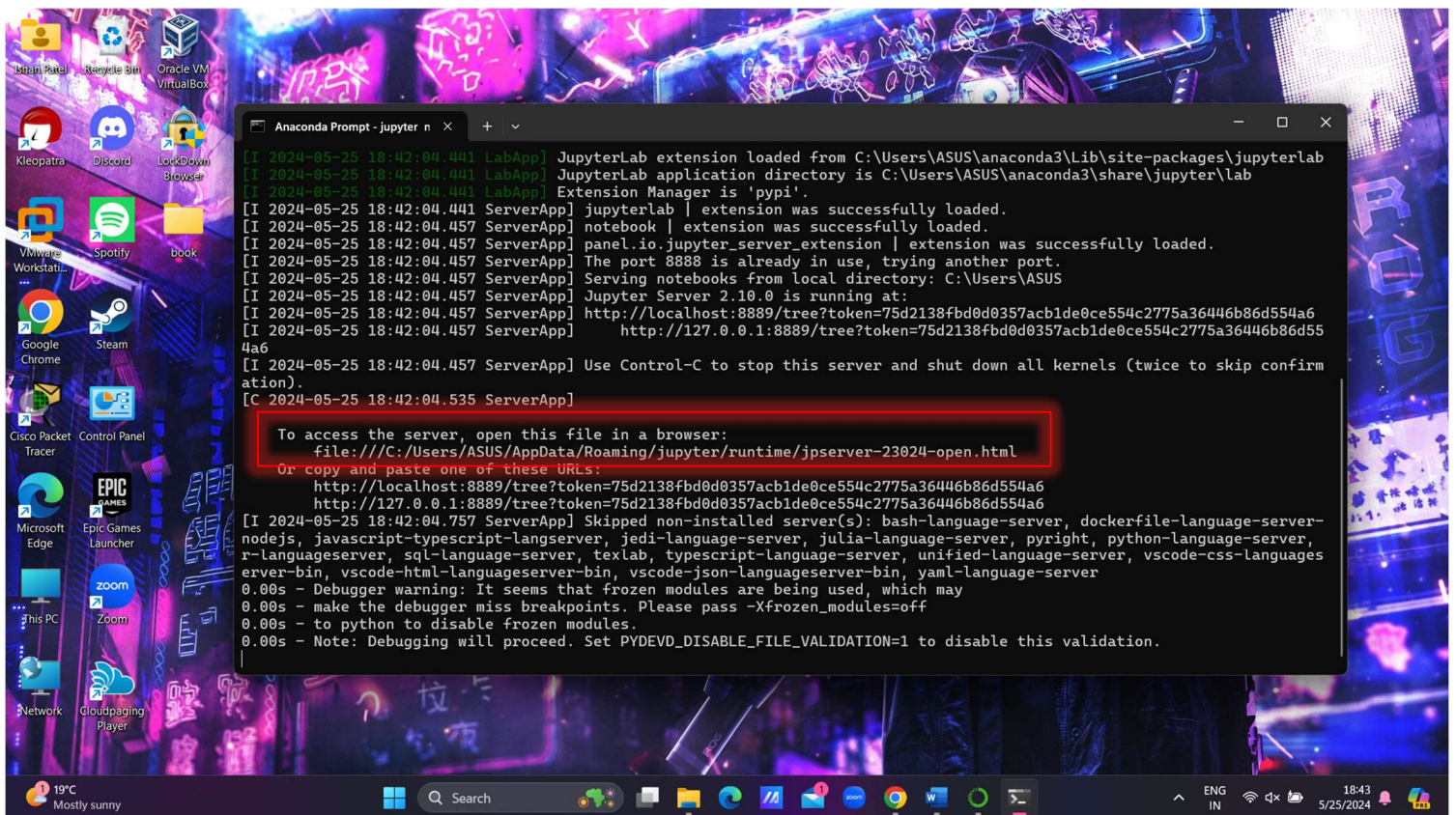
You are asked to enter student names below:

1. Ishan Aakash Patel - 146151238	2.
3.	4.

Team leader is \_\_Ishan Patel\_\_\_\_\_

**Only one submission from the team is expected. It will be done by the current team leader; however this role should be rotated from one teamwork to another. Screen0 (see below) should be made on the Leader's computer.**

**Preparation - Step 0.** At the start, make screenshot of the starting screen. The screenshot must contain indication of the laptop ownership (like user name).



## **Preparation - Step 1. Prepare technology environment on your computer**

Prepare your environment where you will be able to run Python or R scripts. Detailed about working on environment can be found in the Chapter 2 of the textbook:

- Data-Driven security by Jay Jacobs, Bob Rudis. ISBN: 978-1-118-79372-5

### For Python

You have the Anaconda-Python-Pandas environment ready to go (after the CYT175).

Make sure that you can start Jupyter Notebook. If needed, review the demos referred in the CYT175 Labs task descriptions.

Make sure that your local Jupyter host server is up and running.

### For R

Install R/RStudio how it is described in the Chapter 2.

## **The Assignment 1 Task description**

Major source of information:

- “Data Driven Security” text book, Chapter 3.
- Python/R scripts and data file are available in the Lab task description, zipped file book.rar
- Python script (clean one) is also attached to the task description.

Objectives of Assignment 1

- Primary – Learn the content of IP Reputation Database, recommended to be used as the feed to Threat Intelligence practice.
- Secondary – using samples of code, to make next step in learning Python and Pandas tools.

## **Start Workflow**

Note: For R, you need to accommodate the technical activities accordingly (e.g. take relevant Listings, etc.)

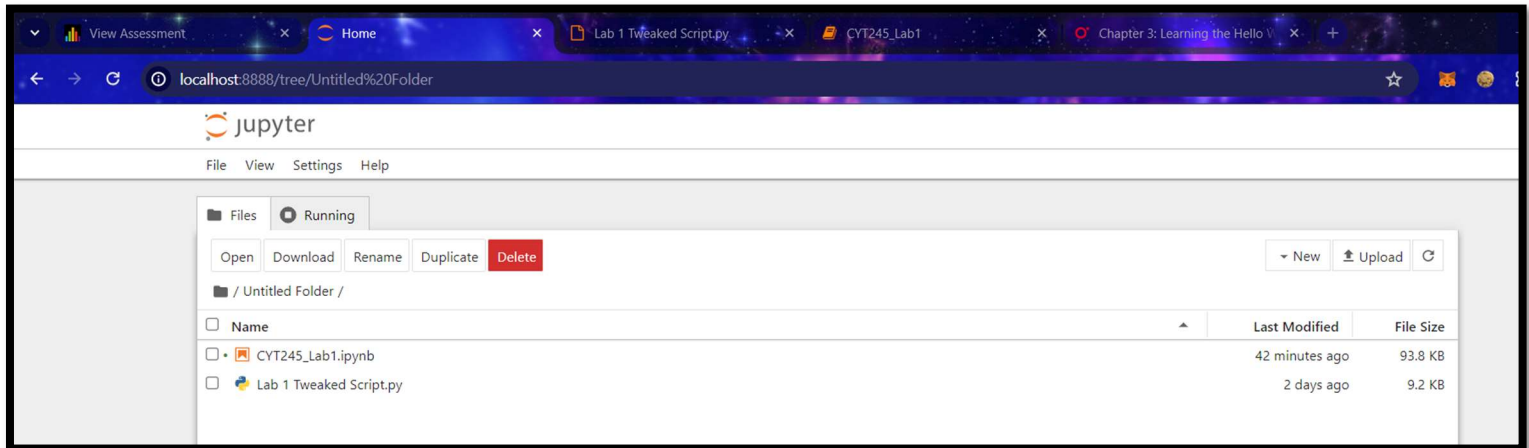
## **Learning**

Study materials from Chapter 3 to capture the content. In particular, pay attention to textual comments. Combine reading with running Python scripts and answering the questions.

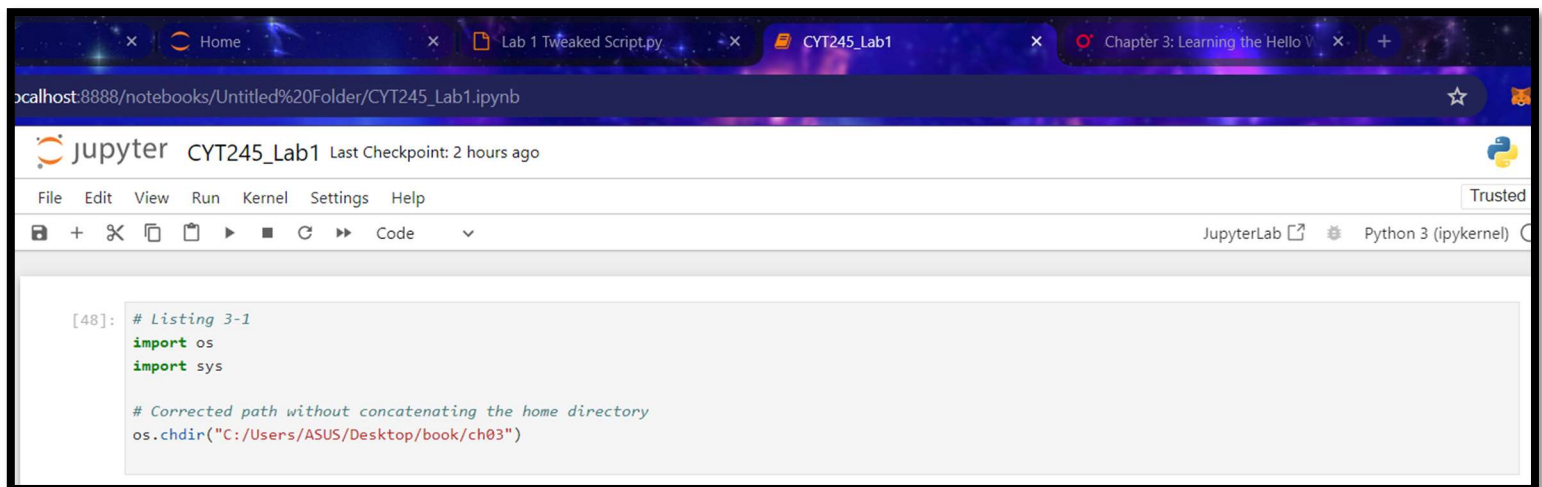
Note: Screenshots are required for each step. Include them into your submission.

Step 1. Unzip the book.rar and move the folder book to your Anaconda environment.

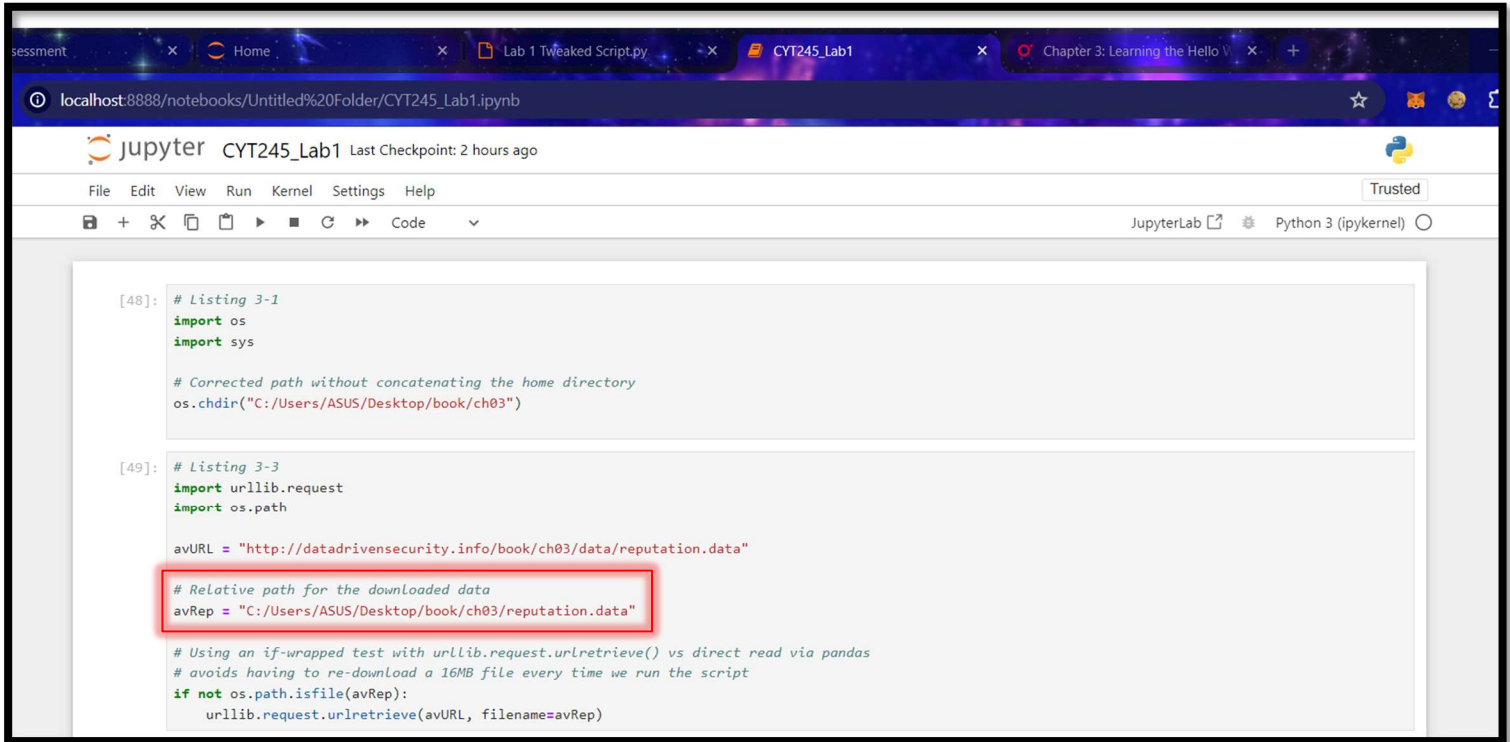
Doing that, you make samples of code and data easily available.



Step 2. Open the Python script file and run Listing 1 portion in your notebook. Resolve error messages if you have them. This way you are making the sample of data available for next steps.



**Step 3.** Run the Listing 3-3. You set relative path for the downloaded data.



The screenshot shows a JupyterLab interface with a notebook titled 'CYT245\_Lab1'. The code in the cell is as follows:

```
[48]: # Listing 3-1
import os
import sys

# Corrected path without concatenating the home directory
os.chdir("C:/Users/ASUS/Desktop/book/ch03")

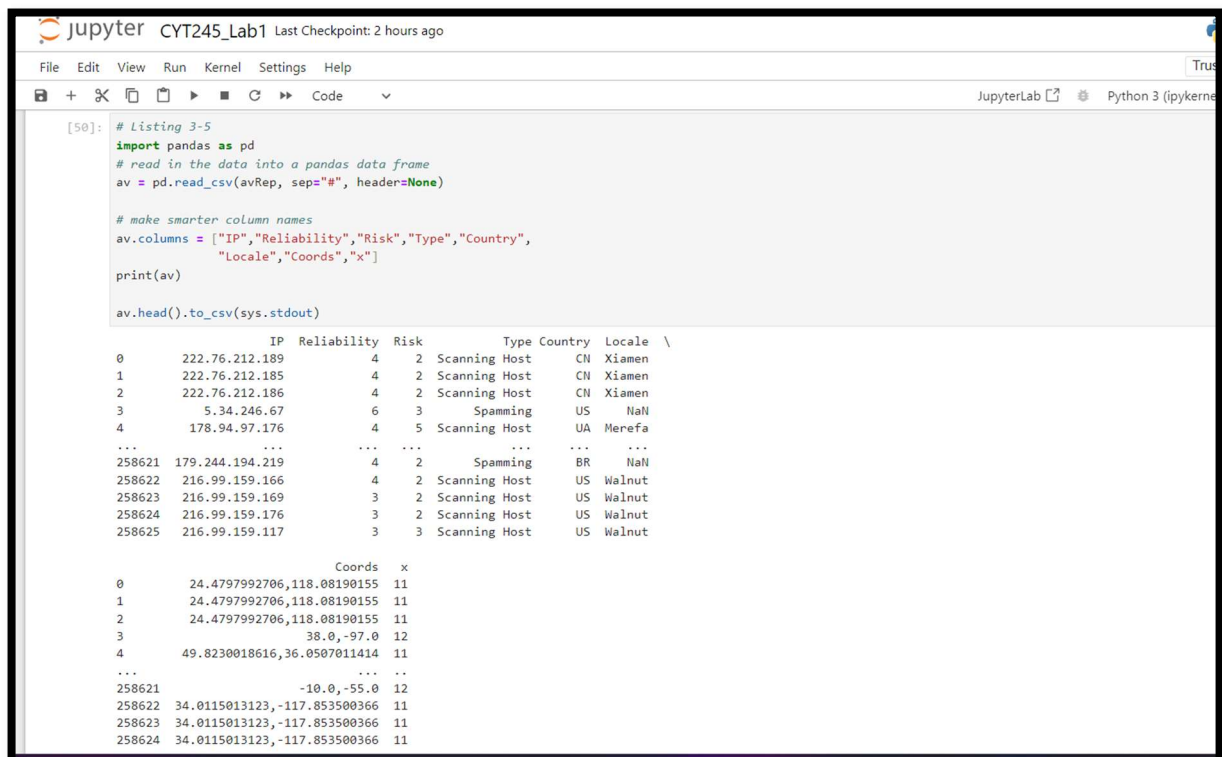
[49]: # Listing 3-3
import urllib.request
import os.path

avURL = "http://datadrivensecurity.info/book/ch03/data/reputation.data"

# Relative path for the downloaded data
avRep = "C:/Users/ASUS/Desktop/book/ch03/reputation.data"

# Using an if-wrapped test with urllib.request.urlretrieve() vs direct read via pandas
# avoids having to re-download a 16MB file every time we run the script
if not os.path.isfile(avRep):
    urllib.request.urlretrieve(avURL, filename=avRep)
```

**Step 4.** Run Listing 3-5. At this point of time you will obtain the result showing first 5 rows from the file.



The screenshot shows the same JupyterLab interface with the next cell of code. The code is as follows:

```
[50]: # Listing 3-5
import pandas as pd
# read in the data into a pandas data frame
av = pd.read_csv(avRep, sep=";", header=None)

# make smarter column names
av.columns = ["IP", "Reliability", "Risk", "Type", "Country",
              "Locale", "Coords", "x"]
print(av)

av.head().to_csv(sys.stdout)
```

The output of the code is a table showing the first 5 rows of the data:

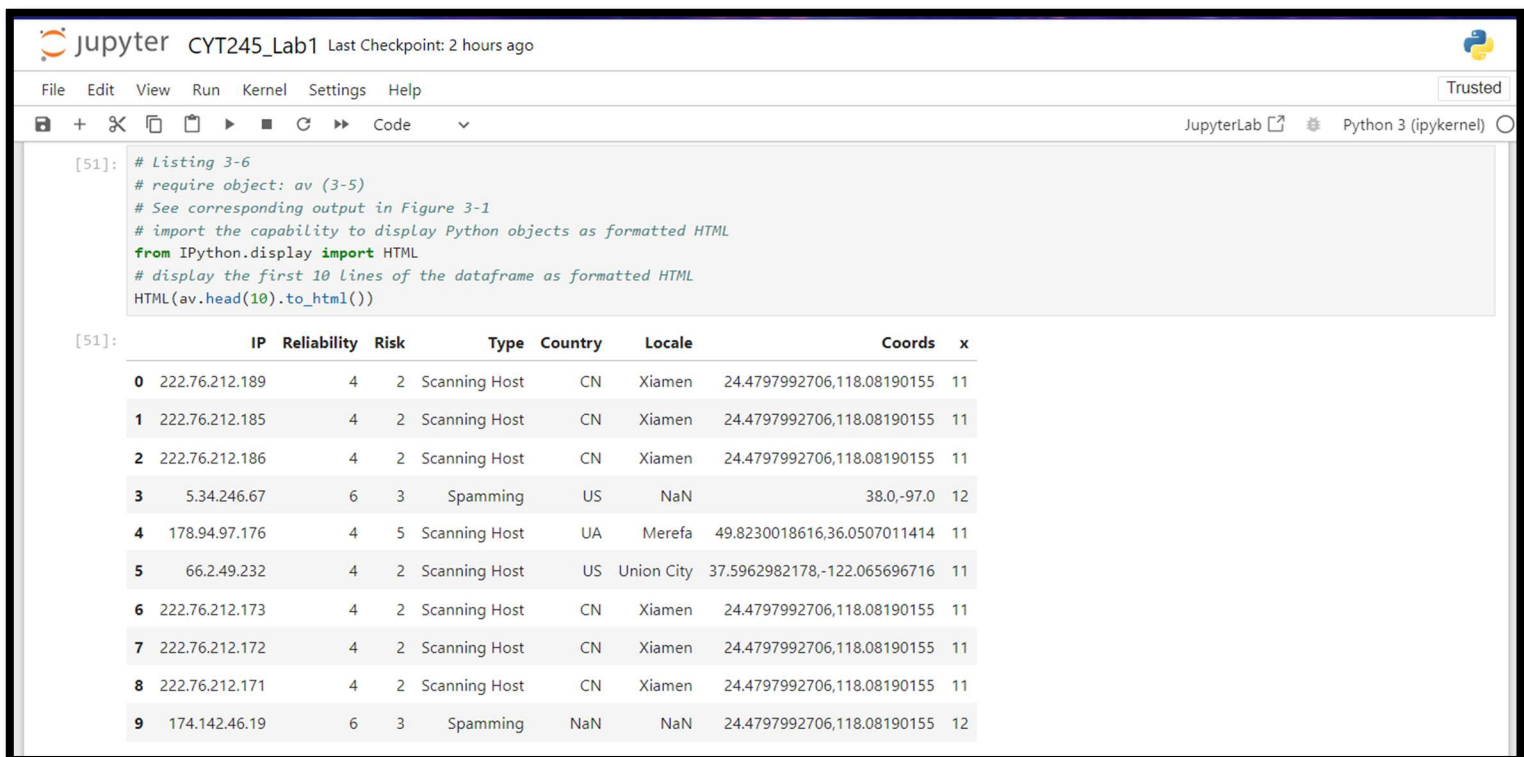
	IP	Reliability	Risk	Type	Country	Locale
0	222.76.212.189	4	2	Scanning Host	CN	Xiamen
1	222.76.212.185	4	2	Scanning Host	CN	Xiamen
2	222.76.212.186	4	2	Scanning Host	CN	Xiamen
3	5.34.246.67	6	3	Spamming	US	NaN
4	178.94.97.176	4	5	Scanning Host	UA	Merefa

The output also includes a section for 'Coords' and 'x' values, which are not shown in the table above.

This code defines the structure of IP Reputation Database. Run the code and observe the result.  
Answer the following questions:

1. What is Pandas name for the IP Reputation Database csv file?  
⇒ avRep as we had assigned the path of reputation.data in the previous listing.
2. What are Columns names of the Pandas data frame?  
⇒ The column names of the Pandas DataFrame are:  
⇒ "IP"  
⇒ "Reliability"  
⇒ "Risk"  
⇒ "Type"  
⇒ "Country"  
⇒ "Locale"  
⇒ "Coords"  
⇒ "x"

**Step 5.** Run Listing 3-6. You will see HTML formatted output of the same data frame.



```
[51]: # Listing 3-6
# require object: av (3-5)
# See corresponding output in Figure 3-1
# import the capability to display Python objects as formatted HTML
from IPython.display import HTML
# display the first 10 lines of the dataframe as formatted HTML
HTML(av.head(10).to_html())
```

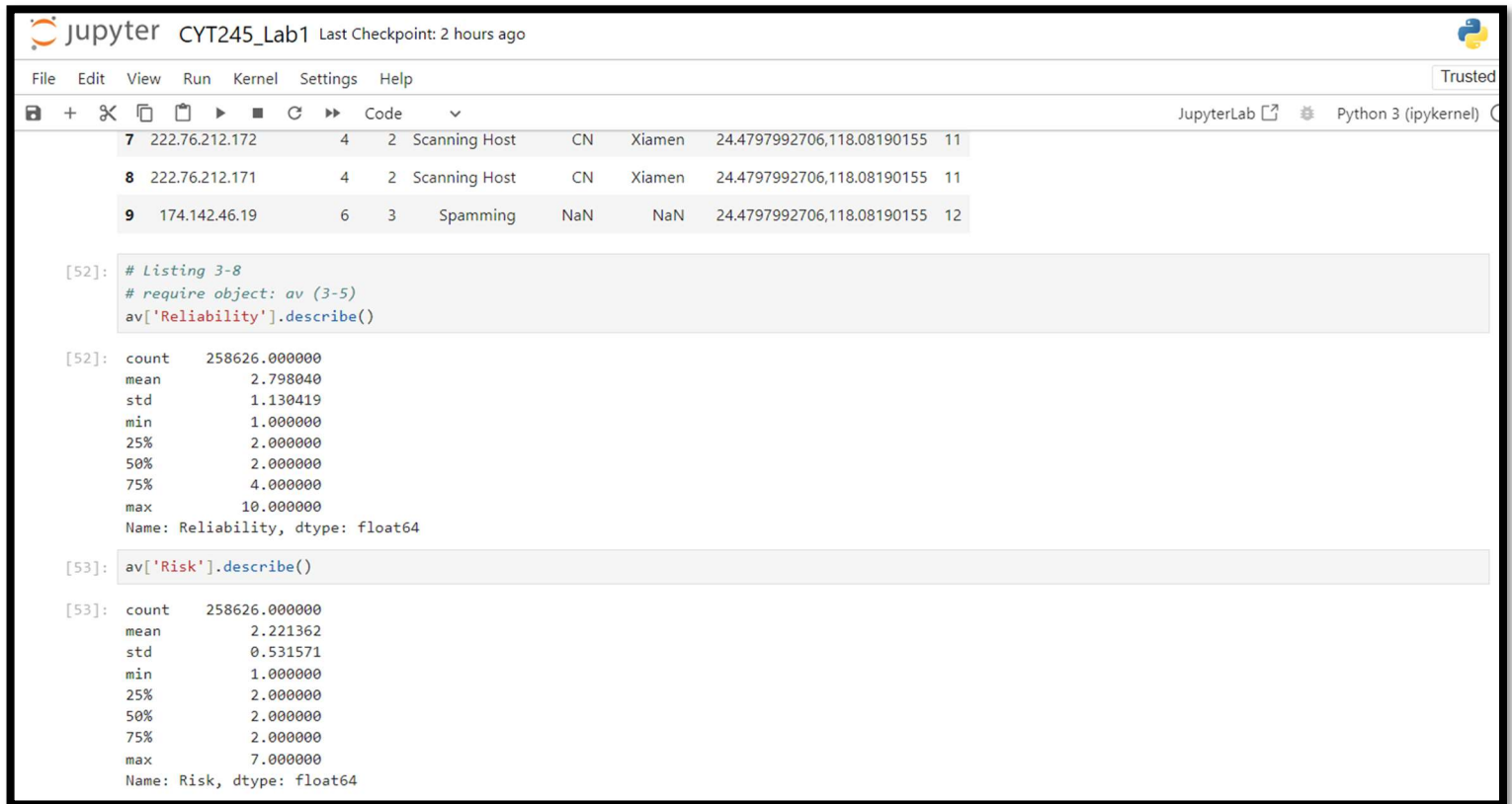
	IP	Reliability	Risk	Type	Country	Locale	Coords	x
0	222.76.212.189	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
1	222.76.212.185	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
2	222.76.212.186	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
3	5.34.246.67	6	3	Spamming	US	NaN	38.0,-97.0	12
4	178.94.97.176	4	5	Scanning Host	UA	Merefa	49.8230018616,36.0507011414	11
5	66.2.49.232	4	2	Scanning Host	US	Union City	37.5962982178,-122.065696716	11
6	222.76.212.173	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
7	222.76.212.172	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
8	222.76.212.171	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
9	174.142.46.19	6	3	Spamming	NaN	NaN	24.4797992706,118.08190155	12

Question:

1. What are Python code line lines that allow doing so (copy and paste from the code)  
⇒ from IPython.display import HTML  
HTML(av.head(10).to\_html())



**Step 6.** Run Listing 3-8. You now start exploring data. This portion of code demonstrates understanding of **quantitative category of** data, in other words, data with values that can be used for calculation. There is a need to generate so called the basic “descriptive statistics” (see the definition below) on the variables. It will be used for reporting and visualization purposes. The Run the code and see the results of calculation.



The screenshot shows a JupyterLab environment with a file named 'CYT245\_Lab1'. The top menu bar includes File, Edit, View, Run, Kernel, Settings, and Help. Below the menu is a toolbar with icons for saving, opening, and running code. The main area displays a table with 11 columns and 3 rows of data. Below the table, there are two code cells. The first cell contains a comment and a call to the describe() method on the 'Reliability' column. The second cell contains a call to the describe() method on the 'Risk' column. The output of the first cell shows a summary of statistics for 'Reliability', and the output of the second cell shows a summary of statistics for 'Risk'.

	7	222.76.212.172	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11
8	222.76.212.171	4	2	Scanning Host	CN	Xiamen	24.4797992706,118.08190155	11	
9	174.142.46.19	6	3	Spamming	NaN	NaN	24.4797992706,118.08190155	12	

```
[52]: # Listing 3-8
# require object: av (3-5)
av['Reliability'].describe()

[52]: count    258626.000000
      mean       2.798040
      std       1.130419
      min       1.000000
      25%       2.000000
      50%       2.000000
      75%       4.000000
      max      10.000000
      Name: Reliability, dtype: float64

[53]: av['Risk'].describe()

[53]: count    258626.000000
      mean     2.221362
      std     0.531571
      min     1.000000
      25%     2.000000
      50%     2.000000
      75%     2.000000
      max     7.000000
      Name: Risk, dtype: float64
```

Answer the following questions:

1. What is the Pandas function to generate descriptive statistics?
  - ⇒ The Pandas function to generate descriptive statistics is describe().
  - ⇒ This function provides a summary of the central tendency, dispersion, and shape of a dataset's distribution, excluding NaN values. It calculates various statistical measures such as count, mean, standard deviation, minimum, quartiles, and maximum for each numerical column in the DataFrame.

### Step 7. Listing 3-10.

It might happen that you receive the syntax error if you run this Listing. More complicated data object definition is used here. It belongs to the **qualitative** category of data. In Pandas this class should be declared as Categorical, and that is not what we prepared to do now. But still, take a look at the code and the result, shown in the book. First you see the results showing the number of malicious nodes calculated by Reliability, Risk, Type, and Country separately. With the last outcome you can see the number of malicious nodes by Country.

```
localhost:8888/notebooks/Untitled%20Folder/CYT245_Lab1.ipynb
jupyter CYT245_Lab1 Last Checkpoint: 2 hours ago
File Edit View Run Kernel Settings Help
+ ✂ 📄 📌 ▶ ■ 🔁 ⏩ Code ▼

Name: Risk, dtype: float64

[54]: # Listing 3-10
def factor_col(col):
    factor = pd.Categorical(col)
    return pd.Series(factor).value_counts().reindex(factor.categories)

rel_ct = pd.Series(av['Reliability']).value_counts()
risk_ct = pd.Series(av['Risk']).value_counts()
type_ct = pd.Series(av['Type']).value_counts()
country_ct = pd.Series(av['Country']).value_counts()

[55]: print(factor_col(av['Reliability']))

1      5612
2     149117
3      10892
4      87040
5           7
6       4758
7        297
8         21
9        686
10       196
Name: count, dtype: int64

[56]: print(factor_col(av['Risk']))

1         39
2      213852
3      33719
4       9588
5       1328
6         90
7         10
Name: count, dtype: int64
```

[56]: `print(factor_col(av['Risk']))`

```
1      39
2  213852
3   33719
4   9588
5   1328
6    90
7    10
Name: count, dtype: int64
```

[57]: `print(factor_col(av['Type']).head(n=10))`

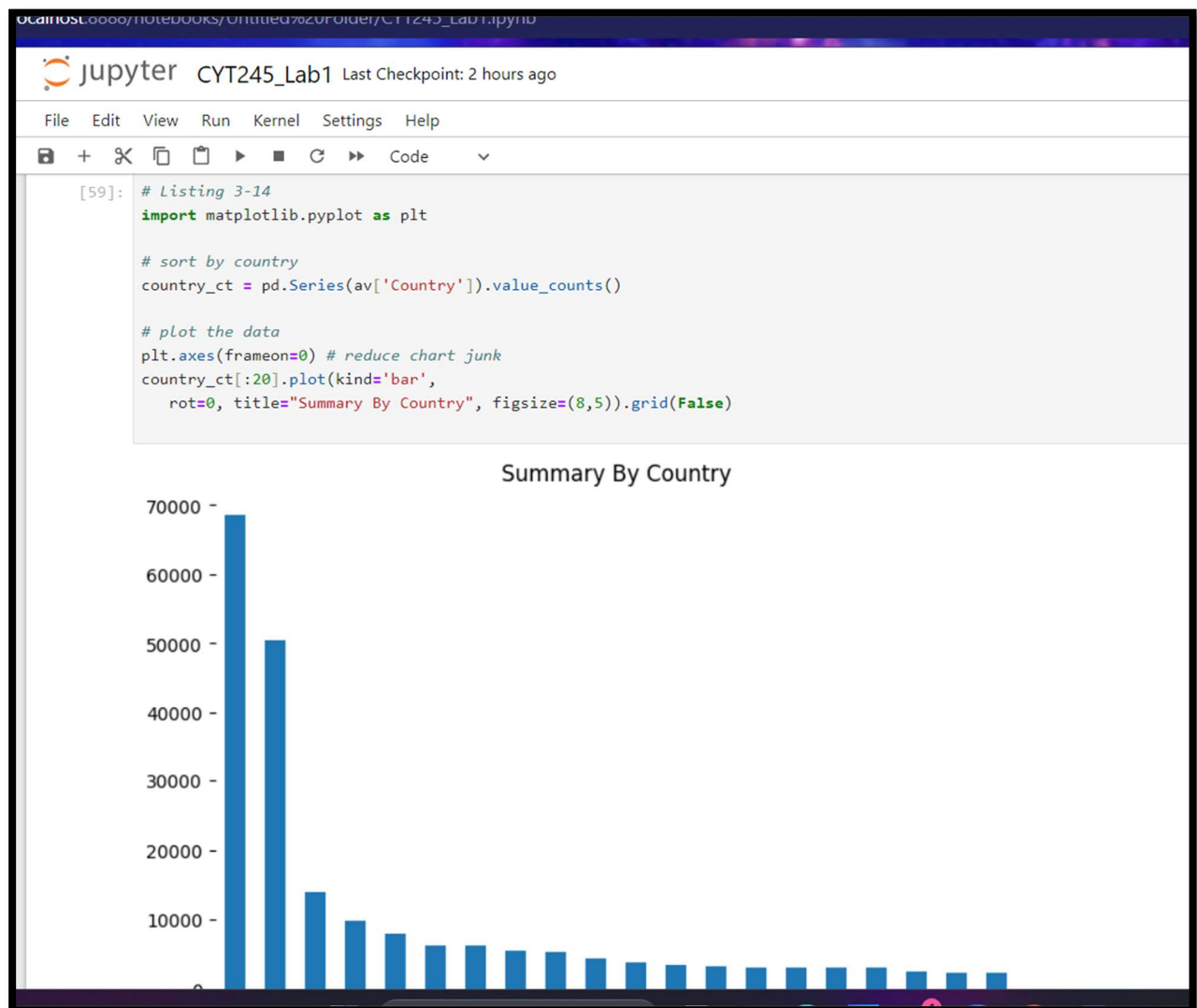
```
APT;Malware Domain      1
C&C                     610
C&C;Malware Domain     31
C&C;Malware IP         20
C&C;Scanning Host       7
Malicious Host        3770
Malicious Host;Malware Domain  4
Malicious Host;Malware IP     2
Malicious Host;Scanning Host  163
Malware Domain         9274
Name: count, dtype: int64
```

[58]: `print(factor_col(av['Country']).head(n=10))`

```
A1      267
A2       2
AE    1827
AL       4
AM       6
AN       3
AO     256
AR    3046
AT      51
AU     155
Name: count, dtype: int64
```



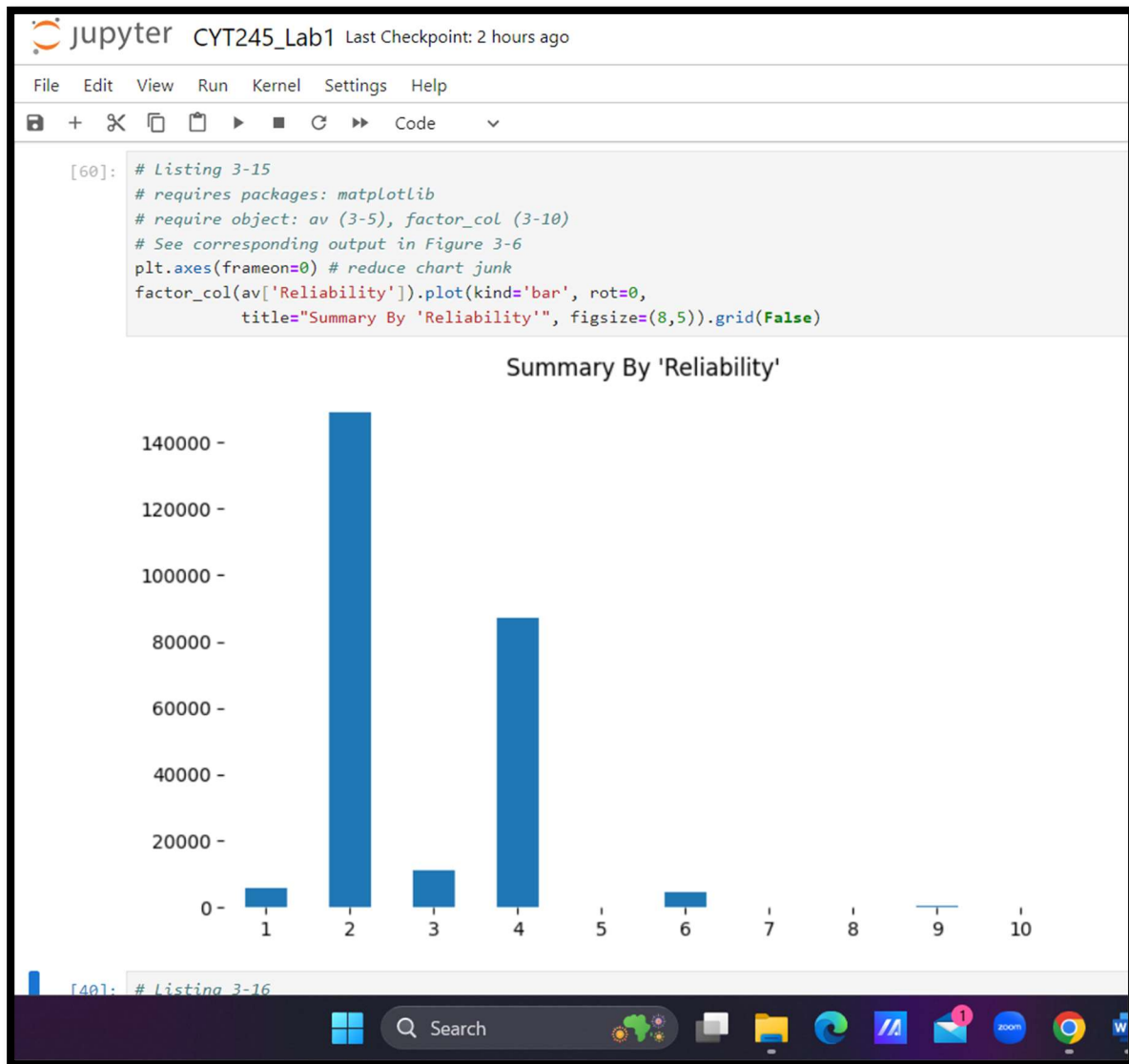
**Step 8.** Run Listing 3-14. Number of records from the data frame will be shown as the graph, named Summary by Country.



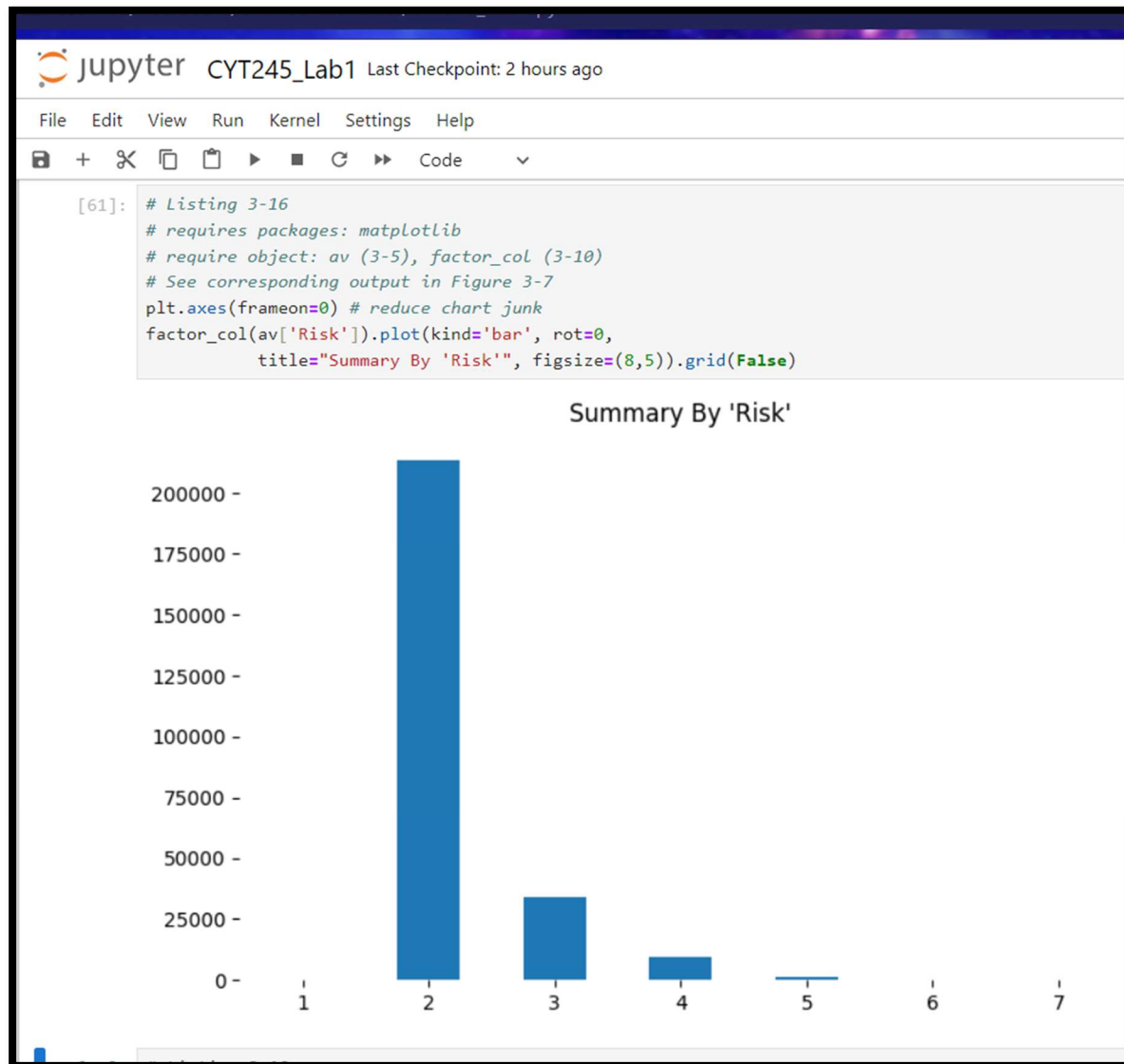
Questions:

- If a country does not have valid country code, will the records be taken for calculation?  
⇒ No, if a country does not have a valid country code or if the country code is missing, the records associated with that country will not be included in the calculation for the graph named "Summary by Country".

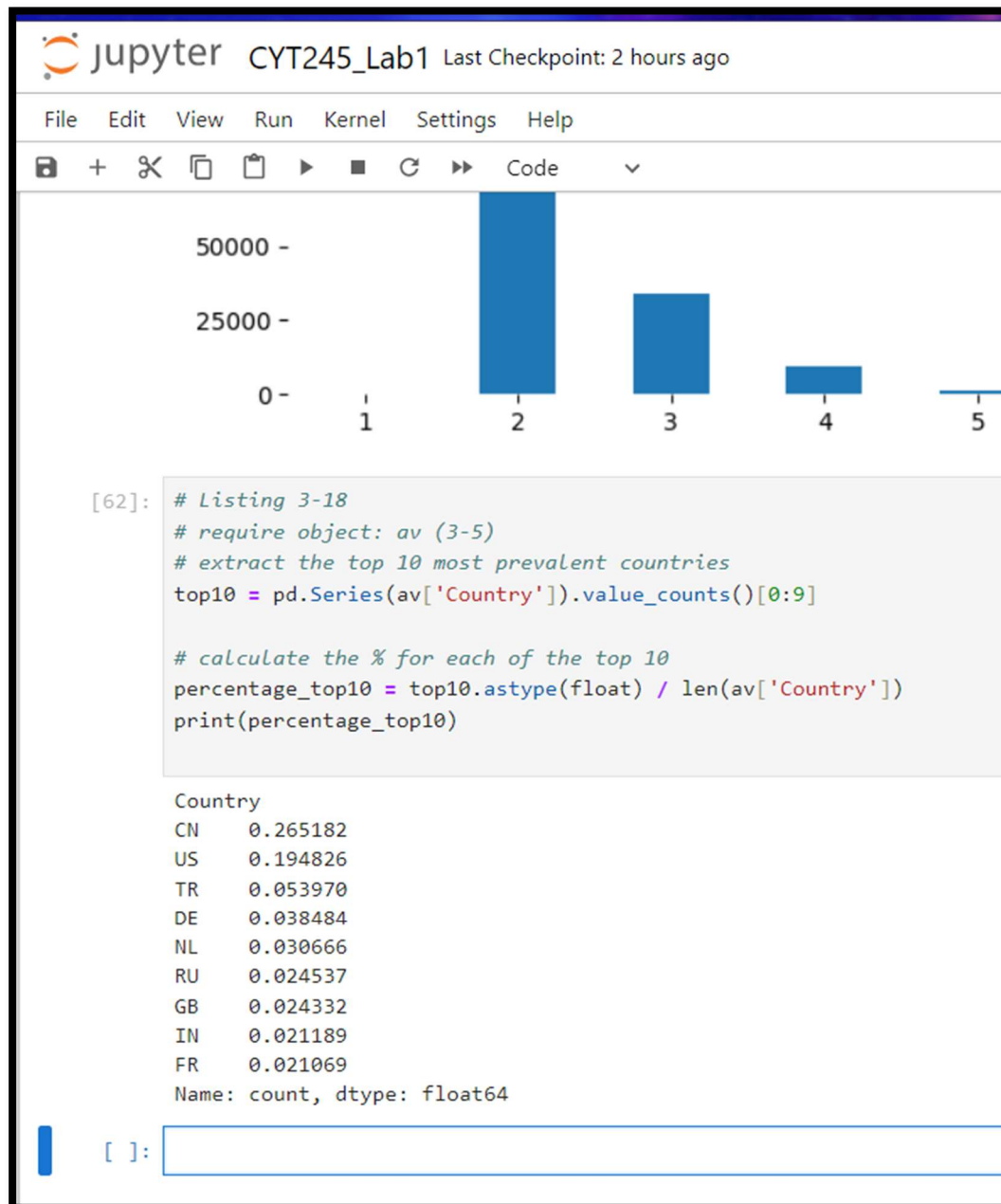
**Step 9.** Listing 3-15. The result shows Reliability chart for top 10 countries (see Figure 3-6).



Step 10. Listing 3-16. The result shows Risk chart for top 10 countries (see Figure 3-7).



**Step 11.** Run Listing 3-18. The result will show data by country in percentage. In this top ten list you will notice that in accordance to this data sample China and US give almost 46% of the malicious nodes in the list.



Question:

- What line of Python code do this calculation (copy and paste)?  
⇒ `Percentage_top10 = top10.astype(float) / len(av['Country'])`

**END of the Lab Workflow**

## **Submission and Rubrics**

- This Lab can be completed individually or as the Team work – up to 4 people. Max Score – 4%
- Submission includes MS Word document uploaded to the BB. The name of the document must follow Submission Upload Requirements (see below).

Submission includes:

- Steps 1 to 11 are run and screenshots are present.
- Answers to the Questions included into the Steps accordingly.
- Full collection of screenshots and correct answers – 3%
- Partially completed screenshots or not correct answers will result in some extraction accordingly (not less than 8 screenshots and right answers)
- Less than 8 screenshots – 2%

### **Submission Upload Requirements**

Make online submission to BB, only one submission from your team.

If you have more than one document, wrap it up to ZIP, 7ZIP, or RAR folder

Name the file you will uploading as indicated below. The name must include:

- Course ID (CYT245)
- What is this (e.g. lab1, assignment 1, etc )
- Authors by name(s)

**Sample: CYT245MLab1\_PeterJohnMohammadSue**

**Note: submissions that do not follow the requirements will not be accepted**