Peer Review Assignment - Data Engineer - Webscraping

Estimated time needed: **20** minutes

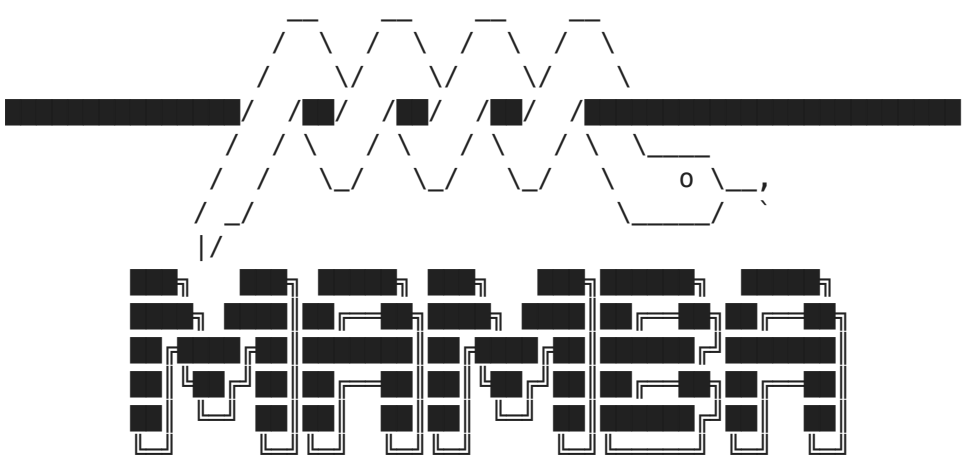
Objectives

In this part you will:

- Use webscraping to get bank information

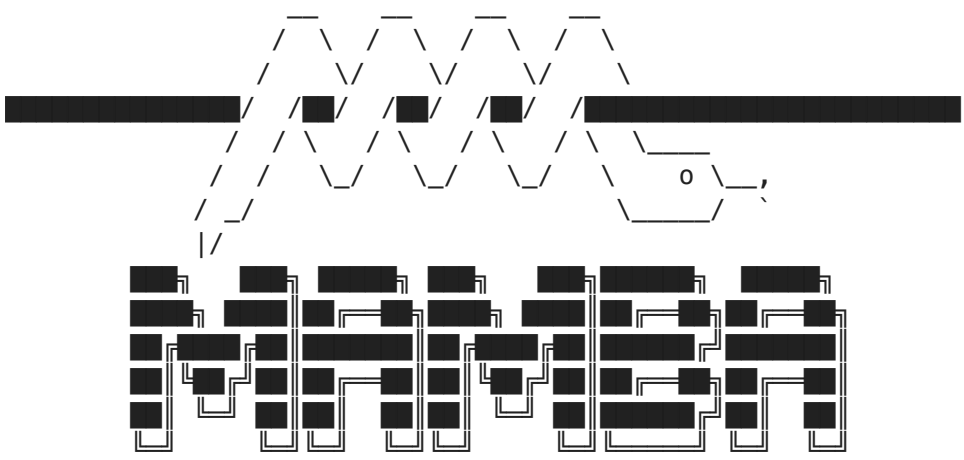
For this lab, we are going to be using Python and several Python libraries. Some of these libraries might be installed in your lab environment or in SN Labs. Others may need to be installed by you. The cells below will install these libraries when executed.

In [1]:
```
#!mamba install pandas==1.3.3 -y
#!mamba install requests==2.26.0 -y
!mamba install bs4==4.10.0 -y
!mamba install html5lib==1.1 -y
```



```
mamba (0.15.3) supported by @QuantStack

GitHub:  https://github.com/mamba-org/mamba
Twitter: https://twitter.com/QuantStack
```

```
Looking for: ['bs4==4.10.0']

pkgs/main/linux-64       [>                   ] (--:--) No change
pkgs/main/linux-64       [====================] (00m:00s) No change
pkgs/main/noarch         [>                   ] (--:--) No change
pkgs/main/noarch         [====================] (00m:00s) No change
pkgs/r/linux-64          [>                   ] (--:--) No change
pkgs/r/linux-64          [====================] (00m:00s) No change
pkgs/r/noarch            [>                   ] (--:--) No change
pkgs/r/noarch            [====================] (00m:00s) No change

Pinned packages:
  - python 3.7.*


Transaction

  Prefix: /home/jupyterlab/conda/envs/python

  All requested packages already installed
```



```
        mamba (0.15.3) supported by @QuantStack

        GitHub:  https://github.com/mamba-org/mamba
        Twitter: https://twitter.com/QuantStack
```

```
Looking for: ['html5lib==1.1']

pkgs/main/linux-64       Using cache
pkgs/main/noarch         Using cache
pkgs/r/linux-64          Using cache
pkgs/r/noarch            Using cache

Pinned packages:
  - python 3.7.*


Transaction

  Prefix: /home/jupyterlab/conda/envs/python

  All requested packages already installed
```

Imports

Import any additional libraries you may need here.

In [13]:
```
from bs4 import BeautifulSoup
import html5lib
import requests
import pandas as pd
```

Extract Data Using Web Scraping

The wikipedia webpage https://en.wikipedia.org/wiki/List_of_largest_banks provides information about largest banks in the world by various parameters. Scrape the data from the table 'By market capitalization' and store it in a JSON file.

Webpage Contents

Gather the contents of the webpage in text format using the `requests` library and assign it to the variable `html_data`

In [14]:
```
#Write your code here
url = "https://en.wikipedia.org/wiki/List_of_largest_banks"
html_data = requests.get(url).text
```

Question 1 Print out the output of the following line, and remember it as it will be a quiz question:

In [15]:
```
html_data[101:124]
```

Out[15]:
```
'List of largest banks –'
```

Scraping the Data

Question 2 Using the contents and `beautiful soup` load the data from the `By market capitalization` table into a `pandas` dataframe. The dataframe should have the bank `Name` and `Market Cap (US$ Billion)` as column names. Display the first five rows using head.

Using BeautifulSoup parse the contents of the webpage.

In [16]:
```
#Replace the dots below
soup = BeautifulSoup(html_data,"html.parser")
```

Load the data from the `By market capitalization` table into a pandas dataframe. The dataframe should have the bank `Name` and `Market Cap (US$ Billion)` as column names. Using the empty dataframe `data` and the given loop extract the necessary data from each row and append it to the empty dataframe.

In [22]:
```
data = pd.DataFrame(columns=["Name", "Market Cap (US$ Billion)"])
for row in soup.find_all('tbody')[2].find_all('tr'):
    cols = row.find_all('td')
    if len(cols) == 0:
                continue
    else:
                data = data.append({#'Rank': col[0].text.strip(),
                          'Name': cols[1].text.strip(),
                          'Market Cap (US$ Billion)': cols[2].text.strip()
```

Question 3 Display the first five rows using the `head` function.

In [24]:
```
#Write your code here
data.head()
```

Out[24]:

	Name	Market Cap (US$ Billion)
0	JPMorgan Chase	368.78
1	Industrial and Commercial Bank of China	295.65
2	Bank of America	279.73
3	Wells Fargo	214.34
4	China Construction Bank	207.98

Loading the Data

Usually you will Load the `pandas` dataframe created above into a JSON named `bank_market_cap.json` using the `to_json()` function, but this time the data will be sent to another team who will split the data file into two files and inspect it. If you save the data it will interfere with the next part of the assignment.

In [ ]:
```
#Write your code here
```

Authors

Ramesh Sannareddy, Joseph Santarcangelo and Azim Hirjani

Other Contributors

Rav Ahuja

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2022-07-12	0.2	Appalabhaktula Hema	Corrected the code and markdown
2020-11-25	0.1	Ramesh Sannareddy	Created initial version of the lab