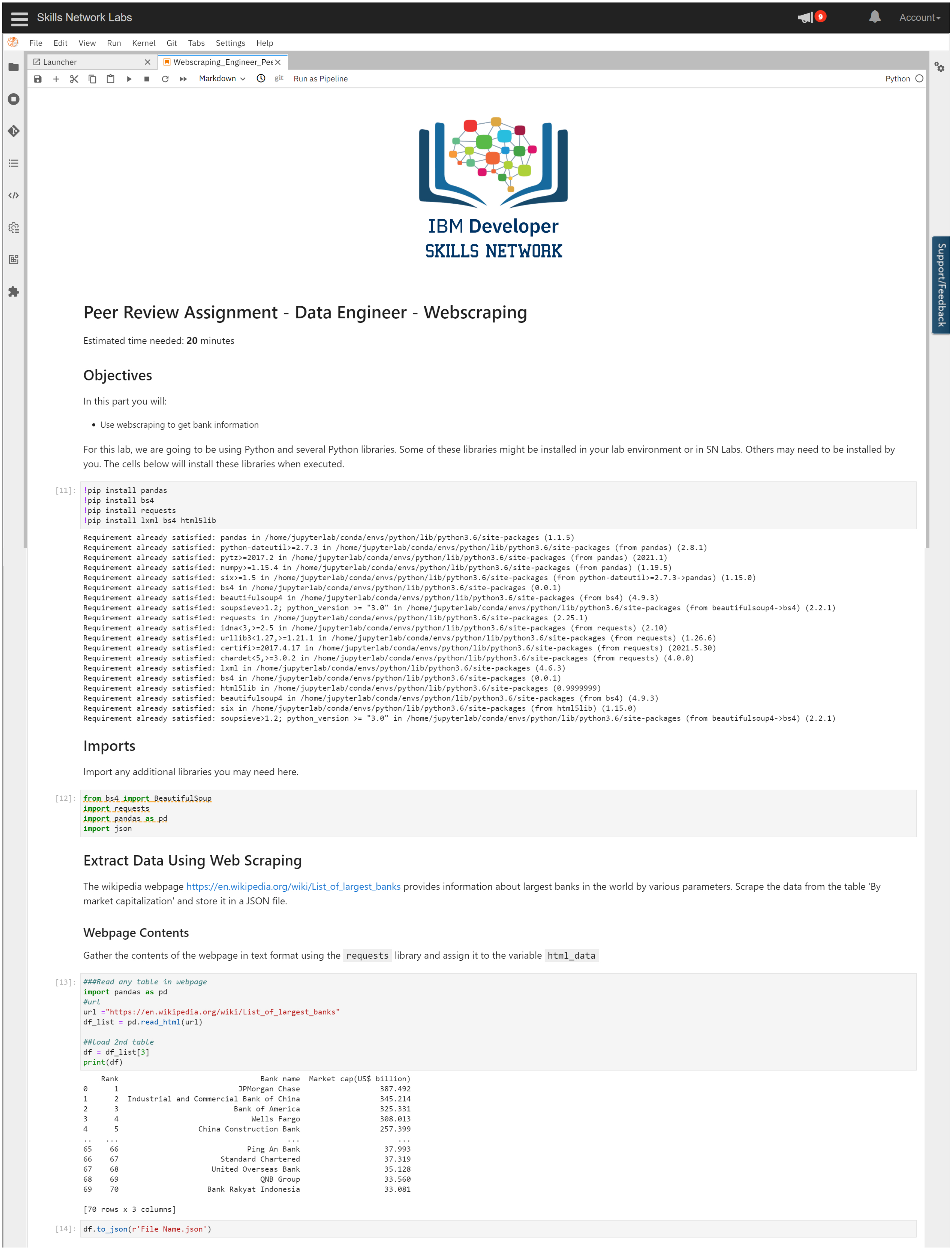

Peer Review Assignment - Data Engineer - Webscraping

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.

[11]:
```
!pip install pandas
!pip install bs4
!pip install requests
!pip install lxml bs4 html5lib
```

```
Requirement already satisfied: pandas in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (1.1.5)
Requirement already satisfied: python-dateutil>=2.7.3 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from pandas) (2.8.1)
Requirement already satisfied: pytz>=2017.2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from pandas) (2021.1)
Requirement already satisfied: numpy>=1.15.4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from pandas) (1.19.5)
Requirement already satisfied: six>=1.5 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from python-dateutil>=2.7.3->pandas) (1.15.0)
Requirement already satisfied: bs4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (0.0.1)
Requirement already satisfied: beautifulsoup4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from bs4) (4.9.3)
Requirement already satisfied: soupsieve>1.2; python_version >= "3.0" in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from beautifulsoup4->bs4) (2.2.1)
Requirement already satisfied: requests in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (2.25.1)
Requirement already satisfied: idna<3,>=2.5 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (2.10)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (1.26.6)
Requirement already satisfied: certifi>=2017.4.17 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (2021.5.30)
Requirement already satisfied: chardet<5,>=3.0.2 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from requests) (4.0.0)
Requirement already satisfied: lxml in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (4.6.3)
Requirement already satisfied: bs4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (0.0.1)
Requirement already satisfied: html5lib in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (0.9999999)
Requirement already satisfied: beautifulsoup4 in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from bs4) (4.9.3)
Requirement already satisfied: six in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from html5lib) (1.15.0)
Requirement already satisfied: soupsieve>1.2; python_version >= "3.0" in /home/jupyterlab/conda/envs/python/lib/python3.6/site-packages (from beautifulsoup4->bs4) (2.2.1)
```

Imports

Import any additional libraries you may need here.

[12]:
```
from bs4 import BeautifulSoup
import requests
import pandas as pd
import json
```

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`

[13]:
```
###Read any table in webpage
import pandas as pd
#url
url ="https://en.wikipedia.org/wiki/List_of_largest_banks"
df_list = pd.read_html(url)

##load 2nd table
df = df_list[3]
print(df)
```

```
    Rank                                Bank name  Market cap(US$ billion)
0      1                           JPMorgan Chase                  387.492
1      2  Industrial and Commercial Bank of China                  345.214
2      3                          Bank of America                  325.331
3      4                              Wells Fargo                  308.013
4      5                  China Construction Bank                  257.399
..   ...                                      ...                      ...
65    66                             Ping An Bank                   37.993
66    67                       Standard Chartered                   37.319
67    68                     United Overseas Bank                   35.128
68    69                                QNB Group                   33.560
69    70                    Bank Rakyat Indonesia                   33.081

[70 rows x 3 columns]
```

[14]:
```
df.to_json(r'File Name.json')
```

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

[15]:
```
df[101:124]
```

[15]:

Rank	Bank name	Market cap(US$ billion)

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 country `Name` and `Market Cap (US$ Billion)` as column names. Display the first five rows using head.

Using BeautifulSoup parse the contents of the webpage.

[16]:
```
#Replace the dots below
soup=print(df.head())
```

```
   Rank                           Bank name  Market cap(US$ billion)
0     1                      JPMorgan Chase                  387.492
1     2  Industrial and Commercial Bank of China                  345.214
2     3                     Bank of America                  325.331
3     4                         Wells Fargo                  308.013
4     5            China Construction Bank                  257.399
```

Load the data from the `By market capitalization` table into a pandas dataframe. The dataframe should have the country `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.

[23]:
```
#data = pd.DataFrame(columns=["Name", "Market Cap (US$ Billion)"])

#for row in soup.find_all('tbody')[3].find_all('tr'):
#    col = row.find_all('td')
#    if len(col) == 0:
#        continue
#    else:
#        df1 = df1.append({'Rank': col[0].text.strip(),
#                          'Bank name': col[1].text.strip(),
#                          'Market cap(US$ billion)': col[2].text.strip()}, ignore_index=True)
```

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

[24]:
```
#Write your code here
print(df.head())
```

```
   Rank                           Bank name  Market cap(US$ billion)
0     1                      JPMorgan Chase                  387.492
1     2  Industrial and Commercial Bank of China                  345.214
2     3                     Bank of America                  325.331
3     4                         Wells Fargo                  308.013
4     5            China Construction Bank                  257.399
```

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.

[25]:
```
#Write your code here
print(df)
```

```
    Rank                           Bank name  Market cap(US$ billion)
0      1                      JPMorgan Chase                  387.492
1      2  Industrial and Commercial Bank of China                  345.214
2      3                     Bank of America                  325.331
3      4                         Wells Fargo                  308.013
4      5            China Construction Bank                  257.399
..   ...                                 ...                      ...
65    66                        Ping An Bank                   37.993
66    67                   Standard Chartered                   37.319
67    68                United Overseas Bank                   35.128
68    69                           QNB Group                   33.560
69    70              Bank Rakyat Indonesia                   33.081

[70 rows x 3 columns]
```

Authors

Ramesh Sannareddy, Joseph Santarcangelo and Azim Hirjani

Other Contributors

Rav Ahuja

Change Log

Date (YYYY-MM-DD)	Version	Changed By	Change Description
2020-11-25	0.1	Ramesh Sannareddy	Created initial version of the lab

Simple 0 1 Python | Idle Fully initialized Mem: 253.67 / 6144.00 MB Saving completed Mode: Command Ln 1, Col 1 English (American) Webscraping_Engineer_Peer_Review_Assignment.ipynb