Forward School

Program Code: J620-002-4:2020

Program Name: FRONT-END SOFTWARE DEVELOPMENT

Title : Exercise using BeautifulSoup and Selenium

Name: Ooi Caaron

IC Number: 990701-07-5837

Date : 4/7/23

Introduction : Exercise using BeautifulSoup and Selenium is a practical exercise that involves utilizing two powerful tools, BeautifulSoup and Selenium, for web scraping and web automation tasks.

Conclusion : Still need to practice more and do revision

Exe09 - Exercise Using BeautifulSoup and Selenium on News Web Portal

Extract daily COVID-19 statistics from theStar

Location: https://www.thestar.com.my/news/nation/2020/03/23/covid-19-current-situation-in-malaysia-updated-daily (https://www.thestar.com.my/news/nation/2020/03/23/covid-19-current-situation-in-malaysia-updated-daily)

In [7]:

```
import requests
from bs4 import BeautifulSoup

url='https://www.thestar.com.my/news/nation/2020/03/23/covid-19-current-situation-in-mal

# get the webpage
html = requests.get(url)
# load webpage into bs4
bs = BeautifulSoup(html.content, 'html.parser')
# get data simply by looking for all <a> links
for link in bs.find_all('a'):
    print(link)
#     if 'href' in link.attrs:
#         print(link.attrs['href'])
```

```
<a class="navbar-brand brand-prime" data-content-id="https://www.thesta
r.com.my" data-content-title="The Star Online" data-content-type="Navig
ation" data-list-type="Header" href="/">
<svg aria-label="the star online" class="icon" height="55" role="img" w
idth="164">
<image border="0" height="55" src="https://cdn.thestar.com.my/Themes/im
g/logo-tsol-logov3.png" width="164" xlink:href="https://cdn.thestar.co
m.my/Themes/img/logo-tsol-fullv3.svg"/>
</svg>
</a>
<a class="btn--subscribe" data-content-id="https://www.thestar.com.my/s
ubscription" data-content-title="Subscription" data-content-type="Navig
ation" data-list-type="Header" href="/subscription">Subscriptions</a>
<a class="login" data-content-id="https://sso.thestar.com.my/?lng=en&am
p;channel=1&amp;ru=HNQ8Auw31qgZZU47ZjHUhHKJStkK3H51/pPcFdJ1gQ9cFgPiSala
sDvF6DeumuZwrPFzdYjofJj9eX1n44olyqGHD3HJYujVJKnBGSMMB/zfChfXgzd4SeyxRdN
XN6ZWbrt8Vq9CGyeRv3tJQMZkgrPs0PgxqXZTlEZW/jQG2aZ+b1eksd4EfiZDBUcWQcFYvs
1m3Fkd04fguPM90q6guFbCG4ZqfYK1HTduYl2eQNi53cvg+bra/Y0o0cgRGLoa7eTLY69Y
N/+roj7uviwmtQ==" data-content-title="Log In" data-content-type="Outbou
```

In [11]:

```
import requests
from bs4 import BeautifulSoup

url='https://www.thestar.com.my/news/nation/2020/03/23/covid-19-current-situation-in-mal

# get the webpage


# load webpage into bs4
bs = BeautifulSoup(html.content,'html.parser')

# get data simply by looking for all <a> links
for link in bs.find_all('a'):
    print(link.text)
```

```
Subscriptions

                        Log In


Manage Profile

Change Password

Manage Logins

Manage Subscription

Transaction History
```

Check HTML code of the Web page again

Notice that there is an iFrame Tag highlighted above?

The actual location of the source web page is embeded within the iframe of theStar

Change the URL to the actual source.

In [1]:

```
import requests
from bs4 import BeautifulSoup

url='https://public.flourish.studio/visualisation/1641110/embed?auto=1'

# get the data
driver.get(url)

# load data into bs4
soup = BeautifulSoup(driver.page_source,'html.parser')

data=[]
# get data simply by looking for each a links
for tr in soup.find_all('div',attrs={'class':'tr body-row'}):
    data.append(tr.text)

driver.close()
data
```

```
---------------------------------------------------------------------------
-
NameError                                 Traceback (most recent call las
t)
Cell In[1], line 7
      4 url='https://public.flourish.studio/visualisation/1641110/embed?au
to=1'
      6 # get the data
----> 7 drver.get(url)
      9 # load data into bs4
     10 soup = BeautifulSoup(driver.page_source,'html.parser')

NameError: name 'drver' is not defined
```

In [13]:

```
# soup.find_all('div')
```

Out[13]:

```
<bound method Tag.prettify of <!DOCTYPE html>
<html><head>
<meta charset="utf-8"/>
<meta content="width=device-width, initial-scale=1" name="viewport"/>
<base target="_blank"/>
<link href="https://flo.uri.sh/template/1065/v24/static/style.css" rel
="stylesheet" type="text/css"/>
<link href="https://fonts.googleapis.com/css?family=Source+Sans+Pro:40
0,700" rel="stylesheet" type="text/css"/>
<title>COVID-19 MALAYSIA TABLE</title></head>
<body>
<style id="cell-styling"></style>
<script>window.Flourish = {"static_prefix":"https://flo.uri.sh/templat
e/1065/v24/static","environment":"live"};</script><script>var template=
function(t){"use strict";var s={},f={table_min_width:300,table_border_c
olor:"#aaaaaa",table_border_width:0,sorting:{enabled:!0,order:"ascendin
g",column_index:null},reloader:{},color:{custom_palette:"Clinton:#1d699
6\nTrump:#cc503e"},popup:{font_size:"1rem"},bar_columns:{enabled:!0,typ
```

Cannot Use BeautifulSoup

Check the Javascript found above.

The data for the table is within the Javascript coding.

2 options.

Option 1. Try to Scrape the Javascript. Not that possible, unless fully understand how the Javascript program going to output the HTML to the Web Page.

Option 2. Use Selenium Webdriver to run the Javascript within the webdriver and then scrape the HTML output.

In [2]:

```
# Use Selenium
from selenium import webdriver
from bs4 import BeautifulSoup

driver = webdriver.Chrome('C://GoogleDriver//chromedriver')

url='https://public.flourish.studio/visualisation/1641110/embed?auto=1'

# get the data
driver.get(url)

# load data into bs4
soup = BeautifulSoup(driver.page_source,'html.parser')
data = []

# get data simply by looking for each a links
for tr in soup.find_all('div', attrs={'class':'tr body-row'}):
    data.append(tr.text)

driver.close()
data
```

Out[2]:

```
['22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n']
```

In [3]:

```
from selenium import webdriver
from bs4 import BeautifulSoup

driver = webdriver.Chrome('C://GoogleDriver//chromedriver')

url='https://public.flourish.studio/visualisation/1641110/embed?auto=1'

# get the data
driver.get(url)

# load data into bs4
soup = BeautifulSoup(driver.page_source,'html.parser')
data = []

# get data simply by looking for each a links
for tr in soup.find_all('div', attrs={'class':'tr body-row'}):
    for td in soup.find_all('div', attrs={'class':'tr'}):
        data.append(td.text)

driver.close()
data
```

Out[3]:

```
['DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n',
 'DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n',
 'DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n',
 'DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n',
 'DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n',
 'DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n',
 'DateTotal casesNew casesTotal deathsTotal recovered',
 '22-Apr-21\n384688\n2875\n1407\n361267\n',
 '21-Apr-21\n381813\n2340\n1400\n358726\n',
 '20-Apr-21\n379473\n2341\n1389\n356816\n',
 '19-Apr-21\n377132\n2078\n1386\n355224\n',
 '18-Apr-21\n375054\n2195\n1378\n353822\n',
 '17-Apr-21\n372859\n2331\n1370\n352395\n',
 '16-Apr-21\n370528\n2551\n1365\n350563\n']
```

In [32]:

```
from selenium import webdriver
from bs4 import BeautifulSoup
import time
driver = webdriver.Chrome('C://GoogleDriver//chromedriver')

url='https://public.flourish.studio/visualisation/1641110/embed?auto=1'

# get the data
driver.get(url)


# soup = BeautifulSoup(driver.page_source,'html.parser')
# data=[]

# for tr in soup.find_all('div', attrs={'class':'tr body-row'}):
#     for td in soup.find_all('div', attrs={'class':'td'}):
#         data.append(td.text.rstrip())

# data


data = []

for page in range(1, 17):
    soup = BeautifulSoup(driver.page_source, 'html.parser')
    rows = soup.find_all("div", class_="tr body-row")

    for i in range(len(rows)):
        cells = rows[i].find_all("div", class_="td")
        row_data = {
            'Date': '',
            'Total Cases': '',
            'New Cases': '',
            'Total Deaths': '',
            'Total Recovered': ''
        }
        row_data['Date'] = cells[0].text.strip()
        row_data['Total Cases'] = cells[1].text.strip()
        row_data['New Cases'] = cells[2].text.strip()
        row_data['Total Deaths'] = cells[3].text.strip()
        row_data['Total Recovered'] = cells[4].text.strip()
        data.append(row_data)

    next_button = driver.find_element_by_xpath('//*[@id="pagination"]/button[2]')
    if next_button.get_attribute('aria-disabled') == 'true':
        break
    next_button.click()



# for i, d in enumerate(data):
#     print(f"Comment {i+1}: {d}")
#     print('')
```

In [33]:

```
import pandas as pd
df = pd.DataFrame(data)

# Display the dataframe
df
```

Out[33]:

	Date	Total Cases	New Cases	Total Deaths	Total Recovered
0	22-Apr-21	384688	2875	1407	361267
1	21-Apr-21	381813	2340	1400	358726
2	20-Apr-21	379473	2341	1389	356816
3	19-Apr-21	377132	2078	1386	355224
4	18-Apr-21	375054	2195	1378	353822
...	...	...	...	...	...
107	5-Jan-21	122845	2027	509	99449
108	4-Jan-21	120818	1741	501	98228
109	3-Jan-21	119077	1704	494	97218
110	2-Jan-21	117373	2295	483	94492
111	1-Jan-21	115078	2068	474	91171

112 rows × 5 columns

In [25]:

```
# Next Page
driver.find_element_by_xpath('/html/body/main/section[4]/div[1]/div/div[4]/button[2]').c

soup = BeautifulSoup(driver.page_source,'html.parser')

data=[]
# get data simply by looking for each a links
for tr in soup.find_all('div', attrs={'class':'tr body-row'}):
    for td in soup.find_all('div', attrs={'class':'td'}):
        data.append(td.text)

data

# depends
# if first time scrape, must scrape all previous pages. then paginate and get those data
# if only need to get the latest everyday, then no need to grab the same data all over a

# look at this class="pagination-total"
```

Out[25]:

```
['Date',
 'Total cases',
 'New cases',
 'Total deaths',
 'Total recovered',
 '17-Dec-20\n',
 '89133\n',
 '1220\n',
 '432\n',
 '74030\n',
 '16-Dec-20\n',
 '87913\n',
 '1295\n',
 '429\n',
 '72733\n',
 '15-Dec-20\n',
 '86618\n',
 '1772\n',
```

Footnote:

HTML iframe tag

Specification:

https://www.w3.org/html/wg/spec/the-iframe-element.html (https://www.w3.org/html/wg/spec/the-iframe-element.html)

EXERCISE: -Scrape table on this URL: "" -Use Selenium to scrape data -Scrape data from 1st Jan 2021 until 20th Mar 2021 -Use drive.click() to navigate pagination -Feel free to drop me questions/Google/refer notes during this exercise.