Forward School

Program Code: J620-002-4:2020

Program Name: FRONT-END SOFTWARE DEVELOPMENT

Title : Exercise using BeautifulSoup and Selenium

Name: Chuay Xiang Ze

IC Number: 021224070255

Date : 6/7/2023

Introduction : Learning how to use things learnt from lessons of BeautifulSoup and Selenium to scrape data on a news webiste.

Conclusion : Managed to complete tasks relating to the topic.

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

In [19]:

```
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
data = requests.get(url)

# load webpage into bs4
soup = BeautifulSoup(data.text, 'html.parser')

# get data simply by looking for all <a> links
soup.find_all('a')
```

Out[19]:

```
[<a class="navbar-brand brand-prime" data-content-id="https://www.thest
ar.com.my" data-content-title="The Star Online" data-content-type="Navi
gation" data-list-type="Header" href="/">
 <svg aria-label="the star online" class="icon" height="55" role="img"
width="164">
 <image border="0" height="55" src="https://cdn.thestar.com.my/Themes/i
mg/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/
subscription" data-content-title="Subscription" data-content-type="Navi
gation" data-list-type="Header" href="/subscription">Subscriptions</a>,
 <a class="login" data-content-id="https://sso.thestar.com.my/?lng=en&a
mp;channel=1&amp;ru=HNQ8Auw31qgZZU47ZjHUhHKJStkK3H51/pPcFdJ1gQ9cFgPiSal
asDvF6DeumuZwrPFzdYjofJj9eX1n44olyqGHD3HJYujVJKnBGSMMB/zfChfXgzd4SeyxRd
NXN6ZWbrt8Vq9CGyeRv3tJQMZkgrPs0PgxqXZTlEZW/jQG2aZ+b1eksd4EfiZDBUcWQcFYv
s1m3Fkd04fguPM90g6guFbCG4ZqfYK1HTduYl2eQNj53cvg+bra/Y0o0cgRGLoa7eTLY69Y
```

In [22]:

```python
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
data = requests.get(url)

# load webpage into bs4
soup = BeautifulSoup(data.text, 'html.parser')

data = []
# get data simply by looking for all <a> links
for tr in soup.find_all('a'):
    data.append(tr.text)
data
```

Out[22]:

```
['\n\n\n\n',
 'Subscriptions',
 '\n                        Log In\n                    ',
 '',
 'Manage Profile\n                                ',
 'Change Password\n                                ',
 'Manage Logins\n                                ',
 'Manage Subscription\n                                ',
 'Transaction History\n                                ',
 'Manage Billing Info\n                                ',
 'Manage For You\n                                ',
 'Manage Bookmarks\n                                ',
 'Package & Pricing\n                                ',
 'FAQs\n                                ',
 'Log Out\n                                ',
 '\n\n\n\n',
 '\n\n',
 '\n                    StarPlus\n                    ',
```

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 [5]:

```
import requests
from bs4 import BeautifulSoup

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

# get the data
data = requests.get(url)

# load data into bs4
soup = BeautifulSoup(data.text, '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)

data
```

Out[5]:

```
[]
```

In [6]:

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

Out[6]:

```
<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 [8]:

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

driver = webdriver.Chrome('C:\\Users\Xiang Ze\Downloads\chromedriver_win32\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[8]:

```
['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 [10]:

```
from selenium import webdriver
from bs4 import BeautifulSoup

driver = webdriver.Chrome('C:\\Users\Xiang Ze\Downloads\chromedriver_win32\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': 'td'}):
        data.append(td.text)

driver.close()
data
```

Out[10]:

```
['Date',
 'Total cases',
 'New cases',
 'Total deaths',
 'Total recovered',
 '22-Apr-21\n',
 '384688\n',
 '2875\n',
 '1407\n',
 '361267\n',
 '21-Apr-21\n',
 '381813\n',
 '2340\n',
 '1400\n',
 '358726\n',
 '20-Apr-21\n',
 '379473\n',
 '2341\n',
```

In [11]:

```
from selenium import webdriver
from bs4 import BeautifulSoup

driver = webdriver.Chrome('C:\\Users\Xiang Ze\Downloads\chromedriver_win32\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': 'td'}):
        data.append(td.text.rstrip())
data
```

Out[11]:

```
['Date',
 'Total cases',
 'New cases',
 'Total deaths',
 'Total recovered',
 '22-Apr-21',
 '384688',
 '2875',
 '1407',
 '361267',
 '21-Apr-21',
 '381813',
 '2340',
 '1400',
 '358726',
 '20-Apr-21',
 '379473',
 '2341',
```

In [16]:

```
# 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.rstrip())
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[16]:

```
['Date',
 'Total cases',
 'New cases',
 'Total deaths',
 'Total recovered',
 '18-Mar-21',
 '328466',
 '1213',
 '1223',
 '312461',
 '17-Mar-21',
 '327253',
 '1219',
 '1220',
 '310958',
 '16-Mar-21',
 '326034',
 '1063',
```

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)

In [42]:

```
# EXERCISE:
#     -Scrape table on this URL: "https://public.flourish.studio/visualisation/1641110/e
#     -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.

from datetime import datetime
from selenium import webdriver
from bs4 import BeautifulSoup
import time
import pandas as pd

driver = webdriver.Chrome('C:\\Users\Xiang Ze\Downloads\chromedriver_win32\chromedriver.
url = 'https://public.flourish.studio/visualisation/1641110/embed?auto=1'
driver.get(url)

data = []

for page in range(1, 17):
    soup = BeautifulSoup(driver.page_source, 'html.parser')

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

    next_button = driver.find_element_by_xpath('/html/body/main/section[4]/div[1]/div/di

driver.quit()

df_data = []
for i in range(0, len(data), 5):
    date_str = data[i]
    total_cases = data[i+1]
    new_cases = data[i+2]
    total_deaths = data[i+3]
    total_recovered = data[i+4]

    date = datetime.strptime(date_str, '%d-%b-%y')
    if date >= datetime(2021, 1, 1) and date <= datetime(2021, 3, 20):
        df_data.append([date_str, total_cases, new_cases, total_deaths, total_recovered]

df = pd.DataFrame(df_data, columns=['Date', 'Total Cases', 'New Cases', 'Total Deaths',
df = df.set_index("Date")
df
```

Out[42]:

	Total Cases	New Cases	Total Deaths	Total Recovered
Date				
1-Jan-21	115078	2068	474	91171
2-Jan-21	117373	2295	483	94492
3-Jan-21	119077	1704	494	97218
4-Jan-21	120818	1741	501	98228
5-Jan-21	122845	2027	509	99449
...	...	...	...	...
16-Mar-21	326034	1063	1218	309612
17-Mar-21	327253	1219	1220	310958
18-Mar-21	328466	1213	1223	312461
19-Mar-21	330042	1576	1225	314457
20-Mar-21	331713	1671	1229	316042

79 rows × 4 columns

In [ ]: