WEB SCRAPING – ASSIGNMENT 2

In [1]:
```
!pip install selenium
```

```
Defaulting to user installation because normal site-packages is not writ
eable
Requirement already satisfied: selenium in c:\users\91968\appdata\roamin
g\python\python39\site-packages (4.8.2)
Requirement already satisfied: certifi>=2021.10.8 in c:\programdata\anac
onda3\lib\site-packages (from selenium) (2021.10.8)
Requirement already satisfied: trio-websocket~=0.9 in c:\users\91968\app
data\roaming\python\python39\site-packages (from selenium) (0.9.2)
Requirement already satisfied: trio~=0.17 in c:\users\91968\appdata\roam
ing\python\python39\site-packages (from selenium) (0.22.0)
Requirement already satisfied: urllib3[socks]~=1.26 in c:\programdata\an
aconda3\lib\site-packages (from selenium) (1.26.9)
Requirement already satisfied: idna in c:\programdata\anaconda3\lib\site
-packages (from trio~=0.17->selenium) (3.3)
Requirement already satisfied: sortedcontainers in c:\programdata\anacon
da3\lib\site-packages (from trio~=0.17->selenium) (2.4.0)
Requirement already satisfied: outcome in c:\users\91968\appdata\roaming
\python\python39\site-packages (from trio~=0.17->selenium) (1.2.0)
Requirement already satisfied: cffi>=1.14 in c:\programdata\anaconda3\li
b\site-packages (from trio~=0.17->selenium) (1.15.0)
Requirement already satisfied: exceptiongroup>=1.0.0rc9 in c:\users\9196
8\appdata\roaming\python\python39\site-packages (from trio~=0.17->seleni
um) (1.1.0)
Requirement already satisfied: attrs>=19.2.0 in c:\programdata\anaconda3
\lib\site-packages (from trio~=0.17->selenium) (21.4.0)
Requirement already satisfied: sniffio in c:\programdata\anaconda3\lib\s
ite-packages (from trio~=0.17->selenium) (1.2.0)
Requirement already satisfied: async-generator>=1.9 in c:\users\91968\ap
pdata\roaming\python\python39\site-packages (from trio~=0.17->selenium)
(1.10)
Requirement already satisfied: pycparser in c:\programdata\anaconda3\lib
\site-packages (from cffi>=1.14->trio~=0.17->selenium) (2.21)
Requirement already satisfied: wsproto>=0.14 in c:\users\91968\appdata\r
oaming\python\python39\site-packages (from trio-websocket~=0.9->seleniu
m) (1.2.0)
Requirement already satisfied: PySocks!=1.5.7,<2.0,>=1.5.6 in c:\program
data\anaconda3\lib\site-packages (from urllib3[socks]~=1.26->selenium)
(1.7.1)
Requirement already satisfied: h11<1,>=0.9.0 in c:\users\91968\appdata\r
oaming\python\python39\site-packages (from wsproto>=0.14->trio-websocket
~=0.9->selenium) (0.14.0)
```

In [ ]:
```
# QUESTION NO.1
```

In [ ]:
```
import selenium
import pandas as pd
from selenium import webdriver
import warnings
warnings.filterwarnings('ignore')
from selenium.common.exceptions import StaleElementRefrenceException, No
from selenium.webdriver.common.by import By
import time
```

In [ ]:
```
https://chromedriver.chromium.org/downloads
```

In [8]:
```
driver=webdriver.Chrome(r"chromedriver.exe")
```

In [9]:
```
driver.get("https://www.naukri.com/")
```

In [ ]:
```
designation=driver.find_element(By.CLASS_NAME,"suggestor-input ")
designation.send_keys('Data Analyst')
```

In [ ]:
```
location=driver.find_element(By.XPATH,"/html/body/div[1]/div[6]/div/div/
location.send_keys('Bangalore')
```

In [ ]:
```
search=driver.find_element(By.CLASS_NAME,"qsbSubmit")
search.click()
```

In [23]:
```
job_title=[]
job_location=[]
company_name=[]
experience_required=[]
```

In [ ]:
```
#Scraping job title
title_tags=driver.find_elements(By.XPATH,'//a[@class="title ellipsis"] '
for i in title_tags[0,10]:
    title=i.text
    job_title.append(title)

#Scraping job location
location_tags=driver.find_element(By.XPATH,'//span[@class="ellipsis flef
for i in location_tags[0:10]:
    location=i.text
    job_location.append(location)

#Scraping company name
company_tags=driver.find_elements(By.XPATH,'//a[@class="subTitle ellipsi
for i in company_tags[0:10]:
    company=i.text
    company_name.append(company)

#Scraping job experience
experience_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis f
for i in experience_tags[0:10]:
    exp=i.text
    experience_required.append(exp)
```

In [ ]:
```
print(len(job_title),len(job_location),len(company_name),len(experience_
```

In [ ]:
```
import pandas as pd
df=pd.DataFrame({'Title':job_title,'Location':job_location,'Company_name
df
```

In [ ]:
```
# QUESTION NO.2
```

In [ ]:
```
driver.get("https://www.naukri.com/")
```

In [ ]:
```
designation=driver.find_element(By.CLASS_NAME,"suggestor-input ")
designation.send_keys(' Data Scientist ')
```

In [ ]:
```
location=driver.find_element(By.XPATH,"/html/body/div[1]/div[6]/div/div/
location.send_keys('Bangalore')
```

In [ ]:
```
search=driver.find_element(By.CLASS_NAME,"qsbSubmit")
search.click()
```

In [19]:
```
job_title=[]
job_location=[]
company_name=[]
experience_required=[]
```

In [ ]:
```
#Scraping job title
title_tags=driver.find_elements(By.XPATH,'//a[@class="title ellipsis"] '
for i in title_tags[0,10]:
    title=i.text
    job_title.append(title)

#Scraping job location
location_tags=driver.find_element(By.XPATH,'//span[@class="ellipsis flef
for i in location_tags[0:10]:
    location=i.text
    job_location.append(location)

#Scraping company name
company_tags=driver.find_elements(By.XPATH,'//a[@class="subTitle ellipsi
for i in company_tags[0:10]:
    company=i.text
    company_name.append(company)

#Scraping job experience
experience_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis f
for i in experience_tags[0:10]:
    exp=i.text
    experience_required.append(exp)
```

In [ ]:
```
print(len(job_title),len(job_location),len(company_name),len(experience_
```

In [20]:
```
import pandas as pd
df=pd.DataFrame({'Title':job_title,'Location':job_location,'Company_name
df
```

Out[20]:

Title	Location	Company_name	Experience

In [ ]:
```
# QUESTION NO.3
```

In [ ]:
```
driver.get("https://www.naukri.com/")
```

In [ ]:
```
designation=driver.find_element(By.CLASS_NAME,"suggestor-input ")
designation.send_keys(' Data Scientist ')
```

In [ ]:
```
location=driver.find_element(By.XPATH,"/html/body/div[1]/div[6]/div/div/
location.send_keys('Bangalore')
```

In [ ]:
```
search=driver.find_element(By.CLASS_NAME,"qsbSubmit")
search.click()
```

In [ ]:
```
job_title=[]
job_location=[]
company_name=[]
experience_required=[]
```

In [ ]:
```
#Scraping job title
title_tags=driver.find_elements(By.XPATH,'//a[@class="title ellipsis"] '
for i in title_tags[0,10]:
    title=i.text
    job_title.append(title)

#Scraping job location
location_tags=driver.find_element(By.XPATH,'//span[@class="ellipsis flef
for i in location_tags[0:10]:
    location=i.text
    job_location.append(location)

#Scraping company name
company_tags=driver.find_elements(By.XPATH,'//a[@class="subTitle ellipsi
for i in company_tags[0:10]:
    company=i.text
    company_name.append(company)

#Scraping job experience
experience_tags=driver.find_elements(By.XPATH,'//span[@class="ellipsis f
for i in experience_tags[0:10]:
    exp=i.text
    experience_required.append(exp)
```

In [ ]:
```
print(len(job_title),len(job_location),len(company_name),len(experience_
```

In [ ]:
```
import pandas as pd
df=pd.DataFrame({'Title':job_title,'Location':job_location,'Company_name
df
```

In [ ]:
```
# QUESTION NO.4
```

In [ ]:
~~~~
# Scraping data from Flipkart

### Step 1: Importing necessary libraries

We will be using the BeautifulSoup and requests libraries in order to ex

```python
import requests
from bs4 import BeautifulSoup
```

### Step 2: Making a request to the website

We will make a `GET` request to the website using the `requests` library

```python
url = "https://www.flipkart.com/search?q=sunglasses"

response = requests.get(url)
```

### Step 3: Parsing the response

We will use the `BeautifulSoup` library to parse the HTML response from

```python
soup = BeautifulSoup(response.content, 'html.parser')
```

### Step 4: Extracting the required data

We will extract the required data from the HTML response using the `find

```python
items = soup.find_all('div', {'class': '_3O0U0u'})

data = []

for item in items:
    product_name = item.find('div', {'class': '_3wU53n'}).text
    price = item.find('div', {'class': '_1vC4OE'}).text
    data.append({
        'product_name': product_name,
        'price': price
    })

print(data)
```

### Step 5: Iterate over the next pages

We will use the `find()` method to find the `next` button from the HTML

```python
next_button = soup.find('a', {'class': '_3fVaIS'})
```

We will use a `while` loop to iterate over the next pages and extract th

```python
while next_button:
    url = "https://www.flipkart.com" + next_button['href']
    response = requests.get(url)
    soup = BeautifulSoup(response.content, 'html.parser')
    items = soup.find_all('div', {'class': '_3O0U0u'})

    for item in items:
        product_name = item.find('div', {'class': '_3wU53n'}).text
        price = item.find('div', {'class': '_1vC4OE'}).text
        data.append({
            'product_name': product_name,
            'price': price
        })

    next_button = soup.find('a', {'class': '_3fVaIS'})

print(data)
```
~~~~

In [ ]:
```
# QUESTION NO.5
```

In [ ]:
```
import time
import pandas as pd
from selenium import webdriver

url = 'https://www.flipkart.com/apple-iphone-11-black-64-gb/product revi

driver = webdriver.Chrome('chromedriver.exe')
driver.get(url)

reviews = []

for i in range(5):
    reviews = driver.find_elements_by_xpath("//div[@class='_3gijNv col-1

    for review in reviews:
        reviews.append(review.text)

    button = driver.find_element_by_xpath("//a[@class='_3fVaIS']//span[@

    button.click()
    time.sleep(3)

data = pd.DataFrame(data=reviews, columns=['Reviews'])
data.to_csv('iphone11_reviews.csv', index=False)

driver.close()
```

In [ ]:
```
# QUESTION NO.6
```

In [ ]:
```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Initialize the webdriver
driver = webdriver.Chrome()

# Go to the amazon.in website
driver.get("https://www.flipkart.com/")

# Enter sneakers in the search field
search_field = driver.find_element_by_id("twotabsearchtextbox")
search_field.send_keys("sneakers")

# Click the search icon
search_button = driver.find_element_by_xpath("//input[@value='Go']")
search_button.click()

# Set the CPU Type filter to sneakers
wait = WebDriverWait(driver, 100)
element = wait.until(EC.element_to_be_clickable((By.XPATH, "//span[text(
element.click()

# Scrape the brand, product description, and price
sneakers = driver.find_elements_by_xpath("//div[@class='_3704LK']//div[@

for sneakers in sneakers:
    brand = sneakers.find_element_by_xpath(".//span[contains(@class, 'a-
    product = sneakers.find_element_by_xpath(".//span[contains(@class, '
    price = sneakers.find_element_by_xpath(".//span[contains(@class, 'a-

    print(brand)
    print(product)
    print(price)
    print()
```

In [34]:
```
# QUESTION NO.7
```

In [ ]:
```
from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.support import expected_conditions as EC

# Initialize the webdriver
driver = webdriver.Chrome()

# Go to the amazon.in website
driver.get("https://www.amazon.in/")

# Enter laptop in the search field
search_field = driver.find_element_by_id("twotabsearchtextbox")
search_field.send_keys("Laptop")

# Click the search icon
search_button = driver.find_element_by_xpath("//input[@value='Go']")
search_button.click()

# Set the CPU Type filter to Intel Core i7
wait = WebDriverWait(driver, 10)
element = wait.until(EC.element_to_be_clickable((By.XPATH, "//span[text(
element.click()

# Scrape the title, ratings, and price of each laptop
laptops = driver.find_elements_by_xpath("//div[@class='sg-row']//div[@da

for laptop in laptops:
    title = laptop.find_element_by_xpath(".//span[contains(@class, 'a-te
    ratings = laptop.find_element_by_xpath(".//span[contains(@class, 'a-
    price = laptop.find_element_by_xpath(".//span[contains(@class, 'a-pr

    print(title)
    print(ratings)
    print(price)
    print()
```

In [35]:
```
# QUESTION NO.8
```

In [ ]:
```
#import library
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
import csv

#open browser
driver = webdriver.Chrome()

#open the website
driver.get("https://www.azquotes.com/top_quotes")

#scrap
quotes = []
authors = []
types = []

#loop through each element
for quote in driver.find_elements_by_class_name('quoteText'):
    quotes.append(quote.text)

for author in driver.find_elements_by_class_name('authorOrTitle'):
    authors.append(author.text)

for type in driver.find_elements_by_class_name('type'):
    types.append(type.text)

#store the scraped data in csv file
with open('quotes.csv', 'w', newline='') as csvfile:
    writer = csv.writer(csvfile)
    writer.writerow(['Quotes', 'Authors', 'Types'])
    for i in range(len(quotes)):
        writer.writerow([quotes[i], authors[i], types[i]])

#print top 1000 quotes of all time
print("Top 1000 Quotes of All Time: ")
for i in range(len(quotes)):
    print(quotes[i], '-', authors[i], '(' + types[i] + ')')

#close the browser
driver.close()
```

In [ ]:
```
# QUESTION NO.9
```

In [ ]:
```
#importing libraries
from selenium import webdriver
import pandas as pd

#getting the webpage
driver = webdriver.Chrome()
driver.get("https://www.jagranjosh.com/")

#clicking on the GK option
gk_option = driver.find_element_by_xpath('/html/body/div[1]/div[3]/div[1
gk_option.click()

#clicking on the List of all Prime Ministers of India
list_of_all_prime_ministers = driver.find_element_by_link_text('List of
list_of_all_prime_ministers.click()

#scrapping the data
name_list = driver.find_elements_by_xpath('/html/body/div[2]/div[3]/div[
born_dead_list = driver.find_elements_by_xpath('/html/body/div[2]/div[3]
term_of_office_list = driver.find_elements_by_xpath('/html/body/div[2]/d
remarks_list = driver.find_elements_by_xpath('/html/body/div[2]/div[3]/d

#storing the scrapped data
data_list = []
for i in range(len(name_list)):
    data_list.append({'Name':name_list[i].text, 'Born-Dead':born_dead_li

#creating dataframe
data_frame = pd.DataFrame(data_list)

#displaying the dataframe
print(data_frame)
```

In [ ]:
```
# QUESTION NO.10
```

In [ ]:
```
import pandas as pd
from selenium import webdriver

# create a new Chrome session
driver = webdriver.Chrome()
driver.implicitly_wait(30)
driver.get("https://www.motor1.com/")

#click on the list option
list_btn = driver.find_element_by_xpath('//*[@id="primary-navigation"]/u
list_btn.click()

#click on 50 most expensive cars in the world
expensive_btn = driver.find_element_by_xpath('//*[@id="main-content"]/di
expensive_btn.click()

#scraping data
table_data = driver.find_elements_by_xpath('//*[@id="main-content"]/div[

#creating dataframe
cars_df = pd.DataFrame(columns=['Car_Name','Price'])

for row in table_data:
    data = row.find_elements_by_tag_name('td')
    car_name = data[0].text
    price = data[1].text
    cars_df = cars_df.append({'Car_Name':car_name,'Price':price},ignore_

driver.quit()

# print the dataframe
print(cars_df)
```

In [ ]: