WEB SCRAPING – ASSIGNMENT 4

1. Scrape the details of most viewed videos on YouTube from Wikipedia. Url = https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos You need to find following details: A) Rank B) Name C) Artist D) Upload date E) Views

In [ ]:
```
from selenium import webdriver
from selenium.common.exceptions import NoSuchElementException

driver = webdriver.Chrome('C:/path/to/chromedriver.exe')
driver.get('https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos')

rank = []
name = []
artist = []
upload_date = []
views = []

table = driver.find_element_by_id('mw-content-text').find_element_by_tag_name('table')

for row in table.find_elements_by_tag_name('tr'):
    cells = row.find_elements_by_tag_name('td')
    if len(cells) == 5:
        try:
            rank.append(cells[0].text)
            name.append(cells[1].text)
            artist.append(cells[2].text)
            upload_date.append(cells[3].text)
            views.append(cells[4].text)
        except NoSuchElementException as e:
            print(e)

driver.quit()
```

2. Scrape the details teamindia/sinternationalfixtures from bcci.tv. Url = https://www.bcci.tv/. You need to find following details: A) Match title (i.e. 1stODI) B) Series C) Place D) Date E) Time Note: - From bcci.tv home page you have reach to the international fixture page through code

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

# open the web browser and navigate to the bcci.tv home page
driver = webdriver.Chrome()
driver.get('https://www.bcci.tv/')

# click on the 'International Fixtures' link
wait = WebDriverWait(driver, 10)
element = wait.until(EC.element_to_be_clickable((By.LINK_TEXT, 'International Fixtures')))
element.click()

# extract the table containing the international fixtures
table = driver.find_element_by_xpath('//*[@id="root"]/div/div/div[2]/div/div[2]/div[2]/div/div[3]/div[2]/div/div[2]/div/div[2]/table')

# loop through the table rows and extract the data
for row in table.find_elements_by_tag_name('tr'):
    cells = row.find_elements_by_tag_name('td')
    if len(cells) > 0:
        match_title = cells[0].text
        series = cells[1].text
        place = cells[2].text
        date = cells[3].text
        time = cells[4].text

        print(f"Match Title: {match_title}, Series: {series}, Place: {place}, Date: {date}, Time: {time}")

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

3. Scrape the details of State-wise GDP ofindia fromstatisticstime.com. Url = http://statisticstimes.com/ You have to find following details: A) Rank B) State C) GSDP(18-19)- at current prices D) GSDP(18-20)- at current prices E) Share(18-19) F) GDP($ billion) Note: - From statisticstimes home page you have to reach to economy page through code

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

In [ ]:
```
#Initializing the webdriver
driver = webdriver.Chrome()
driver.maximize_window()

#opening the website
driver.get("http://statisticstimes.com/")

#waiting for the page to load
WebDriverWait(driver,10).until(EC.presence_of_element_located((By.XPATH,"//span[contains(text(),'Economy')]")))

#clicking on the Economy tab
driver.find_element_by_xpath("//span[contains(text(),'Economy')]").click()

#waiting for the page to load
WebDriverWait(driver,10).until(EC.presence_of_element_located((By.XPATH,"//span[contains(text(),'State Wise GDP of India')]")))

#clicking on State Wise GDP of India
driver.find_element_by_xpath("//span[contains(text(),'State Wise GDP of India')]").click()

#waiting for the page to load
WebDriverWait(driver,10).until(EC.presence_of_element_located((By.XPATH,"//tr[@class='tablecont']")))

#scraping the data
data = driver.find_elements_by_xpath("//tr[@class='tablecont']")

#creating a list to store the data
state_wise_gdp_data = []

#looping through the data to get the required data
for d in data:
        information = d.text.split('\n')
        rank = information[0]
        state = information[1]
        gsdp_18_19 = information[2]
        gsdp_19_20 = information[3]
        share_18_19 = information[4]
        gdp_billion = information[5]
        state_wise_gdp_data.append([rank,state,gsdp_18_19,gsdp_19_20,share_18_19,gdp_billion])

#printing the scraped data
print(state_wise_gdp_data)
```

4.Scrape the details of trending repositories on Github.com. Url = https://github.com/ You have to find the following details: A) Repository title B) Repository description C) Contributors count D) Language used Note: - From the home page you have to click on the trending option from Explore menu through code

In [ ]:
```
from selenium import webdriver

driver = webdriver.Chrome()
driver.get('https://github.com/')

# Click on the "Trending" option in the Explore menu
trending_link = driver.find_element_by_css_selector('a[href$="trending"]')
trending_link.click()

# Get all the repository cards
repo_cards = driver.find_elements_by_class_name('repo-list-item')

for card in repo_cards:
    # Extract the details from the card
    title = card.find_element_by_class_name('h3').text
    description = card.find_element_by_class_name('py-1').text
    contributors = card.find_element_by_class_name('f6').text
    language = card.find_element_by_class_name('mr-3').text

    # Print the details
    print(f"Title: {title}")
    print(f"Description: {description}")
    print(f"Contributors: {contributors}")
    print(f"Language: {language}")
    print()

driver.quit()
```

1. Scrape the details of top 100 songs on billboard.com. Url = https:www.billboard.com/ You have to find the following details: A) Song name B) Artist name C) Last week rank D) Peak rank E) Weeks on board Note: - From the home page you have to click on the charts option then hot 100-page link through code

In [ ]:
```
from selenium import webdriver

#Create an instance of the Firefox driver
driver = webdriver.Firefox()

#Navigate to the Billboard website
driver.get("https://www.billboard.com/")

#Click on the Charts option
chart_link = driver.find_element_by_xpath("//a[@title='Charts Home']")
chart_link.click()

#Click the Hot 100 link
hot_100_link = driver.find_element_by_xpath("//a[@data-track-label='Hot 100']")
hot_100_link.click()

#Wait for the page to load
time.sleep(3)

#Scrape the details of top 100 songs
for i in range(1,101):
    #Find the element for the song name
    song_name_xpath = "//tr[@class='chart-list-item'][" + str(i) + "]/td[@class='chart-list-item__title']/div/span"
    song_name_element = driver.find_element_by_xpath(song_name_xpath)
    #Get the song name
    song_name = song_name_element.text

    #Find the element for the artist name
    artist_name_xpath = "//tr[@class='chart-list-item'][" + str(i) + "]/td[@class='chart-list-item__artist']/div/span"
    artist_name_element = driver.find_element_by_xpath(artist_name_xpath)
    #Get the artist name
    artist_name = artist_name_element.text

    #Find the element for the last week rank
    last_week_xpath = "//tr[@class='chart-list-item'][" + str(i) + "]/td[@class='chart-list-item__last-week']/div/span"
    last_week_element = driver.find_element_by_xpath(last_week_xpath)
    #Get the last week rank
    last_week = last_week_element.text

    #Find the element for the peak rank
    peak_position_xpath = "//tr[@class='chart-list-item'][" + str(i) + "]/td[@class='chart-list-item__peak']/div/span"
    peak_position_element = driver.find_element_by_xpath(peak_position_xpath)
    #Get the peak position
    peak_position = peak_position_element.text

    #Find the element for the weeks on board
    weeks_on_board_xpath = "//tr[@class='chart-list-item'][" + str(i) + "]/td[@class='chart-list-item__weeks-on-chart']/div/span"
    weeks_on_board_element = driver.find_element_by_xpath(weeks_on_board_xpath)
    #Get the weeks on board
    weeks_on_board = weeks_on_board_element.text

    #Print the details
    print("Song:", song_name)
    print("Artist:", artist_name)
    print("Last Week Rank:", last_week)
    print("Peak Rank:", peak_position)
    print("Weeks on Board:", weeks_on_board)
    print("-----------------------------")

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

1. Scrape the details of Highest sellingnovels. Url = https://www.theguardian.com/news/datablog/2012/aug/09/best-selling-books-all-time-fifty-shades-greycompare You have to find the following details: A) Book name B) Author name C) Volumes sold D) Publisher E) Genre

In [ ]:
```
from selenium import webdriver
from selenium.common.exceptions import NoSuchElementException

browser = webdriver.Chrome()
url = "https://www.theguardian.com/news/datablog/2012/aug/09/best-selling-books-all-time-fifty-shades-greycompare"

browser.get(url)

book_name = []
author_name = []
volumes_sold = []
publisher = []
genre = []

# Find the table containing the data
table = browser.find_element_by_css_selector('div#block-system-main > table.datatable.sortable.sticky-enabled.tablefooter-processed.sticky-table')

# Get the rows of the table
rows = table.find_elements_by_css_selector('tbody tr')

for row in rows:
    # Get all the columns in the row
    columns = row.find_elements_by_css_selector('td')

    # Extract the data from each column
    book_name.append(columns[0].text)
    author_name.append(columns[1].text)
    volumes_sold.append(columns[2].text)
    publisher.append(columns[3].text)
    genre.append(columns[4].text)

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

1. Scrape the details most watched tv series of all time from imdb.com. Url = https://www.imdb.com/list/ls095964455/ You have to find the following details: A) Name B) Year span C) Genre D) Run time E) Ratings F) Votes

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

# Create a new instance of the Chrome driver
driver = webdriver.Chrome()

# Go to the IMDb website
driver.get("https://www.imdb.com/list/ls095964455/")

# Wait for the list to load
WebDriverWait(driver, 10).until(EC.presence_of_element_located((By.XPATH, '//*[@id="main"]/div/span/div/div/div[3]/table/tbody')))

# Get the list of series
series_list = driver.find_elements_by_xpath('//*[@id="main"]/div/span/div/div/div[3]/table/tbody/tr')

# Iterate over the list of series to get the details
for series in series_list:

    # Get the name
    name = series.find_element_by_xpath('.//td[2]/a').text

    # Get the year span
    year_span = series.find_element_by_xpath('.//td[2]/span').text

    # Get the genre
    genre = series.find_element_by_xpath('.//td[3]').text

    # Get the run time
    run_time = series.find_element_by_xpath('.//td[4]').text

    # Get the ratings
    ratings = series.find_element_by_xpath('.//td[5]').text

    # Get the votes
    votes = series.find_element_by_xpath('.//td[6]').text

    print('Name:', name)
    print('Year span:', year_span)
    print('Genre:', genre)
    print('Run time:', run_time)
    print('Ratings:', ratings)
    print('Votes:', votes)
    print()
```

1. Details of Datasetsfrom UCI machine learning repositories. Url = https://archive.ics.uci.edu/ You have to find the following details: A) Dataset name B) Data type C) Task D) Attribute type E) No of instances F) No of attribute G) Year Note: - from the home page you have to go to the ShowAllDataset page through code

In [ ]:
```
#Importing the necessary libraries
import requests
import pandas as pd
from bs4 import BeautifulSoup

#Getting the request from the URL
page = requests.get("https://archive.ics.uci.edu/ml/datasets.php")

#Creating a soup object for parsing the HTML
soup = BeautifulSoup(page.content, 'html.parser')

#Collecting the data table
table = soup.find('table')

#Creating a list of all the rows in the data table
table_rows = table.find_all('tr')

#Creating empty lists to store the data
dataset_name = []
data_type = []
task = []
attribute_type = []
no_of_instances = []
no_of_attribute = []
year = []

#Looping through each row and collecting the data
for tr in table_rows:
    td = tr.find_all('td')
    dataset_name.append(td[1].text.strip())
    data_type.append(td[2].text.strip())
    task.append(td[3].text.strip())
    attribute_type.append(td[4].text.strip())
    no_of_instances.append(td[5].text.strip())
    no_of_attribute.append(td[6].text.strip())
    year.append(td[7].text.strip())

#Creating the dataframe
data = pd.DataFrame(list(zip(dataset_name, data_type, task, attribute_type, no_of_instances, no_of_attribute, year)),
                    columns =['Dataset Name', 'Data Type', 'Task', 'Attribute Type', 'No of Instances', 'No of Attribute', 'Year'])

#Printing the dataFrame
data
```

1. Scrape the details of Data science recruiters Url = https://www.naukri.com/hr-recruiters-consultants You have to find the following details: A) Name B) Designation C)Company D)Skills they hire for E) Location Note: - From naukri.com homepage click on the recruiters option and the on the search pane type Data science and click on search. All this should be done through code

In [ ]:
```
fro 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
from selenium.common.exceptions import TimeoutException
```

In [ ]:
```
#Initializing the driver
driver = webdriver.Chrome(executable_path="path/to/chromedriver")

#opening the url
url = "https://www.naukri.com/hr-recruiters-consultants"

#opening the page
driver.get(url)

#waiting for the page to load
timeout = 10
try:
    WebDriverWait(driver, timeout).until(EC.visibility_of_element_located((By.XPATH, "//input[@placeholder='Search recruiters']")))
except TimeoutException:
    print("Timed out waiting for page to load")
    driver.quit()

#finding the search box
search_box = driver.find_element_by_xpath("//input[@placeholder='Search recruiters']")

#specifying the search term
search_term = "Data science"

#placing the search term in the search box
search_box.send_keys(search_term)

#clicking the search button
search_button = driver.find_element_by_xpath("//button[@class='rw-btn rw-btn--primary sbc"]")
search_button.click()

#finding the list of recruiters
recruiter_list = driver.find_elements_by_xpath("//div[@class='rw-grid rw-grid--equal-width rw-grid--no-gutter rw-grid--wrap rw-mb--medium"]")

#looping through the list of recruiters
for recruiter in recruiter_list:
    #finding the name
    name = recruiter.find_element_by_xpath('.//div[@class="rw-grid__cell rw-mr--small"]/a').text
    #finding the designation
    designation = recruiter.find_element_by_xpath(".//div[@class='rw-grid__cell rw-mr--small"]/p[1]").text
    #finding the company
    company = recruiter.find_element_by_xpath(".//div[@class="rw-grid__cell rw-mr--small"]/p[2]").text
    #finding the skills they hire for
    skills = recruiter.find_element_by_xpath(".//div[@class="rw-grid__cell rw-mr--small"]/p[3]").text
    #finding the location
    location = recruiter.find_element_by_xpath(".//div[@class="rw-grid__cell rw-mr--small"]/p[4]").text

    #printing the details
    print("Name:", name)
    print("Designation:", designation)
    print("Company:", company)
    print("Skills they hire for:", skills)
    print("Location:", location)
    print("-----------------------------------------")

#closing the driver
driver.quit()
```

In [ ]:
```
```