Web Scraping Assignment 4

In [1]:

```
# importing libraries of selenium
import selenium
from selenium import webdriver
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.common.by import By
from selenium.common.exceptions import NoSuchElementException, StaleElementReference
from selenium.webdriver.support import expected_conditions as EC

import pandas as pd
from time import sleep
import re
import warnings
warnings.filterwarnings("ignore")
```

Q1.

Scrape the details of most viewed videos on YouTube from Wikipedia. Url = https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos (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 [2]:

```
#loading the driver and getting the url
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()
url='https://en.wikipedia.org/wiki/List_of_most-viewed_YouTube_videos'
driver.get(url)
sleep(10)

#getting each data row wise from the table like structure data.
ranks=driver.find_elements(By.XPATH,'//table[@class="wikitable sortable jquery-table
names=driver.find_elements(By.XPATH,'//table[@class="wikitable sortable jquery-table
artists=driver.find_elements(By.XPATH,'//table[@class="wikitable sortable jquery-tab
views=driver.find_elements(By.XPATH,'//table[@class="wikitable sortable jquery-table
upload_date=driver.find_elements(By.XPATH,'//table[@class="wikitable sortable jquery

#creating empty list
most_views_youtube=[]
#creating empty dictionary for temporary usage
table={}

# iterate each lists to get into dataframe
for i in range(len(ranks)):
    table={"RANK":ranks[i].text,
           " NAME":names[i].text,
           "ARTISTS":artists[i].text,
           "VIEWS (in millions)": views[i].text,
           "UPLOAD DATE" : upload_date[i].text}
    most_views_youtube.append(table)

#quit the driver
driver.quit()

# making the dataframe
df_data=pd.DataFrame(most_views_youtube)
df_data
```

Out[2]:

	RANK	NAME	ARTISTS	VIEWS (in millions)	UPLOAD DATE
0	1.	"Baby Shark Dance"[6]	Pinkfong Baby Shark - Kids' Songs & Stories	12.85	June 17, 2016
1	2.	"Despacito"[9]	Luis Fonsi	8.16	January 12, 2017
2	3.	"Johny Johny Yes Papa"[16]	LooLoo Kids	6.70	October 8, 2016
3	4.	"Bath Song"[17]	Cocomelon – Nursery Rhymes	6.20	May 2, 2018
4	5.	"Shape of You"[18]	Ed Sheeran	6.00	January 30, 2017
5	6.	"See You Again"[21]	Wiz Khalifa	5.89	April 6, 2015
6	7.	"Phonics Song with Two Words"[26]	ChuChu TV	5.30	March 6, 2014
7	8.	"Wheels on the Bus"[27]	Cocomelon – Nursery Rhymes	5.24	May 24, 2018
8	9.	"Uptown Funk"[28]	Mark Ronson	4.92	November 19, 2014
9	10.	"Learning Colors – Colorful Eggs on a Farm"[29]	Miroshka TV	4.89	February 27, 2018
10	11.	"Gangnam Style"[30]	Psy	4.80	July 15, 2012
11	12.	"Masha and the Bear – Recipe for Disaster"[35]	Get Movies	4.55	January 31, 2012
12	13.	"Dame Tu Cosita"[36]	El Chombo	4.35	April 5, 2018
13	14.	"Axel F"[37]	Crazy Frog	3.91	June 16, 2009
14	15.	"Sugar"[38]	Maroon 5	3.87	January 14, 2015
15	16.	"Roar"[39]	Katy Perry	3.80	September 5, 2013
16	17.	"Counting Stars"[40]	OneRepublic	3.79	May 31, 2013
17	18.	"Sorry"[41]	Justin Bieber	3.66	October 22, 2015
18	19.	"Baa Baa Black Sheep"[42]	Cocomelon – Nursery Rhymes	3.64	June 25, 2018
19	20.	"Thinking Out Loud"[43]	Ed Sheeran	3.60	October 7, 2014
20	21.	"Waka Waka (This Time for Africa)"[44]	Shakira	3.59	June 4, 2010
21	22.	"Dark Horse"[45]	Katy Perry	3.52	February 20, 2014
22	23.	"Lakdi Ki Kathi"[46]	Jingle Toons	3.48	June 14, 2018
23	24.	"Faded"[47]	Alan Walker	3.45	December 3, 2015
24	25.	"Perfect"[48]	Ed Sheeran	3.45	November 9, 2017
25	26.	"Let Her Go"[49]	Passenger	3.44	July 25, 2012

	RANK	NAME	ARTISTS	VIEWS (in millions)	UPLOAD DATE
26	27.	"Girls Like You"[50]	Maroon 5	3.42	May 31, 2018
27	28.	"Humpty the train on a fruits ride"[51]	Kiddiestv Hindi – Nursery Rhymes & Kids Songs	3.41	January 26, 2018
28	29.	"Lean On"[52]	Major Lazer	3.38	March 22, 2015
29	30.	"Bailando"[53]	Enrique Iglesias	3.38	April 11, 2014

Q2.

Scrape the details team India's international fixtures from bcci.tv. Url = https://www.bcci.tv/ (https://www.bcci.tv/). You need to find following details: A) Match title (I.e. 1st ODI) 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 [3]:

```
#loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()

#loading the url into the driver
url='https://www.bcci.tv/'
driver.get(url)
# webdriverwait to wait the driver explicitely
WebDriverWait(driver,20).until(EC.presence_of_element_located((By.CLASS_NAME,"footer

#click on the international fixtures
international=driver.find_element(By.XPATH,'//li[@class="nav-item"]/a')
international.click()
sleep(10)

# getting each element in a listed form
Match_title=driver.find_elements(By.XPATH,'//span[@class="matchOrderText ng-binding
tour=driver.find_elements(By.XPATH,'//h5[@class="match-tournament-name ng-binding"]'
place=driver.find_elements(By.XPATH,'//div[@class="match-place ng-scope"]')
date=driver.find_elements(By.XPATH,'//div[@class="match-dates ng-binding"]')
time=driver.find_elements(By.XPATH,'//div[@class="match-time no-margin ng-binding"]'

# creating empty list
fixture=[]
# creating empty dictionary
temporary_dict={}

# iterate the scraped list to get into listed dictionary(key value pair)
for i in range(len(Match_title)):
    temporary_dict={"Match Title": Match_title[i].text,
                    "Series":tour[i].text,
                    "Place":place[i].text.split("-")[1],
                    "Date" : date[i].text,
                    "Time":time[i].text}
    fixture.append(temporary_dict)
#quit the driver
driver.quit()

# creating the listed dictionary into dataframe format
df_fixtures=pd.DataFrame(fixture)
df_fixtures
```

Out[3]:

	Match Title	Series	Place	Date	Time
0	1st T20I -	INDIA WOMEN TOUR OF BANGLADESH 2023	Shere Bangla National Stadium, Mirpur, Dhaka	9 JUL 2023	1:30 PM IST
1	2nd T20I -	INDIA WOMEN TOUR OF BANGLADESH 2023	Shere Bangla National Stadium, Mirpur, Dhaka	11 JUL 2023	1:30 PM IST
2	1st Test -	INDIA TOUR OF WEST INDIES 2023	Windsor Park, Dominica	12 JUL 2023	7:30 PM IST
3	3rd T20I -	INDIA WOMEN TOUR OF BANGLADESH 2023	Shere Bangla National Stadium, Mirpur, Dhaka	13 JUL 2023	1:30 PM IST
4	1st ODI -	INDIA WOMEN TOUR OF BANGLADESH 2023	Shere Bangla National Stadium, Mirpur, Dhaka	16 JUL 2023	9:00 AM IST
5	2nd ODI -	INDIA WOMEN TOUR OF BANGLADESH 2023	Shere Bangla National Stadium, Mirpur, Dhaka	19 JUL 2023	9:00 AM IST
6	2nd Test -	INDIA TOUR OF WEST INDIES 2023	Queen's Park Oval, Trinidad	20 JUL 2023	7:30 PM IST
7	3rd ODI -	INDIA WOMEN TOUR OF BANGLADESH 2023	Shere Bangla National Stadium, Mirpur, Dhaka	22 JUL 2023	9:00 AM IST

Q3.

Scrape the details of State-wise GDP of India from statisticstime.com. Url = http://statisticstimes.com/ (http://statisticstimes.com/) You have to find following details: A) Rank B) State C) GSDP(18-19)- at current prices D) GSDP(19-20)- 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 [4]:

```
#loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()

#loading the url
url='https://www.statisticstimes.com/'
driver.get(url)
sleep(5)

#got it button
try:
    got_it=driver.find_element(By.XPATH,'/html/body/div[1]/div/a')
    got_it.click()
except:
    print("exception handled fot got it")

#click on economy
economy=driver.find_elements(By.XPATH,'//div[@class="navbar"]/div/button')
economy[1].click()

# click on india
ind=driver.find_element(By.XPATH,'//div[@class="navbar"]/div[2]/div/a[3]')
ind.click()

```

In [5]:

```
#click on indian state
indian_states=driver.find_element(By.XPATH,'/html/body/div[2]/div[2]/div[2]/ul/li[1]
indian_states.click()
sleep(10)

# getting each data from table into listed form
rank=driver.find_elements(By.XPATH,'//table[@id="table_id"]/tbody/tr/td[1]')
state=driver.find_elements(By.XPATH,'//table[@id="table_id"]/tbody/tr/td[2]')
gsdp_18_19=driver.find_elements(By.XPATH,'//table[@id="table_id"]/tbody/tr/td[4]')
gsdp_19_20=driver.find_elements(By.XPATH,'//table[@id="table_id"]/tbody/tr/td[3]')
share_18_19=driver.find_elements(By.XPATH,'//table[@id="table_id"]/tbody/tr/td[5]')
gdp=driver.find_elements(By.XPATH,'//table[@id="table_id"]/tbody/tr/td[6]')

#create empty list
data_list=[]
#create temporary dictionary
temp_dict={}

#iterate the listed data to form listed dictionary
for i in range(len(rank)):
    temp_dict={"RANK":rank[i].text,
               "STATE":state[i].text,
               "GSDP (Cr INR at Current prices) 18-19" : gsdp_18_19[i].text,
               "GSDP (Cr INR at Current prices) 19-20" :gsdp_19_20[i].text,
               "SHARE 18-19" : share_18_19[i].text,
               "GDP ($billion) 2019" : gdp[i].text}
    data_list.append(temp_dict)
#quit the driver
driver.quit()

#creating dataframe
df_economy=pd.DataFrame(data_list)
df_economy
```

Out[5]:

	RANK	STATE	GSDP (Cr INR at Current prices) 18-19	GSDP (Cr INR at Current prices) 19-20	SHARE 18-19	GDP ($billion) 2019
0	1	Maharashtra	2,632,792	-	13.94%	399.921
1	2	Tamil Nadu	1,630,208	1,845,853	8.63%	247.629
2	3	Uttar Pradesh	1,584,764	1,687,818	8.39%	240.726
3	4	Gujarat	1,502,899	-	7.96%	228.290
4	5	Karnataka	1,493,127	1,631,977	7.91%	226.806
5	6	West Bengal	1,089,898	1,253,832	5.77%	165.556
6	7	Rajasthan	942,586	1,020,989	4.99%	143.179
7	8	Andhra Pradesh	862,957	972,782	4.57%	131.083
8	9	Telangana	861,031	969,604	4.56%	130.791
9	10	Madhya Pradesh	809,592	906,672	4.29%	122.977
10	11	Kerala	781,653	-	4.14%	118.733
11	12	Delhi	774,870	856,112	4.10%	117.703
12	13	Haryana	734,163	831,610	3.89%	111.519
13	14	Bihar	530,363	611,804	2.81%	80.562
14	15	Punjab	526,376	574,760	2.79%	79.957
15	16	Odisha	487,805	521,275	2.58%	74.098
16	17	Assam	315,881	-	1.67%	47.982
17	18	Chhattisgarh	304,063	329,180	1.61%	46.187
18	19	Jharkhand	297,204	328,598	1.57%	45.145
19	20	Uttarakhand	245,895	-	1.30%	37.351
20	21	Jammu & Kashmir	155,956	-	0.83%	23.690
21	22	Himachal Pradesh	153,845	165,472	0.81%	23.369
22	23	Goa	73,170	80,449	0.39%	11.115
23	24	Tripura	49,845	55,984	0.26%	7.571
24	25	Chandigarh	42,114	-	0.22%	6.397
25	26	Puducherry	34,433	38,253	0.18%	5.230
26	27	Meghalaya	33,481	36,572	0.18%	5.086
27	28	Sikkim	28,723	32,496	0.15%	4.363
28	29	Manipur	27,870	31,790	0.15%	4.233
29	30	Nagaland	27,283	-	0.14%	4.144
30	31	Arunachal Pradesh	24,603	-	0.13%	3.737
31	32	Mizoram	22,287	26,503	0.12%	3.385
32	33	Andaman & Nicobar Islands	-	-	-	-

Q4.

Scrape the details of trending repositories on Github.com. Url = https://github.com/ (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 [6]:

```
#loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()

#loading the url
url='https://github.com/'
driver.get(url)
sleep(5)

# Click on open source
open_source=driver.find_element(By.XPATH,'/html/body/div[1]/div[1]/header/div/div[2]
open_source.click()
sleep(5)

#click on trending
trending=driver.find_element(By.XPATH,'/html/body/div[1]/div[1]/header/div/div[2]/di
trending.click()
sleep(5)

#getting url of each tending data available on github
url=[]
urls=driver.find_elements(By.XPATH,'//article[@class="Box-row"]/h2/a')
for i in urls:
    url.append(i.get_attribute("href"))
print(len(url))


#creating empty list
title=[]
description=[]
contributors=[]
language=[]

# iterate each url
for i in range(len(url)):
    driver.get(url[i])
    sleep(5)

    #getting title
    try:
        title_=driver.find_element(By.XPATH,'//div[@class="flex-auto min-width-0 wid
        title.append(title_.text)
    except NoSuchElementException:
        title.append(None)

    #Getting description
    try:
        desc=driver.find_element(By.XPATH,'//div[@class="BorderGrid-cell"]/p')
        description.append(desc.text)
    except NoSuchElementException:
        description.append(None)

    try:
        contributor=driver.find_element(By.XPATH,'//span[@class="Counter ml-1"]')
        contributors.append(contributor.text)
    except NoSuchElementException:
        contributors.append(None)

    try:
```

```
        lang=driver.find_element(By.XPATH,'//span[@class="color-fg-default text-bold
        language.append(lang.text)
    except NoSuchElementException:
        language.append(None)

#quit the driver
driver.quit()

df_github=pd.DataFrame({"Repository Title":title,
                "Repository Description":description,
                "Contributor Count":contributors,
                "Language":language})
df_github
```

25

Out[6]:

	Repository Title	Repository Description	Contributor Count	Language
0	DragGAN	Official Code for DragGAN (SIGGRAPH 2023)		Python
1	ChatGLM2-6B	ChatGLM2-6B: An Open Bilingual Chat LLM | 开源双语...		Python
2	FastSAM	Fast Segment Anything		Python
3	freegpt-webui	GPT 3.5/4 with a Chat Web UI. No API key requi...	3	Python
4	embedchain	Framework to easily create LLM powered bots ov...		Python
5	spacedrive	Spacedrive is an open source cross-platform fi...	64	Rust
6	open-resume	OpenResume is a powerful open-source resume bu...		TypeScript
7	papers-we-love	Papers from the computer science community to ...		Shell
8	skateshop	An open source e-commerce skateshop build with...		TypeScript
9	Web-Dev-For-Beginners	24 Lessons, 12 Weeks, Get Started as a Web Dev...		JavaScript
10	diy-spacemouse	A DIY navigation device for Fusion360		C++
11	ChatGLM-6B	ChatGLM-6B: An Open Bilingual Dialogue Languag...		Python
12	PanoHead	Code Repository for CVPR 2023 Paper "PanoHead:...		Python
13	awesome-chatgpt-prompts-zh	ChatGPT 中文调教指南。各种场景使用指南。学习怎么让它听你的话。		None
14	first-contributions	🚀✨ Help beginners to contribute to open source...	5,000+	None
15	actual	A local-first personal finance system		JavaScript
16	gpt4free	The official gpt4free repository | various col...	83	Python
17	svelte	Cybernetically enhanced web apps		JavaScript
18	DragGAN	Unofficial Implementation of DragGAN - "Drag Y...	9	Python
19	UniAD	[CVPR 2023 Best Paper] Planning-oriented Auton...	6	Python
20	QGIS	QGIS is a free, open source, cross platform (l...	20	C++
21	Chat2DB	🔥 🔥 🔥 An intelligent and versatile general-pur...		Java
22	pygwalker	PyGWalker: Turn your pandas dataframe into a T...	11	Python
23	ggml	Tensor library for machine learning		C
24	quivr	🧠 Dump all your files and thoughts into your p...		TypeScript

Q5.

Scrape the details of top 100 songs on billiboard.com. Url = https:/www.billboard.com/ (http://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 [7]:

```
#loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()

#getting url and load into driver
url='https:/www.billboard.com/'
driver.get(url)
sleep(5)

#click on chart
charts=driver.find_element(By.XPATH,'/html/body/div[3]/header/div/div[2]/div/div/div
charts.click()
sleep(5)

# click on hot 100
hot_100=driver.find_element(By.XPATH,'/html/body/div[3]/main/div[2]/div[1]/div[1]/di
hot_100.click()
sleep(5)

#click on close
close=driver.find_element(By.XPATH,'/html/body/div[2]/div/span')
close.click()

#getting all data into listed form
this_week=driver.find_elements(By.XPATH,'//div[@class="o-chart-results-list-row-cont
song=driver.find_elements(By.XPATH,'//div[@class="o-chart-results-list-row-container
artist=driver.find_elements(By.XPATH,'//div[@class="o-chart-results-list-row-contain
last_week_rank=driver.find_elements(By.XPATH,'//div[@class="o-chart-results-list-row
peak_rank=driver.find_elements(By.XPATH,'//div[@class="o-chart-results-list-row-cont
weeks=driver.find_elements(By.XPATH,'//div[@class="o-chart-results-list-row-containe

#creating empty list
hot_100_songs=[]
#creating empty dictionary for temporary purpose
temp_dict={}

# iterate all listed data to create listed dictionary
for i in range(len(this_week)):
    temp_dict={"This Week Rank":this_week[i].text,
               "Song Name": song[i].text,
               "Artist Name": artist[i].text,
               "Last Week Rank": last_week_rank[i].text ,
               "Peak Rank" : peak_rank[i].text,
               "Weeks On Chart": weeks[i].text}
    hot_100_songs.append(temp_dict)

# Quit the driver
driver.quit()

# create DataFrame
df_hot_100_songs=pd.DataFrame(hot_100_songs)
df_hot_100_songs
```

Out[7]:

	This Week Rank	Song Name	Artist Name	Last Week Rank	Peak Rank	Weeks On Chart
0	1	Last Night	Morgan Wallen	1	1	21
1	2	Fast Car	Luke Combs	3	2	13
2	3	Calm Down	Rema & Selena Gomez	4	3	42
3	4	Flowers	Miley Cyrus	2	1	23
4	5	All My Life	Lil Durk Featuring J. Cole	5	2	6
...	...	...	...	...	...	...
95	96	Angel, Pt. 1	Kodak Black, NLE Choppa, Jimin, JVKE & Muni Long	-	65	2
96	97	Girl In Mine	Parmalee	-	97	1
97	98	Moonlight	Kali Uchis	90	80	11
98	99	Classy 101	Feid x Young Miko	-	99	1
99	100	Bluffin	Gucci Mane & Lil Baby	-	100	1

100 rows × 6 columns

Q6.

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

In [8]:

```
# loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()

#loading the url into driver
url='https://www.theguardian.com/news/datablog/2012/aug/09/best-selling-books-all-ti
driver.get(url)
sleep(10)

# getting all data from the table into listed form
rank=driver.find_elements(By.XPATH,'//table[@class="in-article sortable"]/tbody/tr/t
title=driver.find_elements(By.XPATH,'//table[@class="in-article sortable"]/tbody/tr/
author=driver.find_elements(By.XPATH,'//table[@class="in-article sortable"]/tbody/tr
volume_sales=driver.find_elements(By.XPATH,'//table[@class="in-article sortable"]/tb
publisher=driver.find_elements(By.XPATH,'//table[@class="in-article sortable"]/tbody
genere=driver.find_elements(By.XPATH,'//table[@class="in-article sortable"]/tbody/tr

#creating empty list
novels=[]
# creating empty dictionary for temporary purpose
temp_dict={}

# Iterate listed data to form listed dictionary
for i in range(len(rank)):
    temp_dict={"RANK": rank[i].text,
               "TITLE": title[i].text,
               "AUTHOR" : author[i].text,
               "VOLUMES SOLD":volume_sales[i].text,
               "PUBLISHER": publisher[i].text,
               "GENERE":genere[i].text}
    novels.append(temp_dict)

# Quit the driver
driver.quit()

# making DataFrame
novel_df=pd.DataFrame(novels)
novel_df
```

Out[8]:

	RANK	TITLE	AUTHOR	VOLUMES SOLD	PUBLISHER	GENERE
0	1	Da Vinci Code,The	Brown, Dan	5,094,805	Transworld	Crime, Thriller & Adventure
1	2	Harry Potter and the Deathly Hallows	Rowling, J.K.	4,475,152	Bloomsbury	Children's Fiction
2	3	Harry Potter and the Philosopher's Stone	Rowling, J.K.	4,200,654	Bloomsbury	Children's Fiction
3	4	Harry Potter and the Order of the Phoenix	Rowling, J.K.	4,179,479	Bloomsbury	Children's Fiction
4	5	Fifty Shades of Grey	James, E. L.	3,758,936	Random House	Romance & Sagas
...	...	...	...	...	...	...
95	96	Ghost,The	Harris, Robert	807,311	Random House	General & Literary Fiction
96	97	Happy Days with the Naked Chef	Oliver, Jamie	794,201	Penguin	Food & Drink: General
97	98	Hunger Games,The:Hunger Games Trilogy	Collins, Suzanne	792,187	Scholastic Ltd.	Young Adult Fiction
98	99	Lost Boy,The:A Foster Child's Search for the L...	Pelzer, Dave	791,507	Orion	Biography: General
99	100	Jamie's Ministry of Food:Anyone Can Learn to C...	Oliver, Jamie	791,095	Penguin	Food & Drink: General

100 rows × 6 columns

Q7.

Scrape the details most watched tv series of all time from imdb.com. Url = https://www.imdb.com/list/ls095964455/ (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 [9]:

```
# Loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()

# Loading the url page
url='https://www.imdb.com/list/ls095964455/'
driver.get(url)
sleep(10)

# getting all necessary data into listed form
rank=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div/h3/s
name=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div/h3/a
year_span=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div
genere=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div/p/
run_time=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div/
ratings=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div[2
votes=driver.find_elements(By.XPATH,'//div[@class="lister-item mode-detail"]/div[2]/

# creating empty list
tv_series=[]
# Creating empty dictionary for temporary purpose
temp_disc={}

# Iterate all the listed data to form listed dictionary
for i in range(len(rank)):
    temp_disc={"RANK":rank[i].text,
               "NAME":name[i].text,
               "YEAR OF SPAN": year_span[i].text,
               "GENERE":genere[i].text,
               "RUN TIME":run_time[i].text,
               "RATINGS":ratings[i].text,
               "VOTES":votes[i].text}
    tv_series.append(temp_disc)

# Quit the driver
driver.quit()

#making dataframe:
df_tv_series=pd.DataFrame(tv_series)
df_tv_series
```

Out[9]:

	RANK	NAME	YEAR OF SPAN	GENERE	RUN TIME	RATINGS	VOTES
0	1.	Game of Thrones	(2011–2019)	Action, Adventure, Drama	57 min	9.2	2,173,715
1	2.	Stranger Things	(2016–2024)	Drama, Fantasy, Horror	51 min	8.7	1,251,542
2	3.	The Walking Dead	(2010–2022)	Drama, Horror, Thriller	44 min	8.1	1,032,493
3	4.	13 Reasons Why	(2017–2020)	Drama, Mystery, Thriller	60 min	7.5	303,560
4	5.	The 100	(2014–2020)	Drama, Mystery, Sci-Fi	43 min	7.6	262,731
...	...	...	...	...	...	...	...
95	96.	Reign	(2013–2017)	Drama	42 min	7.4	51,957
96	97.	A Series of Unfortunate Events	(2017–2019)	Adventure, Comedy, Drama	50 min	7.8	63,993
97	98.	Criminal Minds	(2005–)	Crime, Drama, Mystery	42 min	8.1	208,546
98	99.	Scream: The TV Series	(2015–2019)	Comedy, Crime, Drama	45 min	7.1	43,402
99	100.	The Haunting of Hill House	(2018)	Drama, Horror, Mystery	572 min	8.6	260,203

100 rows × 7 columns

Q8.

Details of Datasets from UCI machine learning repositories. Url = https://archive.ics.uci.edu/ (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 [10]:

```
# Loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()
# Loading the url into the driver
url='https://archive.ics.uci.edu/'
driver.get(url)
sleep(5)

# accept button
accept_button=driver.find_element(By.XPATH,'//button[@class="btn-primary btn-sm btn
accept_button.click()

# click on datasets
datasets=driver.find_element(By.XPATH,'/html/body/div/div[1]/div[1]/header/nav/ul/li
datasets.click()
sleep(5)

# Getting URL of each dataset
url=[]
for i in range(63):
    urls=driver.find_elements(By.XPATH,'//div[@class="relative col-span-8 sm:col-spa
    for u in urls:
        url.append(u.get_attribute("href"))

    #click on next button
    if len(url)<=620:
        next_button=driver.find_element(By.XPATH,'//div[@class="btn-group"]/button[2
        next_button.click()
        sleep(5)
print(len(url))

# creating empty list:
name=[]
data_type=[]
task=[]
attribute_type=[]
no_of_instance=[]
year=[]

# Iterate each url and get necessary data
for u in url:
    driver.get(u)
    sleep(5)

    # Dataset Name
    try:
        nm=driver.find_element(By.XPATH,'//h1[@class="text-3xl font-semibold text-pr
        name.append(nm.text)
    except NoSuchElementException:
        name.append(None)

    # Data type
    try:
        d_type=driver.find_element(By.XPATH,'//div[@class="grid grid-cols-8 gap-4 md
        data_type.append(d_type.text)
    except NoSuchElementException:
        data_type.append(None)

    # Associated Tasks
```

```
    try:
        tsk=driver.find_element(By.XPATH,'//div[@class="grid grid-cols-8 gap-4 md:gr
        task.append(tsk.text)
    except NoSuchElementException:
        task.append(None)

    # Attribute Type
    try:
        attribute=driver.find_element(By.XPATH,'//div[@class="grid grid-cols-8 gap-4
        attribute_type.append(attribute.text)
    except NoSuchElementException:
        attribute_type.append(None)

    # No of Instances
    try:
        instance=driver.find_element(By.XPATH,'//div[@class="grid grid-cols-8 gap-4
        no_of_instance.append(instance.text)
    except NoSuchElementException:
        no_of_instance.append(None)

    # Year
    try:
        yr=driver.find_element(By.XPATH,'//h2[@class="text-primary-content"]')
        year.append(yr.text.split("/")[-1])
    except NoSuchElementException:
        year.append(None)

# Quit the Driver
driver.quit()

# Creating DataFrame
df_dataset=pd.DataFrame({"Dataset Name":name,
                         "Data Type":data_type,
                         "Associated Tasks":task,
                         "Attribute Type":attribute_type,
                         "No of Instances":no_of_instance,
                         "Year": year})
df_dataset
```

623

Out[10]:

	Dataset Name	Data Type	Associated Tasks	Attribute Type	No of Instances	Year
0	Iris	Multivariate	Classification	Real	150	1988
1	Heart Disease	Multivariate	Classification	Categorical, Integer, Real	303	1988
2	Adult	Multivariate	Classification	Categorical, Integer	48842	1996
3	Dry Bean Dataset	Multivariate	Classification	Integer, Real	13611	2020
4	Diabetes	Multivariate, Time-Series	-	Categorical, Integer	-	None
...	...	...	...	...	...	...
618	PMU-UD	Univariate	Classification	-	5180	2018
619	Undocumented	-	-	-	-	None
620	EBL Domain Theories	-	-	-	-	None
621	Moral Reasoner	Domain-Theory	-	-	202	1994
622	DGP2 - The Second Data Generation Program	Data-Generator	-	Real	-	None

623 rows × 6 columns

Q9.

Scrape the details of Data science recruiters Url = https://www.naukri.com/hr-recruiters-consultants (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 [11]:

```
#loading the driver
driver=webdriver.Chrome(r"chromedriver.exe")
driver.maximize_window()
# loading the url page
url='https://www.naukri.com/hr-recruiters-consultants'
driver.get(url)
sleep(5)

#click on pop up
try:
    got_it=driver.find_element(By.XPATH,'/html/body/div[1]/div[4]/div[2]/div/button'
    got_it.click()
except NoSuchElementException:
    print("exception handled")

# click on home page
home=driver.find_element(By.XPATH,'/html/body/div[1]/div[3]/div[2]/a/img')
home.click()
sleep(3)

# write on searchbox
input_search=driver.find_element(By.XPATH,'/html/body/div[1]/div[6]/div/div/div[1]/d
input_search.send_keys("Data Science")

#click on search button
search_button=driver.find_element(By.XPATH,'/html/body/div[1]/div[6]/div/div/div[6]'
search_button.click()
sleep(5)

# create empty list
designation=[]
company=[]
skills=[]
location=[]

# designation
try:
    desig=driver.find_elements(By.XPATH,'//a[@class="title ellipsis"]')
    for i in desig:
        designation.append(i.text)
except NoSuchElementException:
    designation.append(None)

# Company Name
try:
    comp=driver.find_elements(By.XPATH,'//div[@class="companyInfo subheading"]/a[1]'
    for i in comp:
        company.append(i.text)
except NoSuchElementException:
    company.append(None)


# Slills required
try:
    skill=driver.find_elements(By.XPATH,'//ul[@class="tags has-description"]')
    for i in skill:
        skills.append(i.text)
except NoSuchElementException:
    skills.append(None)
```

```
60
61  # job location
62  try:
63      locations=driver.find_elements(By.XPATH,'//li[@class="fleft br2 placeHolderLi lo
64      for i in locations:
65          location.append(i.text)
66  except NoSuchElementException:
67      location.append(None)
68
69  print("length of designation= ",len(designation))
70  print("length of company= ",len(company))
71  print("length of skills= ",len(skills))
72  print("length of locations= ",len(location))
73
74  # Quit driver
75  driver.quit()
76
77  # create dataframe
78  naukri_df=pd.DataFrame({"Designation": designation,
79                          "Company": company,
80                          "Skills":skills,
81                          "Location":location})
82  naukri_df
```

```
length of designation=  20
length of company=  20
length of skills=  20
length of locations=  20
```

Out[11]:

	Designation	Company	Skills	L
0	L&D Trainer - Python & Data Science/Data Analy...	AVE-Promagne Business Solutions	Data Analytics\nIT training\nTraining\nMachine...	Kolkata, M Hyderabad/Secund
1	Data Science Engineer	Bizongo	Vision\nAnalytics\nDeep Learning\nNetworking\n...	Bangalore/Be
2	Engineer II-Data Science & Analytics	Raytheon Technologies	Data Science\nstatistical modeling\nmachine le...	Hybrid - Ba Be Karnataka
3	Data Science Engineer	Augusta Infotech	python\nnumpy\nmachine learning\ntensorflow\nP...	Bangalore/ Be Karnataka(Electro
4	Senior Analyst, Data Science	DUN BRADSTREET INFORMATION SERVICES INDIA PRIV...	Analysis\nAnalytical\nData Science\nData analy...	Hyderabad/Secun
5	Senior Lead Consultant - Data Science	All About It India	Computer\nUnix\nData Science\nAnalytical\nData...	Kolkata, Mumb Hyderabad/Sec
6	Data Science Analyst	Accenture	Machine learning\nPredictive modeling\nData mi...	Bangalore/Be
7	Data Science Senior Analyst	Accenture	Data Science\nPredictive Modeling\nmachine lea...	
8	Data Science - Intern	Zupee	Machine Learning\nNeural networks\nDeep Learni...	Delh
9	Software Engineer, Data Science	Epiq Systems	Data Science\nSpark\nScikit-learn\nKeras\nSoft...	Hyderabad/Secun
10	Senior Software Engineer, Data Science	Epiq Systems, Inc.	Languages\nDevelopment\nProduct management\nDa...	Hyderabad/Secund Canad
11	Senior Analyst, Data Science	Venator Holdings	Recruitment\nSenior\nStatistical modeling\nDat...	Hyderabad/Secun
12	Data Science, AI/ML Professional	QA InfoTech Pvt. Ltd	Cloud\nCloud computing\nData Science\nSimulati...	Bangalore/Be