FLIP ROBO

ASSIGNMENT-1

WEB SCRAPING

In all the following questions, you have to use BeautifulSoup to scrape different websites and collect data as per the requirement of the question.

Every answer to the question should be in form of a python function which should take URL as the parameter. Use Jupyter Notebooks to program, upload it on your GitHub and send the link of the Jupyter notebook to your SME.

1) **Write a python program to display all the header tags from wikipedia.org and make data frame.**

Answer)

```
#importing the required libraries
import requests
from bs4 import BeautifulSoup
import pandas as pd
url = "https://en.wikipedia.org/wiki/Main_Page"
response = requests.get(url)
soup = BeautifulSoup(response.content, "html.parser")

headers = soup.find_all(["h1", "h2", "h3", "h4", "h5", "h6"])
headers
# Loop through each header tag and append its text to the list
header_texts = []
for header in headers:
    header_texts.append(header.text)
# Create a Pandas DataFrame from the list of header tag text
df = pd.DataFrame(header_texts, columns=["Header"])
# Print the DataFrame
df
```

2) **Write s python program to display list of respected former presidents of India(i.e. Name , Term ofoffice)from https://presidentofindia.nic.in/former-presidents.htm and make data frame.**

Answer)

```
from bs4 import BeautifulSoup

import requests

page=requests.get('https://presidentofindia.nic.in/former-presidents.htm')

page

soup=BeautifulSoup(page.content)

soup

Name=[]

for i in soup.find_all('h3'):

    Name.append(i.text)
Name
Term=[]
for i in soup.find_all('p'):
    Term.append(i.text)
Term
Detail=[]
for i in soup.find_all('div', class_="presidentListing"):
    Detail.append(i.text)
Detail
import pandas as pd
```

```
df=pd.DataFrame({"PresidentiaL List":Detail})
df
```

3) Write a python program to scrape cricket rankings from icc-cricket.com. You have to scrape and make data frame-

a) Top 10 ODI teams in men's cricket along with the records for matches, points and rating.

Answer)

```
import requests
from bs4 import BeautifulSoup
import pandas as pd
# Send a GET request to the URL of the ODI team rankings page
url = 'https://www.icc-cricket.com/rankings/mens/team-rankings/odi'
response = requests.get(url)
# Parse the HTML content of the page using BeautifulSoup
soup = BeautifulSoup(response.content, 'html.parser')
# Extract the table containing the team rankings data
table = soup.find('table', class_='table')

# Extract the data from the table and store it in lists
teams = []
matches = []
points = []
ratings = []

for row in table.tbody.find_all('tr'):
    team = row.find('span', class_='u-hide-phablet').text.strip()
    match = row.find_all('td')[2].text.strip()
    point = row.find_all('td')[3].text.strip()
    rating = row.find_all('td')[4].text.strip()
    teams.append(team)
    matches.append(match)
    points.append(point)
    ratings.append(rating)
# Create a pandas data frame to display the data
data = {'Team': teams, 'Matches': matches, 'Points': points, 'Rating': ratings}
df = pd.DataFrame(data)
df.index += 1  # Start the index from 1
df = df.head(10)  # Display only the top 10 teams
print(df)
```

b) Top 10 ODI Batsmen along with the records of their team and rating.

Answer)

```
req=requests.get('https://www.icc-cricket.com/rankings/mens/player-rankings/odi/batting')
soup=BeautifulSoup(req.content)
player=soup.find_all('tr',class_=('rankings-block__banner','table-body'))
top10=player[0:10]
data={'Player_Name':[],'Team_Name': [], 'Rating':[]}
for i in top10:
    bat=i.find_all('td',recursive=True)
    data['Player_Name'].append(bat[1].text.replace('\n',''))
    data['Team_Name'].append(bat[2].text.replace('\n',''))
    data['Rating'].append(bat[3].text.replace('\n',''))
ODIBATSMAN=pd.DataFrame(data,index=range(1,11))
    ODIBATSMAN
```

c) Top 10 ODI bowlers along with the records of their team andrating.

Answer)

```
req=requests.get('https://www.icc-cricket.com/rankings/mens/player-rankings/odi/bowling')
soup=BeautifulSoup(req.content)
bowler=soup.find_all('tr',class_=('rankings-block__banner','table-body'))
Top10=bowler[0:10]
bdata={'Player_Name':[],'Team_Name': [], 'Rating':[]}
for i in Top10:
   bat=i.find_all('td',recursive=True)
   bdata['Player_Name'].append(bat[1].text.replace('\n',''))
   bdata['Team_Name'].append(bat[2].text.replace('\n',''))
   bdata['Rating'].append(bat[3].text.replace('\n',''))
ODIBOWL=pd.DataFrame(bdata,index=range(1,11))
    ODIBOWL
```

4) Write a python program to scrape cricket rankings from **icc-cricket.com**. You have to scrape and make **data frame-**

a) Top **10 ODI teams** in women's cricket along with the records for **matches, points and rating**.

Answer)

```
from bs4 import BeautifulSoup
import requests
import pandas as pd
req=requests.get('https://www.icc-cricket.com/rankings/womens/team-rankings/odi')
soup=BeautifulSoup(req.content)
team=soup.find_all('tr',class_=('rankings-block__banner','table-body'))
top10=team[0:10]
data = {'Team_Name':[],'Matches': [],'Points': [],'Rating':[]}

for i in top10:
   pnt=i.find_all('td',recursive=True)
   data['Team_Name'].append(i.find('span',class_='u-hide-phablet').text)
   data['Matches'].append(pnt[2].text)
   data['Points'].append(pnt[3].text)
   data['Rating'].append(pnt[4].text.strip().replace('\n',''))
WomenTeam=pd.DataFrame(data,index=range(1,11))
    WomenTeam
```

b) Top **10 women's ODI Batting** players along with the records of their **team and rating**.

Answer)

```
req=requests.get('https://www.icc-cricket.com/rankings/womens/player-rankings/odi/batting')
soup=BeautifulSoup(req.content)
player=soup.find_all('tr',class_=('rankings-block__banner','table-body'))

top10=player[0:10]
data={'Player_Name':[],'Team_Name': [], 'Rating':[]}
```

```
for i in top10:
    bat=i.find_all('td',recursive=True)
    data['Player_Name'].append(bat[1].text.replace('\n',''))
    data['Team_Name'].append(bat[2].text.replace('\n',''))
    data['Rating'].append(bat[3].text.replace('\n',''))
BATW=pd.DataFrame(data,index=range(1,11))
    BATW
```

c) Top 10 women's ODI all-rounder along with the records of their team and rating.

Answer)

```
req=requests.get('https://www.icc-cricket.com/rankings/womens/player-rankings/odi/all-rounder')
soup=BeautifulSoup(req.content)
player=soup.find_all('tr',class_=('rankings-block__banner','table-body'))

top10=player[0:10]
data={'Player_Name':[],'Team_Name': [], 'Rating':[]}

for i in top10:
  bat=i.find_all('td',recursive=True)
  data['Player_Name'].append(bat[1].text.replace('\n',''))
  data['Team_Name'].append(bat[2].text.replace('\n',''))
  data['Rating'].append(bat[3].text.replace('\n',''))
W_All=pd.DataFrame(data,index=range(1,11))
W_All
```

5) Write a python program to scrape mentioned news details from https://www.cnbc.com/world/?region=world and make **data frame-**

i) Headline

Answer)

```
import requests
from bs4 import BeautifulSoup
page=requests.get('https://www.cnbc.com/world/?region=world')
page
news=BeautifulSoup(page.content)
    news
# Headline
Headline=[]
for i in news.find_all('div',class_='RiverHeadline-headline RiverHeadline-hasThumbnail'):
  Headline.append(i.text)
    Headline
```

ii) Time

Answer)

```
# Time
Time=news.find('time')
Time
Time.text
Time=[]
for i in news.find_all('time'):
  Time.append(i.text)
```

Time

iii) **News Link**

Answer)

```
# Newslink
url = "https://www.cnbc.com/world/?region=world"
webpage = requests.get(url)
trav = BeautifulSoup(webpage.content, "html.parser")
for link in trav.find_all('a'):
    print(type(link), " ", link)
      trav.text
```

6) Write a python program to scrape the details of most downloaded articles from AI in last 90 days.https://www.journals.elsevier.com/artificial-intelligence/most-downloaded-articles

Scrape below mentioned details and make **data frame-**

i) Paper Title

ii) Authors

iii) Published Date

iv) Paper URL

Answer)

```
#importing the required libraries
import requests
from bs4 import BeautifulSoup
import pandas as pd

#sending request to get the html code of the webpage
#diplaying whether the page url is scrapable/accessible
page=requests.get("https://www.journals.elsevier.com/artificial-intelligence/most-downloaded-articles")
page

#getting page html content
soup=BeautifulSoup(page.text,'html.parser')
soup

#creating empty lists for saving the titles and other details

Paper_Title=[]
Authors=[]
Pub_Date=[]
Paper_URL=[]

#for extracting the paper titles
Papers=soup.find_all('h2',class_='sc-1qrq3sd-1 gRGSUS sc-1nmom32-0 sc-1nmom32-1 btcbYu goSKRg')
for i in Papers:
    Paper_Title.append(i.get_text())

#for extracting the Author names
author=soup.find_all('span',class_='sc-1w3fpd7-0 dnCnAO')
for i in author:
    Authors.append(i.get_text())

# for extracting the published date
PubDate=soup.find_all('span',class_='sc-1thf9ly-2 dvggWt')
for i in PubDate:
    Pub_Date.append(i.get_text())
```

```
#for extracting the paper URL
soup2=soup.find('div',class_='sc-orwwe2-3 jOMrrY').find_all('a')
for i in soup2:
    Paper_URL.append(i.get('href', None))

Most_Downloaded=pd.DataFrame({})
Most_Downloaded['Title']=Paper_Title
Most_Downloaded['Author']=Authors
Most_Downloaded['Publish Date']=Pub_Date
Most_Downloaded['URL']=Paper_URL
Most_Downloaded
```

7) Write a python program to scrape mentioned details from **dineout.co.in** and make **data frame-**

i) Restaurant name

Answer)

```
import requests
from bs4 import BeautifulSoup
page=requests.get('https://www.dineout.co.in/delhi-restaurants/welcome-back')
page
soup=BeautifulSoup(page.content)
soup
# Restaurant Name
RN=soup.find('div',class_="restnt-info cursor")
RN
RN.text
RN=[]
for i in soup.find_all('div',class_="restnt-info cursor"):
    RN.append(i.text)
RN
```

ii) Cuisine

Answer)

```
# Cuisines
cuisine=[]
for i in soup.find_all('span',class_="double-line-ellipsis"):
    cuisine.append(i.text.split('|')[1])
cuisine
```

iii) Location

Answer)

```
    # Location
    location=[]
    for i in soup.find_all('div',class_="restnt-loc ellipsis"):
        location.append(i.text)
          location
```

iv) Ratings

Answer)

```
    # Rating
    rating=[]
    for i in soup.find_all('div',class_="restnt-rating rating-4"):
        rating.append(i.text)
          rating
```

v) Image URL

Answer)

```
    # Images
    images=[]
    for i in soup.find_all('img',class_="no-img"):
        images.append(i['data-src'])
          images
    import pandas as pd
    df = pd.DataFrame({'Names':RN,'Cuisine':cuisine,'Locations':location,'Ratings':rating,'Images_url':images})
          df
```