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

In [1]:

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
html = urlopen('https://en.wikipedia.org/wiki/Main_Page')
bs = BeautifulSoup(html, "html.parser")
titles = bs.find_all(['h1', 'h2','h3','h4','h5','h6'])
print('List all the header tags :', *titles, sep='\n\n')
```

```
List all the header tags :

<h1 class="firstHeading mw-first-heading" id="firstHeading" style="display: none"><sp
an class="mw-page-title-main">Main Page</span></h1>

<h1><span class="mw-headline" id="Welcome_to_Wikipedia">Welcome to <a href="/wiki/Wik
ipedia" title="Wikipedia">Wikipedia</a></span></h1>

<h2 class="mp-h2" id="mp-tfa-h2"><span id="From_today.27s_featured_article"></span><s
pan class="mw-headline" id="From_today's_featured_article">From today's featured arti
cle</span></h2>

<h2 class="mp-h2" id="mp-dyk-h2"><span class="mw-headline" id="Did_you_know_...">Did
you know ...</span></h2>

<h2 class="mp-h2" id="mp-itn-h2"><span class="mw-headline" id="In_the_news">In the ne
ws</span></h2>

<h2 class="mp-h2" id="mp-otd-h2"><span class="mw-headline" id="On_this_day">On this d
ay</span></h2>

<h2 class="mp-h2" id="mp-tfp-h2"><span id="Today.27s_featured_picture"></span><span c
lass="mw-headline" id="Today's_featured_picture">Today's featured picture</span></h2>

<h2 class="mp-h2" id="mp-other"><span class="mw-headline" id="Other_areas_of_Wikipedi
a">Other areas of Wikipedia</span></h2>

<h2 class="mp-h2" id="mp-sister"><span id="Wikipedia.27s_sister_projects"></span><spa
n class="mw-headline" id="Wikipedia's_sister_projects">Wikipedia's sister projects</s
pan></h2>

<h2 class="mp-h2" id="mp-lang"><span class="mw-headline" id="Wikipedia_languages">Wik
ipedia languages</span></h2>
```

Write a python program to scrape cricket rankings from icc-cricket.com. You have to scrape and make data framea) Top 10 ODI teams in men’s cricket along with the records for matches, points and rating.

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

In [32]:

```
import requests
from bs4 import BeautifulSoup
import pandas as pd

# Scrape Top 10 ODI teams in women's cricket
url_teams = "https://www.icc-cricket.com/rankings/womens/team-rankings/odi"
response_teams = requests.get(url_teams)
soup_teams = BeautifulSoup(response_teams.content, "html.parser")

teams_data = []
table_teams = soup_teams.find("table", class_="table")
rows_teams = table_teams.find_all("tr")

for row in rows_teams[1:11]:
  team_name = row.find("span", class_="u-hide-phablet").text.strip()
  matches = row.find_all("td")[2].text.strip()
  points = row.find_all("td")[3].text.strip()
  rating = row.find_all("td")[4].text.strip()
  teams_data.append([team_name, matches, points, rating])

# Scrape Top 10 women's ODI Batting players
url_batting = "https://www.icc-cricket.com/rankings/womens/player-rankings/odi/batting
response_batting = requests.get(url_batting)
soup_batting = BeautifulSoup(response_batting.content, "html.parser")

batting_data = []
table_batting = soup_batting.find("table", class_="table")
rows_batting = table_batting.find_all("tr")

for row in rows_batting[1:11]:
  player_name = row.find("td", class_="table-body__cell rankings-table__name name").te
  team = row.find("span", class_="table-body__logo-text").text.strip()
  rating = row.find("td", class_="table-body__cell rating").text.strip()
  batting_data.append([player_name, team, rating])

# Scrape Top 10 women's ODI all-rounders
url_allrounders = "https://www.icc-cricket.com/rankings/womens/player-rankings/odi/all
response_allrounders = requests.get(url_allrounders)
soup_allrounders = BeautifulSoup(response_allrounders.content, "html.parser")

allrounders_data = []
table_allrounders = soup_allrounders.find("table", class_="table")
rows_allrounders = table_allrounders.find_all("tr")

for row in rows_allrounders[1:11]:
  player_name = row.find("td", class_="table-body__cell rankings-table__name name").te
  team = row.find("span", class_="table-body__logo-text").text.strip()
  rating = row.find("td", class_="table-body__cell rating").text.strip()
  allrounders_data.append([player_name, team, rating])

# Create data frames
df_teams = pd.DataFrame(teams_data, columns=["Team", "Matches", "Points", "Rating"])
df_batting = pd.DataFrame(batting_data, columns=["Player", "Team", "Rating"])
df_allrounders = pd.DataFrame(allrounders_data, columns=["Player", "Team", "Rating"])
```

```
# Print the data frames
print("Top 10 ODI teams in men's cricket:")
print(df_teams)
print("\nTop 10 men's ODI Batting players:")
print(df_batting)
print("\nTop 10 men's ODI all-rounders:")
print(df_allrounders)
```

```
---------------------------------------------------------------------------
AttributeError                            Traceback (most recent call last)
Cell In[32], line 31
     28 rows_batting = table_batting.find_all("tr")
     30 for row in rows_batting[1:11]:
---> 31   player_name = row.find("td", class_="table-body__cell rankings-table__name 
name").text.strip()
     32   team = row.find("span", class_="table-body__logo-text").text.strip()
     33   rating = row.find("td", class_="table-body__cell rating").text.strip()

AttributeError: 'NoneType' object has no attribute 'text'
```

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.

In [33]:

```
from lxml import html
import requests
response = requests.get(' https://presidentofindia.nic.in/former-presidents.htm')
doc = html.fromstring(response.text)
title = doc.cssselect('h3.dataset-heading')[0].text_content()
print("The name of the most recently added dataset on data.gov:")
print(title.strip())
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
Cell In[33], line 5
      3 response = requests.get(' https://presidentofindia.nic.in/former-presidents.h
tm')
      4 doc = html.fromstring(response.text)
----> 5 title = doc.cssselect('h3.dataset-heading')[0].text_content()
      6 print("The name of the most recently added dataset on data.gov:")
      7 print(title.strip())

IndexError: list index out of range
```

Write a python program to scrape cricket rankings from icc-cricket.com. You have to scrape and make data framea) Top 10 ODI teams in men’s cricket along with the records for matches, points and rating.

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

In [ ]:

```
import requests
from bs4 import BeautifulSoup
import pandas as pd

# Scrape Top 10 ODI teams in women's cricket
url_teams = "https://www.icc-cricket.com/rankings/womens/team-rankings/odi"
response_teams = requests.get(url_teams)
soup_teams = BeautifulSoup(response_teams.content, "html.parser")

teams_data = []
table_teams = soup_teams.find("table", class_="table")
rows_teams = table_teams.find_all("tr")

for row in rows_teams[1:11]:
  team_name = row.find("span", class_="u-hide-phablet").text.strip()
  matches = row.find_all("td")[2].text.strip()
  points = row.find_all("td")[3].text.strip()
  rating = row.find_all("td")[4].text.strip()
  teams_data.append([team_name, matches, points, rating])

# Scrape Top 10 women's ODI Batting players
url_batting = "https://www.icc-cricket.com/rankings/womens/player-rankings/odi/batting
response_batting = requests.get(url_batting)
soup_batting = BeautifulSoup(response_batting.content, "html.parser")

batting_data = []
table_batting = soup_batting.find("table", class_="table")
rows_batting = table_batting.find_all("tr")

for row in rows_batting[1:11]:
  player_name = row.find("td", class_="table-body__cell rankings-table__name name").te
  team = row.find("span", class_="table-body__logo-text").text.strip()
  rating = row.find("td", class_="table-body__cell rating").text.strip()
  batting_data.append([player_name, team, rating])

# Scrape Top 10 women's ODI all-rounders
url_allrounders = "https://www.icc-cricket.com/rankings/womens/player-rankings/odi/all
response_allrounders = requests.get(url_allrounders)
soup_allrounders = BeautifulSoup(response_allrounders.content, "html.parser")

allrounders_data = []
table_allrounders = soup_allrounders.find("table", class_="table")
rows_allrounders = table_allrounders.find_all("tr")

for row in rows_allrounders[1:11]:
  player_name = row.find("td", class_="table-body__cell rankings-table__name name").te
  team = row.find("span", class_="table-body__logo-text").text.strip()
  rating = row.find("td", class_="table-body__cell rating").text.strip()
  allrounders_data.append([player_name, team, rating])

# Create data frames
df_teams = pd.DataFrame(teams_data, columns=["Team", "Matches", "Points", "Rating"])
df_batting = pd.DataFrame(batting_data, columns=["Player", "Team", "Rating"])
df_allrounders = pd.DataFrame(allrounders_data, columns=["Player", "Team", "Rating"])
```

```
# Print the data frames
print("Top 10 ODI teams in women's cricket:")
print(df_teams)
print("\nTop 10 women's ODI Batting players:")
print(df_batting)
print("\nTop 10 women's ODI all-rounders:")
print(df_allrounders)
```

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 framei) Paper Title ii) Authors iii) Published Date iv) Paper URL

In [ ]:

```
from newspaper import Article

#A new article from TOI
url = "https://www.journals.elsevier.com/artificial-intelligence/most-downloaded-artic

#For different language newspaper refer above table
toi_article = Article(url, language="en") # en for English

#To download the article
toi_article.download()

#To parse the article
toi_article.parse()

#To perform natural language processing ie..nlp
toi_article.nlp()

#To extract title
print("Article's Title:")
print(toi_article.title)
print("n")

#To extract text
print("Article's Text:")
print(toi_article.text)
print("n")

#To extract summary
print("Article's Summary:")
print(toi_article.summary)
print("n")

#To extract keywords
print("Article's Keywords:")
print(toi_article.keywords)
```

Write a python program to scrape mentioned details from dineout.co.inand make data framei) Restaurant name

ii) Cuisine iii) Location iv) Ratings v) Image URL

In [23]:

```
class Restaurant():
    """A class representing a restaurant."""

    def __init__(self, name, cuisine_type):
        """Initialize the restaurant."""
        self.name = name.title()
        self.cuisine_type = cuisine_type

    def describe_restaurant(self):
        """Display a summary of the restaurant."""
        msg = self.name + " serves wonderful " + self.cuisine_type + "."
        print("\n" + msg)

    def open_restaurant(self):
        """Display a message that the restaurant is open."""
        msg = self.name + " is open. Come on in!"
        print("\n" + msg)

restaurant = Restaurant('the bunglow', 'pizza,burgur,chinese,indian food')
print(restaurant.name)
print(restaurant.cuisine_type)

restaurant.describe_restaurant()
restaurant.open_restaurant()
```

```
The Bunglow
pizza,burgur,chinese,indian food

The Bunglow serves wonderful pizza,burgur,chinese,indian food.

The Bunglow is open. Come on in!
```

Write a python program to scrape cricket rankings from icc-cricket.com. You have to scrape and make data framea) Top 10 ODI teams in women's cricket along with the records for matches, points and rating.

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

In [ ]:

```
class User():
    """Represent a simple user profile."""
    def __init__( location):
        """Initialize the user."""
        self.location = location.title()

    def describe_user(self):
        """Display a summary of the user's information."""

        print("Location" )


eric = User('Dehradun','Rajpur')
eric.describe_user()
eric.greet_user()

print = User('Dehradun','Rajpur')
```

In [ ]: