Question no.1) Write a python program to display all the header tags from 'en.wikipedia.org/wiki/Main_Page'

In [1]:
```
from bs4 import BeautifulSoup
import requests
```

In [3]:
```
page = requests.get('https://en.wikipedia.org/wiki/Main_Page')
page
```

Out[3]:
```
<Response [200]>
```

In [4]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[4]:
```
<!DOCTYPE html>
<html class="client-nojs" dir="ltr" lang="en">
<head>
<meta charset="utf-8"/>
<title>Wikipedia, the free encyclopedia</title>
<script>document.documentElement.className="client-js";RLCONF={"wgBreakFrame
s":!1,"wgSeparatorTransformTable":["",""],"wgDigitTransformTable":["",""],"wg
DefaultDateFormat":"dmy","wgMonthNames":["","January","February","March","Apr
il","May","June","July","August","September","October","November","Decembe
r"],"wgRequestId":"8ff6e124-5145-418f-9e2d-e0ce28b3f866","wgCSPNonce":!1,"wgC
anonicalNamespace":"","wgCanonicalSpecialPageName":!1,"wgNamespaceNumber":
0,"wgPageName":"Main_Page","wgTitle":"Main Page","wgCurRevisionId":100459352
0,"wgRevisionId":1004593520,"wgArticleId":15580374,"wgIsArticle":!0,"wgIsRedi
rect":!1,"wgAction":"view","wgUserName":null,"wgUserGroups":["*"],"wgCategori
es":[],"wgPageContentLanguage":"en","wgPageContentModel":"wikitext","wgReleva
ntPageName":"Main_Page","wgRelevantArticleId":15580374,"wgIsProbablyEditabl
e":!1,"wgRelevantPageIsProbablyEditable":!1,"wgRestrictionEdit":["sysop"],"wg
RestrictionMove":["sysop"],"wgIsMainPage":!0,"wgFlaggedRevsParams":{
"tags":{"status":{"levels":-1}}},"wgMediaViewerOnClick":!0,"wgMediaViewerEnab
```

In [6]:
```
Header_tags = []


# Looping the values
for i in soup.find_all('span',class_="mw-headline"):
    Header_tags.append(i.text)  #append values into the empty list

Header_tags
```

Out[6]:
```
["From today's featured article",
 'Did you know\xa0...',
 'In the news',
 'On this day',
 "Today's featured picture",
 'Other areas of Wikipedia',
 "Wikipedia's sister projects",
 'Wikipedia languages']
```

---- -:- ----

Question no.2) Write a python program to display IMDB's Top rated 100 movies' data (i.e. Name, IMDB rating, Year of release) and make data frame.

In [7]:
```
page = requests.get('https://www.imdb.com/list/ls091520106/')
page
```

Out[7]:
```
<Response [200]>
```

In [8]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[8]:
```
<!DOCTYPE html>
<html xmlns:fb="http://www.facebook.com/2008/fbml" xmlns:og="http://ogp.me/ns
#">
<head>
<meta charset="utf-8"/>
<meta content="IE=edge" http-equiv="X-UA-Compatible"/>
<script type="text/javascript">var IMDbTimer={starttime: new Date().getTime
(),pt:'java'};</script>
<script>
    if (typeof uet == 'function') {
      uet("bb", "LoadTitle", {wb: 1});
    }
</script>
<script>(function(t){ (t.events = t.events || {})["csm_head_pre_title"] = new
Date().getTime(); })(IMDbTimer);</script>
<title>Top 100 Movies Bucket List - IMDb</title>
<script>(function(t){ (t.events = t.events || {})["csm_head_post_title"] = ne
w Date().getTime(); })(IMDbTimer);</script>
<script>
```

In [9]:
```
Name = []

for i in soup.find_all('h3',class_="lister-item-header"):
    Name.append(i.text.split('\n')[2])

Name
```

Out[9]:
```
['The Shawshank Redemption',
 'The Godfather',
 'The Godfather: Part II',
 'The Dark Knight',
 '12 Angry Men',
 "Schindler's List",
 'The Lord of the Rings: The Return of the King',
 'Pulp Fiction',
 'Il buono, il brutto, il cattivo',
 'Fight Club',
 'Joker',
 'The Lord of the Rings: The Fellowship of the Ring',
 'Forrest Gump',
 'Inception',
 'Star Wars: Episode V - The Empire Strikes Back',
 'The Lord of the Rings: The Two Towers',
 'The Matrix',
 "One Flew Over the Cuckoo's Nest",
 'Goodfellas',
```

In [ ]:
```

```

In [10]:
```
Year = []

for i in soup.find_all('h3',class_="lister-item-header"):
    Year.append(i.text.split('\n')[3])

Year
```

Out[10]:
```
['(1994)',
 '(1972)',
 '(1974)',
 '(2008)',
 '(1957)',
 '(1993)',
 '(2003)',
 '(1994)',
 '(1966)',
 '(1999)',
 '(2019)',
 '(2001)',
 '(1994)',
 '(2010)',
 '(1980)',
 '(2002)',
 '(1999)',
 '(1975)',
 '(1990)',
```

In [ ]:
```

```

In [11]:
```
Rating = []

for i in soup.find_all('div',class_="ipl-rating-star small"):
    Rating.append(i.text.split('\n')[8])

Rating
```

Out[11]:
```
['9.3',
 '9.2',
 '9',
 '9',
 '9',
 '8.9',
 '8.9',
 '8.9',
 '8.8',
 '8.8',
 '8.4',
 '8.8',
 '8.8',
 '8.8',
 '8.7',
 '8.7',
 '8.7',
 '8.7',
 '8.7',
```

In [13]:
```
print(len(Name),len(Year),len(Rating))
```

```
100 100 100
```

In [15]:
```
import pandas as pd

df = pd.DataFrame({'Name':Name, 'IMDB Rating':Rating, 'Year of Release':Year

df
```

Out[15]:

	Name	IMDB Rating	Year of Release
0	The Shawshank Redemption	9.3	(1994)
1	The Godfather	9.2	(1972)
2	The Godfather: Part II	9	(1974)
3	The Dark Knight	9	(2008)
4	12 Angry Men	9	(1957)
...	...	...	...
95	North by Northwest	8.3	(1959)
96	A Clockwork Orange	8.3	(1971)
97	Snatch	8.3	(2000)
98	Le fabuleux destin d'Amélie Poulain	8.3	(2001)
99	The Kid	8.3	(1921)

100 rows × 3 columns

In [ ]:
```

```

---- -:- ----

Question no.3) Write a python program to display IMDB's Top rated 100 Indian movies' data (i.e. Name, IMDB rating, Year of release) and make data frame.

In [16]:
```
page = requests.get('https://www.imdb.com/india/top-rated-indian-movies/')
page
```

Out[16]:
```
<Response [200]>
```

In [17]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[17]:
```
<!DOCTYPE html>
<html xmlns:fb="http://www.facebook.com/2008/fbml" xmlns:og="http://ogp.me/ns
#">
<head>
<meta charset="utf-8"/>
<meta content="IE=edge" http-equiv="X-UA-Compatible"/>
<style>
                body#styleguide-v2 {
                    background: no-repeat fixed center top #000;
                }
            </style>
<style>
            body#styleguide-v2 #root {
                box-shadow: none;
            }
        </style>
<script type="text/javascript">var IMDbTimer={starttime: new Date().getTime
(),pt:'java'};</script>
<script>
```

In [18]:
```
name = []
for i in soup.find_all('td',class_="titleColumn"):
    name.append(i.text.split('\n')[2])

name
```

Out[18]:
```
['      Sardar Udham',
 '      Nayakan',
 '      Anbe Sivam',
 '      Pariyerum Perumal',
 '      C/o Kancharapalem',
 '      Manichitrathazhu',
 '      Golmaal',
 '      Kireedam',
 '      Apur Sansar',
 '      Natsamrat',
 '      96',
 '      Thevar Magan',
 '      Kumbalangi Nights',
 '      Black Friday',
 '      Pather Panchali',
 '      Soorarai Pottru',
 '      #Home',
 '      Visaaranai',
 '      3 Idiots',
```

In [19]:
```
year = []

for i in soup.find_all('td',class_="titleColumn"):
    year.append(i.text.split('\n')[3])

year
```

Out[19]:
```
['(2021)',
 '(1987)',
 '(2003)',
 '(2018)',
 '(2018)',
 '(1993)',
 '(1979)',
 '(1989)',
 '(1959)',
 '(2016)',
 '(2018)',
 '(1992)',
 '(2019)',
 '(2004)',
 '(1955)',
 '(2020)',
 '(2021)',
 '(2015)',
 '(2009)',
```

In [20]:
```
rating = []

for i in soup.find_all('td',class_="ratingColumn imdbRating"):
    rating.append(i.text.split('\n')[1])

rating
```

Out[20]:
```
['8.6',
 '8.5',
 '8.5',
 '8.5',
 '8.5',
 '8.5',
 '8.5',
 '8.5',
 '8.5',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
 '8.4',
```

In [21]:
```
import pandas as pd

df = pd.DataFrame({'Name':name[0:100], 'IMDB Rating':rating[0:100], 'Year of
df
```

Out[21]:

	Name	IMDB Rating	Year of Release
0	Sardar Udham	8.6	(2021)
1	Nayakan	8.5	(1987)
2	Anbe Sivam	8.5	(2003)
3	Pariyerum Perumal	8.5	(2018)
4	C/o Kancharapalem	8.5	(2018)
...	...	...	...
95	Roja	8.1	(1992)
96	Dil Chahta Hai	8.1	(2001)
97	Rang De Basanti	8.1	(2006)
98	OMG: Oh My God!	8.1	(2012)
99	Uri: The Surgical Strike	8.1	(2019)

100 rows × 3 columns

In [ ]:
```

```

---- -:- ----

Question 4) Write a python program to scrape cricket rankings from 'www.icc-cricket.com' (http://www.icc-cricket.com%E2%80%99). You have to scrape:

i) Top 10 ODI teams in men's cricket along with the records for matches, points and rating. ii) Top 10 ODI Batsmen in men along with the records of their team and rating. iii) Top 10 ODI bowlers along with the records of their team and rating.

Q4 (i)

In [22]:
```
page = requests.get('https://www.icc-cricket.com/rankings/mens/team-rankings
page
```

Out[22]:
```
<Response [200]>
```

In [23]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[23]:
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta content="ICC Ranking for ODI teams International Cricket Council" name
="twitter:title"/>
<meta content="website" property="og:type"/>
<meta content="summary_large_image" property="twitter:card"/>
<meta content="Official International Cricket Council ranking for One Day Int
ernational (ODI) cricket teams. Discover latest ICC rankings table, predict u
pcoming matches, see points and ratings for all teams." name="description"/>
<meta content="@icc" property="twitter:site"/>
<meta content="Official International Cricket Council ranking for One Day Int
ernational (ODI) cricket teams. Discover latest ICC rankings table, predict u
pcoming matches, see points and ratings for all teams." name="twitter:descrip
tion"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
humbnail.jpg" name="twitter:image"/>
<meta content="ICC Ranking for ODI teams International Cricket Council" prope
rty="og:title"/>
```

In [61]:
```
Teams = []

for i in soup.find_all('span',class_="u-hide-phablet"):
    Teams.append(i.text)

Teams = Teams[0:10]

# Teams for the Top 10 ODI Men's Cricket
Teams
```

Out[61]:
```
['New Zealand',
 'England',
 'Australia',
 'India',
 'South Africa',
 'Pakistan',
 'Bangladesh',
 'West Indies',
 'Sri Lanka',
 'Afghanistan']
```

In [27]:
```
banner_matches = []

for i in soup.find_all('td',class_="rankings-block__banner--matches"):
    banner_matches.append(i.text.split(',')[0])

banner_matches
```

Out[27]:
```
['17']
```

In [36]:
```
table_body_matches = []

for i in soup.find_all('td',class_="table-body__cell u-center-text"):
    table_body_matches.append(i.text.split(',')[0])

table_body_matches = table_body_matches[0:18]
table_body_matches = table_body_matches[0::2]
table_body_matches
```

Out[36]:
```
['32', '28', '32', '25', '27', '30', '30', '32', '17']
```

In [60]:
```
Matches = banner_matches + table_body_matches

# Matches for the Top 10 ODI Men's Cricket
Matches
```

Out[60]:
```
['17', '32', '28', '32', '25', '27', '30', '30', '32', '17']
```

In [38]:
```
banner_points = []
for i in soup.find_all('td',class_="rankings-block__banner--points"):
    banner_points.append(i.text)

banner_points
```

Out[38]:
```
['2,054']
```

In [42]:
```
table_body_points = []
for i in soup.find_all('td',class_="table-body__cell u-center-text"):
    table_body_points.append(i.text.split(','))

table_body_points = table_body_points[0:18]
table_body_points = table_body_points[1::2]
table_body_points
```

Out[42]:
```
[['3', '793'],
 ['3', '244'],
 ['3', '624'],
 ['2', '459'],
 ['2', '524'],
 ['2', '740'],
 ['2', '523'],
 ['2', '657'],
 ['1', '054']]
```

In [59]:
```
Points = banner_points + table_body_points

# Points for the Top 10 ODI Men's Cricket
Points
```

Out[59]:
```
['2,054',
 ['3', '793'],
 ['3', '244'],
 ['3', '624'],
 ['2', '459'],
 ['2', '524'],
 ['2', '740'],
 ['2', '523'],
 ['2', '657'],
 ['1', '054']]
```

In [51]:
```
banner_rating = []

for i in soup.find_all('td',class_="rankings-block__banner--rating u-text-ri
    banner_rating.append(i.text.split('\n')[1])



banner_rating
```

Out[51]:
```
['                            121']
```

In [54]:
```
table_body_rating = []
for i in soup.find_all('td',class_="table-body__cell u-text-right rating"):
    table_body_rating.append(i.text)

table_body_rating = table_body_rating[0:9]
table_body_rating
```

Out[54]:
```
['119', '116', '113', '98', '93', '91', '84', '83', '62']
```

In [58]:
```
Rating = banner_rating + table_body_rating


# Rating for the Top 10 ODI Men's Cricket
Rating
```

Out[58]:
```
['                            121',
 '119',
 '116',
 '113',
 '98',
 '93',
 '91',
 '84',
 '83',
 '62']
```

In [ ]:
```
1
```

In [ ]:
```
1
```

Q4 (ii)

In [68]:
```
1 page = requests.get('https://www.icc-cricket.com/rankings/mens/player-rankin
2 page
```

Out[68]:
```
<Response [200]>
```

In [69]:
```
1 soup = BeautifulSoup(page.content)
2 soup
```

Out[69]:
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
name="twitter:title"/>
<meta content="website" property="og:type"/>
<meta content="summary_large_image" property="twitter:card"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="description"/>
<meta content="@icc" property="twitter:site"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="twitter:description"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
humbnail.jpg" name="twitter:image"/>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
property="og:title"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
```

In [72]:
```
1 Banner_Name = []
2 for i in soup.find_all('div',class_="rankings-block__banner--name-large"):
3     Banner_Name.append(i.text)
4
5 Banner_Name
```

Out[72]:
```
['Babar Azam']
```

In [77]:
```
table_body_names = []
for i in soup.find_all('td',class_="table-body__cell rankings-table__name na
    table_body_names.append(i.text.split('\n')[1])

table_body_names = table_body_names[0:9]
table_body_names
```

Out[77]:
```
['Virat Kohli',
 'Rohit Sharma',
 'Ross Taylor',
 'Aaron Finch',
 'Jonny Bairstow',
 'David Warner',
 'Shai Hope',
 'Kane Williamson',
 'Quinton de Kock']
```

In [94]:
```
Names = Banner_Name + table_body_names

# Top 10 ODI Batsmen in men names :-
Names
```

Out[94]:
```
['Babar Azam',
 'Virat Kohli',
 'Rohit Sharma',
 'Ross Taylor',
 'Aaron Finch',
 'Jonny Bairstow',
 'David Warner',
 'Shai Hope',
 'Kane Williamson',
 'Quinton de Kock']
```

In [ ]:
```

```

In [81]:
```
banner_team = []
for i in soup.find_all('div',class_="rankings-block__banner--nationality"):
    banner_team.append(i.text.split('\n')[2])

banner_team
```

Out[81]:
```
['PAK']
```

In [83]:
```
table_body_teams = []
for i in soup.find_all('span',class_="table-body__logo-text"):
    table_body_teams.append(i.text)


table_body_teams = table_body_teams[0:9]
table_body_teams
```

Out[83]:
```
['IND', 'IND', 'NZ', 'AUS', 'ENG', 'AUS', 'WI', 'NZ', 'SA']
```

In [93]:
```
Teams = banner_team + table_body_teams

# Top 10 ODI Batsmen in men Teams :-
Teams
```

Out[93]: ['PAK', 'IND', 'IND', 'NZ', 'AUS', 'ENG', 'AUS', 'WI', 'NZ', 'SA']

In [87]:
```
banner_rating = []
for i in soup.find_all('div',class_="rankings-block__banner--rating"):
    banner_rating.append(i.text)

banner_rating
```

Out[87]: ['873']

In [90]:
```
table_body_ratings = []
for i in soup.find_all('td',class_="table-body__cell rating"):
    table_body_ratings.append(i.text)


table_body_ratings = table_body_ratings[0:9]
table_body_ratings
```

Out[90]: ['844', '813', '801', '779', '775', '762', '758', '754', '747']

In [92]:
```
Ratings = banner_rating + table_body_ratings

#Top 10 ODI Batsmen in men Rating :-
Ratings
```

Out[92]: ['873', '844', '813', '801', '779', '775', '762', '758', '754', '747']

In [ ]:
```

```

In [ ]:
```

```

Q4 (iii)

In [95]:
```
page = requests.get('https://www.icc-cricket.com/rankings/mens/player-rankin
page
```

Out[95]: <Response [200]>

In [96]:
```
soup= BeautifulSoup(page.content)
soup
```

Out[96]:
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
name="twitter:title"/>
<meta content="website" property="og:type"/>
<meta content="summary_large_image" property="twitter:card"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="description"/>
<meta content="@icc" property="twitter:site"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="twitter:description"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
humbnail.jpg" name="twitter:image"/>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
property="og:title"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
```

In [97]:
```
banner_bowler = []

for i in soup.find_all('div',class_="rankings-block__banner--name-large"):
    banner_bowler.append(i.text)

banner_bowler
```

Out[97]:
```
['Trent Boult']
```

In [98]:
```
table_body_bowlers = []

for i in soup.find_all('td',class_="table-body__cell rankings-table__name na
    table_body_bowlers.append(i.text.split('\n')[1])

table_body_bowlers = table_body_bowlers[0:9]

table_body_bowlers
```

Out[98]:
```
['Josh Hazlewood',
 'Mujeeb Ur Rahman',
 'Chris Woakes',
 'Mehedi Hasan',
 'Matt Henry',
 'Jasprit Bumrah',
 'Mitchell Starc',
 'Shakib Al Hasan',
 'Kagiso Rabada']
```

In [99]:
```
# Top 10 men's ODI Bowlers names :-

Bowlers = banner_bowler + table_body_bowlers
Bowlers
```

Out[99]:
```
['Trent Boult',
 'Josh Hazlewood',
 'Mujeeb Ur Rahman',
 'Chris Woakes',
 'Mehedi Hasan',
 'Matt Henry',
 'Jasprit Bumrah',
 'Mitchell Starc',
 'Shakib Al Hasan',
 'Kagiso Rabada']
```

In [100]:
```
banner_team = []

for i in soup.find_all('div',class_="rankings-block__banner--nationality"):
    banner_team.append(i.text.split('\n')[2])

banner_team
```

Out[100]:
```
['NZ']
```

In [101]:
```
table_body_teams = []

for i in soup.find_all('td',class_="table-body__cell nationality-logo rankin
    table_body_teams.append(i.text.split('\n')[2])

table_body_teams = table_body_teams[0:9]
table_body_teams
```

Out[101]:
```
['AUS', 'AFG', 'ENG', 'BAN', 'NZ', 'IND', 'AUS', 'BAN', 'SA']
```

In [102]:
```
# Top 10 men's ODI Bowlers Teams :-

Teams = banner_team + table_body_teams
Teams
```

Out[102]:
```
['NZ', 'AUS', 'AFG', 'ENG', 'BAN', 'NZ', 'IND', 'AUS', 'BAN', 'SA']
```

In [103]:
```
banner_rating = []

for i in soup.find_all('div',class_="rankings-block__banner--rating"):
    banner_rating.append(i.text)

banner_rating
```

Out[103]:
```
['737']
```

In [104]:
```
table_body_ratings = []

for i in soup.find_all('td',class_="table-body__cell rating"):
    table_body_ratings.append(i.text)

table_body_ratings = table_body_ratings[0:9]
table_body_ratings
```

Out[104]: ['709', '708', '700', '692', '691', '679', '652', '650', '646']

In [105]:
```
# Top 10 men's ODI Bowlers ratings :-

Ratings = banner_rating + table_body_ratings
Ratings
```

Out[105]: ['737', '709', '708', '700', '692', '691', '679', '652', '650', '646']

In [ ]:
```

```

In [ ]:
```

```

---- -:- ----

Question no.5) Write a python program to scrape cricket rankings from 'www.icc-cricket.com' (http://www.icc-cricket.com%E2%80%99). You have to scrape:

i) Top 10 ODI teams in women's cricket along with the records for matches, points and rating. ii) Top 10 women's ODI players along with the records of their team and rating. iii) Top 10 women's ODI all-rounder along with the records of their team and rating

Q5 (i)

In [106]:
```
page = requests.get('https://www.icc-cricket.com/rankings/womens/team-rankin
page
```

Out[106]: <Response [200]>

In [107]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[107]:
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta content="ICC Ranking for ODI teams International Cricket Council" name
="twitter:title"/>
<meta content="website" property="og:type"/>
<meta content="summary_large_image" property="twitter:card"/>
<meta content="Official International Cricket Council rankings for test match
cricket teams. Discover latest ICC rankings table, predict upcoming matches,
see points and ratings for all teams." name="description"/>
<meta content="@icc" property="twitter:site"/>
<meta content="Official International Cricket Council rankings for test match
cricket teams. Discover latest ICC rankings table, predict upcoming matches,
see points and ratings for all teams." name="twitter:description"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
humbnail.jpg" name="twitter:image"/>
<meta content="ICC Ranking for ODI teams International Cricket Council" prope
rty="og:title"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
```

In [108]:
```
team = []

for i in soup.find_all('span',class_="u-hide-phablet"):
    team.append(i.text)

team = team[0:10]

# Top 10 ODI Women's tems:-
team
```

Out[108]:
```
['Australia',
 'England',
 'South Africa',
 'India',
 'New Zealand',
 'West Indies',
 'Pakistan',
 'Bangladesh',
 'Sri Lanka',
 'Ireland']
```

In [109]:
```
banner_matches = []

for i in soup.find_all('td',class_="rankings-block__banner--matches"):
    banner_matches.append(i.text)

banner_matches
```

Out[109]:
```
['21']
```

In [110]:
```
table_body_matches = []

for i in soup.find_all('td',class_="table-body__cell u-center-text"):
    table_body_matches.append(i.text)
table_body_matches = table_body_matches[0::2]
table_body_matches
```

Out[110]: ['25', '29', '26', '26', '22', '20', '5', '11', '2']

In [111]:
```
# Top 10 ODI Women's matches :-

matches = banner_matches + table_body_matches
matches
```

Out[111]: ['21', '25', '29', '26', '26', '22', '20', '5', '11', '2']

In [112]:
```
banner_points = []
for i in soup.find_all('td',class_="rankings-block__banner--points"):
    banner_points.append(i.text)
banner_points
```

Out[112]: ['3,379']

In [113]:
```
table_body_points = []
for i in soup.find_all('td',class_="table-body__cell u-center-text"):
    table_body_points.append(i.text)
table_body_points= table_body_points[1::2]

table_body_points
```

Out[113]: ['2,983', '3,390', '2,934', '2,392', '1,872', '1,496', '306', '519', '25']

In [114]:
```
# Top 10 ODI Women's points :-

Points = banner_points + table_body_points
Points
```

Out[114]:
```
['3,379',
 '2,983',
 '3,390',
 '2,934',
 '2,392',
 '1,872',
 '1,496',
 '306',
 '519',
 '25']
```

In [115]:
```
banner_rating = []
for i in soup.find_all('td',class_="rankings-block__banner--rating u-text-ri
    banner_rating.append(i.text.split('\n')[1])
banner_rating
```

Out[115]:
```
['                          161']
```

In [116]:
```
table_body_rating = []
for i in soup.find_all('td',class_="table-body__cell u-text-right rating"):
    table_body_rating.append(i.text)
table_body_rating
```

Out[116]:
```
['119', '117', '113', '92', '85', '75', '61', '47', '13']
```

In [117]:
```
# Top 10 ODI Women's rating :-

Rating = banner_rating + table_body_rating
Rating
```

Out[117]:
```
['                          161',
 '119',
 '117',
 '113',
 '92',
 '85',
 '75',
 '61',
 '47',
 '13']
```

In [ ]:
```

```

In [ ]:
```

```

Q5 (ii)

In [118]:
```
page = requests.get('https://www.icc-cricket.com/rankings/womens/player-rank
page
```

Out[118]:
```
<Response [200]>
```

In [119]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[119]:
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
name="twitter:title"/>
<meta content="website" property="og:type"/>
<meta content="summary_large_image" property="twitter:card"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="description"/>
<meta content="@icc" property="twitter:site"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="twitter:description"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
humbnail.jpg" name="twitter:image"/>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
property="og:title"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
```

In [120]:
```
player1_name = []

for i in soup.find_all('div',class_="rankings-block__banner--name-large"):
    player1_name.append(i.text)

player1_name
```

Out[120]:
```
['Lizelle Lee']
```

In [121]:
```
table_body_names = []

for i in soup.find_all('td',class_="table-body__cell rankings-table__name na
    table_body_names.append(i.text.split('\n')[1])

table_body_names = table_body_names[0:9]
table_body_names
```

Out[121]:
```
['Alyssa Healy',
 'Mithali Raj',
 'Tammy Beaumont',
 'Amy Satterthwaite',
 'Smriti Mandhana',
 'Meg Lanning',
 'Beth Mooney',
 'Heather Knight',
 'Laura Wolvaardt']
```

In [122]:
```
# Top 10 ODI Women's names :-
Names = player1_name + table_body_names
Names
```

Out[122]:
```
['Lizelle Lee',
 'Alyssa Healy',
 'Mithali Raj',
 'Tammy Beaumont',
 'Amy Satterthwaite',
 'Smriti Mandhana',
 'Meg Lanning',
 'Beth Mooney',
 'Heather Knight',
 'Laura Wolvaardt']
```

In [123]:
```
player1_team = []

for i in soup.find_all('div',class_="rankings-block__banner--nationality"):
    player1_team.append(i.text.split('\n')[2])

player1_team
```

Out[123]:
```
['SA']
```

In [124]:
```
table_body_teams = []

for i in soup.find_all('span',class_="table-body__logo-text"):
    table_body_teams.append(i.text)
table_body_teams = table_body_teams[0:9]
table_body_teams
```

Out[124]:
```
['AUS', 'IND', 'ENG', 'NZ', 'IND', 'AUS', 'AUS', 'ENG', 'SA']
```

In [125]:
```
# Top 10 ODI Women's teams :-

Teams = player1_team + table_body_teams
Teams
```

Out[125]:
```
['SA', 'AUS', 'IND', 'ENG', 'NZ', 'IND', 'AUS', 'AUS', 'ENG', 'SA']
```

In [126]:
```
player1_rating = []

for i in soup.find_all('div',class_="rankings-block__banner--rating"):
    player1_rating.append(i.text)
player1_rating
```

Out[126]:
```
['761']
```

In [127]:
```
table_body_rating = []
for i in soup.find_all('td',class_="table-body__cell rating"):
    table_body_rating.append(i.text)
table_body_rating = table_body_rating[0:9]
table_body_rating
```

Out[127]:
```
['750', '738', '728', '717', '710', '699', '690', '674', '672']
```

In [128]:
```
# Top 10 ODI Women's rating :-

Rating = player1_rating + table_body_rating
Rating
```

Out[128]:
```
['761', '750', '738', '728', '717', '710', '699', '690', '674', '672']
```

In [ ]:
```

```

In [ ]:
```

```

Q5 (iii)

In [129]:
```
page = requests.get('https://www.icc-cricket.com/rankings/womens/player-rank
page
```

Out[129]:
```
<Response [200]>
```

In [130]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[130]:
```
<!DOCTYPE html>
<html lang="en">
<head>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
name="twitter:title"/>
<meta content="website" property="og:type"/>
<meta content="summary_large_image" property="twitter:card"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="description"/>
<meta content="@icc" property="twitter:site"/>
<meta content="Official ICC Cricket website - live matches, scores, news, hig
hlights, commentary, rankings, videos and fixtures from the International Cri
cket Council." name="twitter:description"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
humbnail.jpg" name="twitter:image"/>
<meta content="Live Cricket Scores &amp;amp; News International Cricket Council"
property="og:title"/>
<meta content="https://www.icc-cricket.com/resources/ver/i/elements/default-t
```

In [131]:
```
player1_name = []
for i in soup.find_all('div',class_="rankings-block__banner--name-large"):
    player1_name.append(i.text)
player1_name
```

Out[131]:
```
['Marizanne Kapp']
```

In [132]:
```
table_body_names = []
for i in soup.find_all('td',class_="table-body__cell rankings-table__name na
    table_body_names.append(i.text.split('\n')[1])
table_body_names = table_body_names[0:9]
table_body_names
```

Out[132]:
```
['Natalie Sciver',
 'Ellyse Perry',
 'Stafanie Taylor',
 'Deepti Sharma',
 'Ashleigh Gardner',
 'Dane van Niekerk',
 'Jess Jonassen',
 'Katherine Brunt',
 'Jhulan Goswami']
```

In [133]:
```
# Top 10 ODI Women's names :-

Names = player1_name + table_body_names
Names
```

Out[133]:
```
['Marizanne Kapp',
 'Natalie Sciver',
 'Ellyse Perry',
 'Stafanie Taylor',
 'Deepti Sharma',
 'Ashleigh Gardner',
 'Dane van Niekerk',
 'Jess Jonassen',
 'Katherine Brunt',
 'Jhulan Goswami']
```

In [134]:
```
player1_team = []
for i in soup.find_all('div',class_="rankings-block__banner--nationality"):
    player1_team.append(i.text.split('\n')[2])
player1_team
```

Out[134]:
```
['SA']
```

In [135]:
```
table_body_teams = []
player1_name = []
for i in soup.find_all('span',class_="table-body__logo-text"):
    table_body_teams.append(i.text)
table_body_teams = table_body_teams[0:9]
table_body_teams
```

Out[135]: ['ENG', 'AUS', 'WI', 'IND', 'AUS', 'SA', 'AUS', 'ENG', 'IND']

In [136]:
```
# Top 10 ODI Women's team :-

Teams = player1_team + table_body_teams
Teams
```

Out[136]: ['SA', 'ENG', 'AUS', 'WI', 'IND', 'AUS', 'SA', 'AUS', 'ENG', 'IND']

In [137]:
```
player1_rating = []
for i in soup.find_all('div',class_="rankings-block__banner--rating"):
    player1_rating.append(i.text)
player1_rating
```

Out[137]: ['384']

In [138]:
```
table_body_rating = []
for i in soup.find_all('td',class_="table-body__cell rating"):
    table_body_rating.append(i.text)
table_body_rating =  table_body_rating[0:9]
table_body_rating
```

Out[138]: ['372', '365', '322', '299', '275', '274', '272', '272', '251']

In [139]:
```
# Top 10 ODI Women's rating :-

Rating = player1_rating + table_body_rating
Rating
```

Out[139]: ['384', '372', '365', '322', '299', '275', '274', '272', '272', '251']

In [ ]:
```

```

In [ ]:
```

```

---- -:- ----

Question no.6) Write a python program to scrape details of all the mobile phones under Rs. 20,000 listed on Amazon.in. The scraped data should include Product Name, Price, Image URL and Average Rating.

In [141]:
```
page = requests.get('https://www.amazon.in/Mobile-Phone-Under-20000-Rupees/s
page
```

Out[141]: <Response [200]>

In [142]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[142]:
```
<!DOCTYPE html>
<html class="a-no-js" data-19ax5a9jf="dingo" lang="en-in"><!-- sp:feature:hea
d-start -->
<head><script>var aPageStart = (new Date()).getTime();</script><meta charset
="utf-8"/>
<!-- sp:end-feature:head-start -->
<!-- sp:feature:cs-optimization -->
<meta content="on" http-equiv="x-dns-prefetch-control"/>
<link href="https://images-eu.ssl-images-amazon.com" rel="dns-prefetch"/>
<link href="https://m.media-amazon.com" rel="dns-prefetch"/>
<link href="https://completion.amazon.com" rel="dns-prefetch"/>
<!-- sp:end-feature:cs-optimization -->
<!-- sp:feature:aui-assets -->
<link href="https://images-eu.ssl-images-amazon.com/images/I/11EIQ5IGqaL._RC|
01ZTHTZObnL.css,41jsiPzxYxL.css,31qGOnSAToL.css,013z33uKh2L.css,017DsKjNQJL.c
ss,0131vqwP5UL.css,41EWOOlBJ9L.css,11TIuySqr6L.css,01ElnPiDxWL.css,11bGSgD5pD
L.css,01Dm5eKVxwL.css,01IdKcBuAdL.css,01y-XAlI+2L.css,21N4kUH7pxL.css,01oDR3I
ULNL.css,41-PwE7+H0L.css,21j0IlW7xKL.css,01XPHJk60-L.css,116uEeuY3+L.css,21aP
hFy+riL.css,11gneA3MtJL.css,21fecG8pUzL.css,01ulGzBW88L.css,01CFUgsA-YL.css,3
```

In [164]:
```
Names = []
for i in soup.find_all('span',class_="a-size-medium a-color-base a-text-norm
    Names.append(i.text.split('-'))


# Product Names for mobile phones under Rs. 20,000 :-
Names
```

Out[164]:
```
[['OPPO A74 5G (Fantastic Purple,6GB RAM,128GB Storage) ',
  ' 5G Android Smartphone | 5000 mAh Battery | 18W Fast Charge | 90Hz LCD Dis
play'],
 ['Redmi 9 (Sky Blue, 4GB RAM, 64GB Storage) | 2.3GHz Mediatek Helio G35 Octa
core Processor'],
 ['OPPO A74 5G (Fluid Black,6GB RAM,128GB Storage) ',
  ' 5G Android Smartphone | 5000 mAh Battery | 18W Fast Charge | 90Hz LCD Dis
play'],
 ['OPPO A31 (Fantasy White, 6GB RAM, 128GB Storage) with No Cost EMI/Addition
al Exchange Offers'],
 ['Redmi Note 10S (Frost White, 6GB RAM, 64GB Storage) ',
  ' Super Amoled Display | 64 MP Quad Camera | Alexa Built in'],
 ['Samsung Galaxy M12 (Blue,4GB RAM, 64GB Storage) 6000 mAh with 8nm Processo
r | True 48 MP Quad Camera | 90Hz Refresh Rate'],
 ['Redmi 9 Prime (Sunrise Flare, 4GB RAM, 64GB Storage)',
  ' Full HD+ Display & AI Quad Camera'],
 ['Tecno Spark 7T(Jewel Blue, 4GB RAM, 64GB Storage) 6000 mAh Battery| 48 MP
AI Dual Rear Camera'],
 ['OPPO A31 (Mystery Black, 6GB RAM, 128GB Storage) with No Cost EMI/Addition
```

In [165]:
```
Price = []

for i in soup.find_all('span',class_="a-price-whole"):
    Price.append(i.text)

# Prices for mobile phones under Rs. 20,000 :-
Price
```

Out[165]:
```
['15,990',
 '8,499',
 '15,990',
 '11,490',
 '13,999',
 '9,499',
 '10,499',
 '8,499',
 '11,490',
 '6,799',
 '15,999',
 '6,799',
 '7,999',
 '7,999',
 '9,499',
 '16,999']
```

In [158]:
```
Image_URL = []
for i in soup.find_all('div',class_="a-section aok-relative s-image-fixed-he
    Image_URL.append(i)

# Image URL for mobile phones under Rs. 20,000 :-
Image_URL
```

Out[158]:
```
[<div class="a-section aok-relative s-image-fixed-height"><img alt="OPPO A74
5G (Fantastic Purple,6GB RAM,128GB Storage) - 5G Android Smartphone | 5000 mA
h Battery | 18W Fast Charge | 90Hz L..." class="s-image" data-image-index="1"
data-image-latency="s-product-image" data-image-load="" data-image-source-den
sity="1" src="https://m.media-amazon.com/images/I/71geVdy6-OS._AC_UY218_.jpg"
srcset="https://m.media-amazon.com/images/I/71geVdy6-OS._AC_UY218_.jpg 1x, ht
tps://m.media-amazon.com/images/I/71geVdy6-OS._AC_UY327_QL65_.jpg (https://m.
media-amazon.com/images/I/71geVdy6-OS._AC_UY327_QL65_.jpg) 1.5x, https://m.me
dia-amazon.com/images/I/71geVdy6-OS._AC_UY436_QL65_.jpg (https://m.media-amaz
on.com/images/I/71geVdy6-OS._AC_UY436_QL65_.jpg) 2x, https://m.media-amazon.c
om/images/I/71geVdy6-OS._AC_UY545_QL65_.jpg (https://m.media-amazon.com/image
s/I/71geVdy6-OS._AC_UY545_QL65_.jpg) 2.5x, https://m.media-amazon.com/images/
I/71geVdy6-OS._AC_UY654_QL65_.jpg (https://m.media-amazon.com/images/I/71geVd
y6-OS._AC_UY654_QL65_.jpg) 3x"/></div>,
 <div class="a-section aok-relative s-image-fixed-height"><img alt="Redmi 9
 (Sky Blue, 4GB RAM, 64GB Storage) | 2.3GHz Mediatek Helio G35 Octa core Proc
essor" class="s-image" data-image-index="2" data-image-latency="s-product-ima
ge" data-image-load="" data-image-source-density="1" src="https://m.media-ama
zon.com/images/I/71A9Vo1BatL._AC_UY218_.jpg" srcset="https://m.media-amazon.c
```

In [162]:
```
average_rating = []
for i in soup.find_all('span',class_="a-size-base"):
    average_rating.append(i.text)


average_rating = average_rating[0:16]

# Average rating for mobile phones under Rs. 20,000 :-
average_rating
```

Out[162]:
```
['8,979',
 '99,433',
 '8,979',
 '32,663',
 '19,548',
 '51,370',
 '71,721',
 '6,886',
 '32,663',
 '20,753',
 '257,473',
 '86,196',
 '86,196',
 '86,196',
 '51,370',
 '1,861']
```

In [ ]:
```
1
```

In [ ]:
```
1
```

---- -:- ----

Question no.7) Write a python program to scrape house details from mentioned url. It should include house title, location, area, emi and price

https://www.nobroker.in/property/sale/bangalore/Electronic%20City?type=BHK4&searchParam=W3sibGF0IjoxMi44N (https://www.nobroker.in/property/sale/bangalore/Electronic%20City?type=BHK4&searchParam=W3sibGF0IjoxMi44N)

DUyMTQ1LCJsb24iOjc3LjY2MDE2OTUsInBsYWNlSWQiOiJDaElKdy1GUWQ0cHNyanNSSGZkYX
iLCJwbGFjZU5hbWUiOiJFbGVjdHJvbmljIENpdHkifV0=&propertyAge=0&radius=2.0"

In [167]:
```
1 page = requests.get('https://www.nobroker.in/property/sale/bangalore/Electro
2 page
```

Out[167]:
```
<Response [200]>
```

In [168]:
```
1 soup = BeautifulSoup(page.content)
2 soup
```

Out[168]:
```
<!DOCTYPE html>
<html lang="en"><head>
<meta content="794951570520699" property="fb:pages"/>
<link href="https://www.nobroker.in" rel="canonical"/>
<link href="//www.googletagmanager.com" rel="dns-prefetch"/>
<link href="//www.google-analytics.com" rel="dns-prefetch"/>
<link href="//assets.nobroker.in" rel="dns-prefetch"/>
<link href="//images.nobroker.in" rel="dns-prefetch"/>
<link href="//assets.nobroker.in/static/img/favicon.png" id="favicon" rel="sh
ortcut icon"/>
<link href="https://images.nobroker.in/static/img/fav64.png" rel="apple-touch
-icon"/>
<meta charset="utf-8"/><meta charset="utf-8"/>
<meta content="app-id=com.nobroker.app&amp;referrer=utm_source%3Dnobroker%26u
tm_medium%3DmobileWeb" name="google-play-app"/>
<meta content="app-id=1200507100, app-argument=nobrokerapp://" name="apple-it
unes-app"/>
<meta content="#fd3752" name="theme-color"/>
<meta content="flats for sale in Electronic City,  apartments for sale in El
```

In [169]:
```
# House titles of the properties :-

house_title = []
for i in soup.find_all('h2',class_="heading-6 font-semi-bold nb__1AShY"):
    house_title.append(i.text.split(','))

house_title
```

Out[169]:
```
[['2 BHK Apartment  For Sale  In Glr Vintage In Electronic City Phase I '],
 ['3 BHK Flat  For Sale  In Snn Raj Greenbay  In Electronic City Phase Ii '],
 ['3 BHK Flat  For Sale  In Vijaya Nest  In Electronic City '],
 ['2 BHK Apartment  For Sale  In Prestige Sunrise Park - Birchwood In Electroni
c City '],
 ['1 BHK Flat  For Sale  In Prestige Sunrise Park',
  ' Electronic City In Electronic City '],
 ['3 BHK Apartment  For Sale  In Smondoville In Electronic City Phase 1 '],
 ['3 BHK Apartment  For Sale  In Dlf Maiden Heights In Bommasandra '],
 ['1 BHK Flat  For Sale  In Snn Raj Greenbay - Apartments In Electronic City',
  ' Bangalore In Electronic City',
  'phase 2 '],
 ['2 BHK Apartment  For Sale  In Sree Sree Shine In Bommasandra '],
 ['2 BHK Apartment  For Sale  In Paras Maitri In Electronic City ']]
```

In [170]:
```
# Location of the properties :-

location = []
for i in soup.find_all('div',class_="nb__2CMjv"):
    location.append(i.text)

location
```

Out[170]:
```
['Glr Vintage\xa0 12th Cross, Neeladri Nagar, Electronic City Phase I, Bangalor
e, Karnataka, INDIA.',
 'Next to Tech Mahindra',
 'Anantha Nagar Road, Electronic City Phase II, Electronic City, Bengaluru, Kar
nataka, India',
 'Prestige Sunrise Park - Birchwood\xa0 Neotown Rd, Electronics City Phase 1, E
lectronic City, Bengaluru, Karnataka 560100, India',
 'Neotown Rd, Electronics City Phase 1, Electronic City, Bengaluru, Karnataka 5
60100, India',
 'Smondoville\xa0 Neotown Rd, Gollahalli, Electronic City, Bengaluru, Karnataka
560100, India',
 'Dlf Maiden Heights\xa0 DLF Maiden Heights, rajapura, Jigani, Karnataka 56010
5, India',
 '1st Main Road, Electronic City, Phase 2, Next to Tech Mahindra, Bengaluru, Ka
rnataka 560100, India',
 'Sree Sree Shine\xa0 Sree Sree Shine, Bommasandra Jigani Link Rd, Bommasandra
Industrial Area, Bengaluru, Karnataka 560099, India',
 'Paras Maitri\xa0 6th Cross, Anantha Nagar Rd, Ananth Nagar, Phase 1, Kammasan
dra, Electronic City, Vaddara Palya, Karnataka 560100, India']
```

In [171]:
```
# Area of the properties :-

area = []
for i in soup.find_all('div',class_="nb__3oNyC"):
    area.append(i.text)

area
```

Out[171]:
```
['1,110 sqft',
 '1,610 sqft',
 '1,505 sqft',
 '1,342 sqft',
 '630 sqft',
 '1,350 sqft',
 '1,252 sqft',
 '630 sqft',
 '985 sqft',
 '1,013 sqft']
```

In [172]:
```
# EMI of the properties :-

EMI = []
for i in soup.find_all('div',class_="font-semi-bold heading-6"):
    EMI.append(i.text)

EMI = EMI[1::3]
EMI
```

Out[172]:
```
['₹25,791/Month',
 '₹71,643/Month',
 '₹33,815/Month',
 '₹58,460/Month',
 '₹28,084/Month',
 '₹33,242/Month',
 '₹34,388/Month',
 '₹28,084/Month',
 '₹17,194/Month',
 '₹22,352/Month']
```

In [173]:
```
# Prices of the properties :-

price = []
for i in soup.find_all('div',class_="font-semi-bold heading-6"):
    price.append(i.text)
price = price[2::3]
price
```

Out[173]:
```
['₹45 Lacs',
 '₹1.25 Crores',
 '₹59 Lacs',
 '₹1.02 Crores',
 '₹49 Lacs',
 '₹58 Lacs',
 '₹60 Lacs',
 '₹49 Lacs',
 '₹30 Lacs',
 '₹39 Lacs']
```

In [ ]:
```

```

In [ ]:
```

```

---- -:- ----

Question no.8) Write a python program to scrape mentioned details from 'https://www.dineout.co.in/delhi-restaurants/buffet-special' (https://www.dineout.co.in/delhi-restaurants/buffet-special%E2%80%99) :

i) Restaurant name ii) Cuisine iii) Location iv) Ratings v) Image url

In [175]:
```
page = requests.get('https://www.dineout.co.in/delhi-restaurants/buffet-spec
page
```

Out[175]:
```
<Response [200]>
```

In [176]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[176]:
```
<!DOCTYPE html>
<html lang="en"><head><meta charset="utf-8"/><meta content="IE=edge" http-equ
iv="X-UA-Compatible"/><meta content="width=device-width, initial-scale=1.0, m
aximum-scale=1.0, user-scalable=no" name="viewport"/><link href="/manifest.js
on" rel="manifest"/><style type="text/css">
            @font-face {
                font-family: 'dineicon';
                src:  url('/fonts/dineicon.eot');
                src:  url('/fonts/dineicon.eot#iefix') format('embedded-opent
ype'),
                url('/fonts/dineicon.ttf') format('truetype'),
                url('/fonts/dineicon.woff') format('woff'),
                url('/fonts/dineicon.svg#dineicon') format('svg');
                font-weight: normal;
                                font-style: normal;
                                font-display: swap;
            }
            .hide {
                display: none !important;
            `
```

In [177]:
```
restaurant_name = []
for i in soup.find_all('div',class_="restnt-info cursor"):
    restaurant_name.append(i.text.split(',')[0])

# Restaurant Names:-
restaurant_name
```

Out[177]:
```
['Castle BarbequeConnaught Place',
 'Jungle Jamboree3CS Mall',
 'Castle BarbequePacific Mall',
 'Cafe KnoshThe Leela Ambience Convention Hotel',
 'The Barbeque CompanyGardens Galleria',
 'India GrillHilton Garden Inn',
 'Delhi BarbequeTaurus Sarovar Portico',
 'The Monarch - Bar Be Que VillageIndirapuram Habitat Centre',
 'World CafeVibe by The Lalit Traveller',
 'Indian Grill RoomSuncity Business Tower',
 'Mad 4 Bar B QueSector 29',
 'Barbeque 29NIT',
 'GlasshouseDoubleTree By Hilton Gurugram Baani Square']
```

In [178]:
```
cuisine = []
for i in soup.find_all('span',class_="double-line-ellipsis"):
    cuisine.append(i.text.split('|')[1])

# Cuisine names :-
cuisine
```

Out[178]:
```
[' Chinese, North Indian',
 ' North Indian, Barbecue, Italian, Asian',
 ' North Indian, Chinese',
 ' Multi-Cuisine, North Indian, Italian, Continental, Mediterranean',
 ' Barbecue, Chinese, Mughlai, North Indian',
 ' North Indian, Italian, Oriental ',
 ' Barbecue, North Indian',
 ' North Indian, Chinese, Fast Food',
 ' North Indian, Chinese, Continental',
 ' North Indian, Mughlai, Barbecue',
 ' North Indian, Mughlai',
 ' Barbecue, Chinese, North Indian',
 ' Multi-Cuisine, Asian, European, Italian, North Indian']
```

In [179]:
```
location = []
for i in soup.find_all('div',class_="restnt-loc ellipsis"):
    location.append(i.text)

# Location of the Restaraunt :-
location
```

Out[179]:
```
['Connaught Place, Central Delhi',
 '3CS Mall,Lajpat Nagar - 3, South Delhi',
 'Pacific Mall,Tagore Garden, West Delhi',
 'The Leela Ambience Convention Hotel,Shahdara, East Delhi',
 'Gardens Galleria,Sector 38A, Noida',
 'Hilton Garden Inn,Saket, South Delhi',
 'Taurus Sarovar Portico,Mahipalpur, South Delhi',
 'Indirapuram Habitat Centre,Indirapuram, Ghaziabad',
 'Vibe by The Lalit Traveller,Sector 35, Faridabad',
 'Suncity Business Tower,Golf Course Road, Gurgaon',
 'Sector 29, Faridabad',
 'NIT, Faridabad',
 'DoubleTree By Hilton Gurugram Baani Square,Sector 50, Gurgaon']
```

In [180]:
```
image_url = []
for i in soup.find_all('img',class_="no-img"):
    image_url.append(i)

# Image URL of the Restaurant :-
image_url
```

Out[180]:
```
[<img alt="Castle Barbeque" class="lazy-load-img no-img" data-gatype="Restaurant
tImageClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/shar
pen/8/k/b/p86792-16062953735fbe1f4d3fb7e.jpg?tr=tr:n-medium" data-url="/delhi/c
astle-barbeque-connaught-place-central-delhi-86792" data-w-onclick="cardClickHa
ndler"/>,
 <img alt="Jungle Jamboree" class="lazy-load-img no-img" data-gatype="Restaurant
tImageClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/shar
pen/5/a/k/p59633-16046474755fa4fa33c0e92.jpg?tr=tr:n-medium" data-url="/delhi/j
ungle-jamboree-lajpat-nagar-3-south-delhi-59633" data-w-onclick="cardClickHandl
er"/>,
 <img alt="Castle Barbeque" class="lazy-load-img no-img" data-gatype="Restaurant
tImageClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/shar
pen/3/j/o/p38113-15959192065f1fcb666130c.jpg?tr=tr:n-medium" data-url="/delhi/c
astle-barbeque-tagore-garden-west-delhi-38113" data-w-onclick="cardClickHandle
r"/>,
 <img alt="Cafe Knosh" class="lazy-load-img no-img" data-gatype="RestaurantImag
eClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sharpen/
4/j/v/p406-1634016633361651d79326d0.jpg?tr=tr:n-medium" data-url="/delhi/cafe-kn
osh-shahdara-east-delhi-406" data-w-onclick="cardClickHandler"/>,
 <img alt="The Barbeque Company" class="lazy-load-img no-img" data-gatype="Rest
aurantImageClick" data-src="https://im1.dineout.co.in/images/uploads/restauran
t/sharpen/7/q/d/p79307-16051787075fad15532bd7c.jpg?tr=tr:n-medium" data-url="/d
elhi/the-barbeque-company-sector-38a-noida-79307" data-w-onclick="cardClickHand
ler"/>,
 <img alt="India Grill" class="lazy-load-img no-img" data-gatype="RestaurantIma
geClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sharpen/
2/v/t/p2687-1482477169585cce712b90f.jpg?tr=tr:n-medium" data-url="/delhi/india-
grill-saket-south-delhi-2687" data-w-onclick="cardClickHandler"/>,
 <img alt="Delhi Barbeque" class="lazy-load-img no-img" data-gatype="Restaurant
ImageClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sharp
en/5/v/f/p52501-16006856545f68865616659.jpg?tr=tr:n-medium" data-url="/delhi/de
lhi-barbeque-mahipalpur-south-delhi-52501" data-w-onclick="cardClickHandler"/>,
 <img alt="The Monarch - Bar Be Que Village" class="lazy-load-img no-img" data-
gatype="RestaurantImageClick" data-src="https://im1.dineout.co.in/images/upload
s/restaurant/sharpen/3/n/o/p34822-15599107305cfa594a13c24.jpg?tr=tr:n-medium" d
ata-url="/delhi/the-monarch-bar-be-que-village-indirapuram-ghaziabad-34822" dat
a-w-onclick="cardClickHandler"/>,
 <img alt="World Cafe" class="lazy-load-img no-img" data-gatype="RestaurantImag
eClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sharpen/
1/p/y/p12366-1466935020576fa6ecdc359.jpg?tr=tr:n-medium" data-url="/delhi/world
-cafe-sector-35-faridabad-12366" data-w-onclick="cardClickHandler"/>,
 <img alt="Indian Grill Room" class="lazy-load-img no-img" data-gatype="Restaur
antImageClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sh
arpen/5/y/y/p549-15143767525a438e30b3e19.jpg?tr=tr:n-medium" data-url="/delhi/i
ndian-grill-room-golf-course-road-gurgaon-549" data-w-onclick="cardClickHandle
r"/>,
 <img alt="Mad 4 Bar B Que" class="lazy-load-img no-img" data-gatype="Restauran
tImageClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/shar
pen/4/j/e/p43488-15295778165b2b8158ceeef.jpg?tr=tr:n-medium" data-url="/delhi/m
```

```
ad-4-bar-b-que-sector-29-faridabad-43488" data-w-onclick="cardClickHandler"/>,
 <img alt="Barbeque 29" class="lazy-load-img no-img" data-gatype="RestaurantIma
geClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sharpen/
5/w/r/p58842-15624171585d209806d9143.jpg?tr=tr:n-medium" data-url="/delhi/barbe
que-29-nit-faridabad-58842" data-w-onclick="cardClickHandler"/>,
 <img alt="Glasshouse" class="lazy-load-img no-img" data-gatype="RestaurantImag
eClick" data-src="https://im1.dineout.co.in/images/uploads/restaurant/sharpen/
9/m/f/p9875-16057921085fb6716cc44f8.jpg?tr=tr:n-medium" data-url="/delhi/glassh
ouse-sector-50-gurgaon-9875" data-w-onclick="cardClickHandler"/>]
```

In [ ]:
```
1
```

In [ ]:
```
1
```

---- -:- ----

Question no.9) Write a python program to scrape weather details for last 24 hours from 'https://en.tutiempo.net/delhi.html?data=last-24-(https://en.tutiempo.net/delhi.html?data=last-24-) hours' :

i) Hour ii) Temperature iii) Wind iv) Weather condition v) Humidity vi) Pressure

In [181]:
```
1 page = requests.get('https://en.tutiempo.net/delhi.html?data=last-24-hours')
2 page
```

Out[181]:
```
<Response [200]>
```

In [182]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[182]:
```
<!DOCTYPE html>
<html class="pretty-scrollbar" lang="en"><head><meta content="text/html; char
set=utf-8" http-equiv="content-type"/><meta content="width=device-width, init
ial-scale=1.0, maximum-scale=1.0, user-scalable=no" name="viewport"/><meta co
ntent="Tutiempo Network, S.L." name="author"/><meta content="global" name="di
stribution"/><meta content="app-id=526005265" name="apple-itunes-app"/><meta
content="yes" name="mobile-web-app-capable"/><meta content="yes" name="apple-
mobile-web-app-capable"/><meta content="black" name="apple-mobile-web-app-sta
tus-bar-style"/><link href="/favicon.ico" rel="shortcut icon" type="image/ic
o"/><link href="/Browser/apple-icon-57x57.png" rel="apple-touch-icon" sizes
="57x57"/><link href="/Browser/apple-icon-60x60.png" rel="apple-touch-icon" s
izes="60x60"/><link href="/Browser/apple-icon-72x72.png" rel="apple-touch-ico
n" sizes="72x72"/><link href="/Browser/apple-icon-76x76.png" rel="apple-touch
-icon" sizes="76x76"/><link href="/Browser/apple-icon-114x114.png" rel="apple
-touch-icon" sizes="114x114"/><link href="/Browser/apple-icon-120x120.png" re
l="apple-touch-icon" sizes="120x120"/><link href="/Browser/apple-icon-144x14
4.png" rel="apple-touch-icon" sizes="144x144"/><link href="/Browser/apple-ico
n-152x152.png" rel="apple-touch-icon" sizes="152x152"/><link href="/Browser/a
pple-icon-180x180.png" rel="apple-touch-icon" sizes="180x180"/><link href="/B
```

In [194]:
```
hour = []

for i in soup.find_all('td',class_=""):
    hour.append(i.text.split('Mist'))


hour = hour[16:]
hour = hour[0::2]

# Hourly Weather details for last 24 hour :-
hour
```

Out[194]:
```
[['18:00'],
 ['17:30'],
 ['17:00'],
 ['16:30'],
 ['16:00'],
 ['15:30'],
 ['15:00'],
 ['14:30'],
 ['14:00'],
 ['13:30'],
 ['13:00'],
 ['12:30'],
 ['12:00'],
 ['11:30'],
 ['11:00'],
 ['10:30'],
 ['10:00'],
 ['09:30'],
 ['09:00'],
```

In [193]:
```
Temp = []

for i in soup.find_all('td',class_="t Temp"):
    Temp.append(i.text)

# Weather details for last 24 hour - Temperature :-
Temp
```

Out[193]:
```
['25°C',
 '25°C',
 '26°C',
 '26°C',
 '26°C',
 '27°C',
 '27°C',
 '27°C',
 '27°C',
 '27°C',
 '27°C',
 '27°C',
 '26°C',
 '26°C',
 '25°C',
 '23°C',
 '23°C',
 '22°C',
 '22°C',
 '21°C',
```

In [192]:
```
wind = []

for i in soup.find_all('td',class_="wind"):
    wind.append(i.text)

# Weather details for last 24 hour - Wind :-
wind
```

Out[192]:
```
['6 km/h',
 '6 km/h',
 '9 km/h',
 '11 km/h',
 '11 km/h',
 '13 km/h',
 '13 km/h',
 '15 km/h',
 '15 km/h',
 '15 km/h',
 '15 km/h',
 '19 km/h',
 '13 km/h',
 '13 km/h',
 '11 km/h',
 '11 km/h',
 '11 km/h',
 '9 km/h',
 '9 km/h',
```

In [191]:
```
Weather_condition = []

for i in soup.find_all('span',class_="thhip ico i0530 u303"):
    Weather_condition.append(i.text)

# Weather details for last 24 hour - Weather Condition :-
Weather_condition
```

Out[191]:
```
['Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
 'Mist',
```

In [190]:
```
Humidity = []

for i in soup.find_all('td',class_="hr"):
    Humidity.append(i.text)

# Weather details for last 24 hour - Humidity :-
Humidity
```

Out[190]:
```
['47%',
 '44%',
 '42%',
 '42%',
 '42%',
 '39%',
 '42%',
 '45%',
 '45%',
 '48%',
 '51%',
 '54%',
 '54%',
 '57%',
 '65%',
 '69%',
 '83%',
 '83%',
 '94%',
```

In [189]:
```
Pressure = []

for i in soup.find_all('td',class_="prob"):
    Pressure.append(i.text)

# Weather details for last 24 hour - Pressure :-
Pressure
```

Out[189]:
```
['1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1013 hPa',
 '1014 hPa',
 '1014 hPa',
 '1015 hPa',
 '1015 hPa',
 '1016 hPa',
 '1016 hPa',
 '1016 hPa',
 '1016 hPa',
 '1016 hPa',
 '1016 hPa',
 '1016 hPa',
```

In [ ]:
```

```

In [ ]:
```

```

---- -:- ----

Question no.10) Write a python program to scrape monument name, monument description, image url about top 10 monuments from 'https://www.puredestinations.co.uk/top-10-famous-monuments-to-visit-in-india/' (https://www.puredestinations.co.uk/top-10-famous-monuments-to-visit-in-india/')

In [196]:
```
page = requests.get('https://www.puredestinations.co.uk/top-10-famous-monume
page
```

Out[196]:
```
<Response [200]>
```

In [197]:
```
soup = BeautifulSoup(page.content)
soup
```

Out[197]:
```
<!DOCTYPE html>
<!--[if IE 7]>
<html class="ie ie7" lang="en">
<![endif]--><!--[if IE 8]>
<html class="ie ie8" lang="en">
<![endif]--><!--[if IE 9]>
<html class="ie ie9" lang="en">
<![endif]--><!--[if !(IE 7) | !(IE 8) | !(IE 9) ]><!--><html data-useragent
="Mozilla/5.0 (compatible; MSIE 10.0; Windows NT 6.2; Trident/6.0)" lang="e
n">
<!--<![endif]-->
<head>
<title>Top 10 Famous Monuments to Visit In India | Pure Destinations</title>
<meta content="width=device-width, initial-scale=1.0, maximum-scale=1.0, user
-scalable=no" name="viewport"/>
<meta charset="utf-8"/>
<link href="http://gmpg.org/xfn/11" rel="profile"/>
<link href="https://www.puredestinations.co.uk/xmlrpc.php" rel="pingback"/>
<!-- FAVICONS -->
```

In [198]:
```
names = []
for i in soup.find_all('p',class_=""):
    names.append(i.text.split(','))
names


# Names & Descriptions of Top 10 monuments :-
Details = names[4:33]
Details
```

Out[198]:
```
[['Taj Mahal', ' Agra'],
 ['Enlisted in the Seven Wonders of the World',
  ' The Taj Mahal is one of the most beautiful and famous buildings located in the city of Agra. This white marble monument was built by a Mughal Emperor called Shahajahan in memory of his beloved wife. Due to its amazing architecture and the history behind it',
  ' this world heritage site has become very popular to visit by all travellers and romantics from all over the world.'],
 [''],
 ['Golden Temple (Harmandir Sahib)', ' Amritsar '],
 ['The holiest shrine and pilgrimage place located in Amritsar is The Golden Temple known as the Harmandir Sahib. This is the most famous and sacred Sikh Gurdwara in Punjab',
  ' India',
  ' adorned with rich history and gold gilded exterior. If you are interested in culture and history',
  ' be sure to visit this popular attraction in India.'],
 [''],
 ['Meenakshi Temple', ' Madurai'],
 ['Meenakshi Temple is situated on the Southern banks of Vaigai River in the temple city Madurai. This temple is dedicated to Parvati and her consort',
  ' Shiva and is visited by most Hindu and Tamil devotees and architectural lovers throughout the world. It is believed that this shrine houses 33',
  '000 sculptures in its 14 gopurams. It’s no doubt one place to visit if you are impressed with art and cultural history.'],
 [''],
 ['Mysore Palace', ' Mysore'],
 ['The Mysore Palace is a famous historical monument in the city of Mysore in Karnataka. Commonly described as the City of Palaces',
  ' this is the most famous tourist attraction in India after the Taj Mahal. It is a sight not be missed with its spacious halls',
  ' lovely art paintings and Indo-Saracenic style architecture. Best time to visit is at night due to the astonishing illuminated lights covering the whole monument.'],
 [''],
 ['Gateway of India', ' Mumbai'],
 ['Even though Mumbai is famous for its Bollywood actors and movies',
  ' the most famous attraction in Mumbai is The Gateway of India. It is a popular gathering spot for locals',
  ' travellers',
  ' street vendors and photographers and is known as the Taj Mahal of Mumbai. The majestic monument was built to commemorate the visit of King George V and Queen Mary to Bombay. With so much fun and excitement this place is not to be missed with family or on your tour of India.'],
 [''],
 ['Red Fort', ' New Delhi'],
```

```
 ['Declared as the UNESCO's World Heritage Site',
  ' Red Fort is located in the centre of beautiful Delhi. If you love learning
about history and culture then this famous historic monument is a must see plac
e to visit. Built by the Mughal Emperor',
  ' Shah Jahan in 1648',
  ' and housing a number of museums',
  ' its walls are built of red sandstone. The best time to visit is on Independ
ence Day where the Prime Minister of India hoists the national flag at the Red
Fort. End the day by heading to an Indian restaurant and enjoy the varieties of
wonderful cuisines.'],
 [''],
 ['Hawa Mahal', ' Jaipur'],
 ['Explore a blend of beauty and Rajasthan culture',
  ' the Hawa Mahal also known as Palace of Winds is situated in the capital of
Rajasthan',
  ' Jaipur. Built from red and pink sandstones by the Maharaja Sawi Pratap Sing
h in 1799',
  ' this unique five storey structure is one of the most prominent tourist attr
actions in the Jaipur city.'],
 [''],
 ['Qutub Minar', ' New Delhi'],
 ['Discover one of the tallest towers in the world and the second tallest Minar
of India standing elegantly in the Capital city',
  ' New Delhi. Standing at 72.5 metres and consisting of around 379 stairs',
  ' this famous monument represents the rich architecture of India. As it is a
UNESCO World Heritage Site made of red sandstone and decorated with Arabic and
Brahmi inscriptions',
  ' travellers from around the world come to view this most famous tower in Ind
ia.'],
 [''],
 ['Sanchi Stupa', ' Sanchi'],
 ['The beautiful and massive dome',
  ' Sanchi Stupa also known as the Great Stupa is a world renowned Buddhist mon
ument in Sanchi',
  ' India. It was constructed by Emperor Ashoka',
  ' and is one of the oldest stone structures in the heart of India. Experience
the Indian culture by visiting these major attractions in Sanchi including a nu
mber of Buddhist Stupas',
  ' monasteries and temples.'],
 [''],
 ['Charminar', ' Hyderabad'],
 ['No visit to Hyderabad should be complete without visiting the most famous an
d majestic monument known as the Charminar. This magnificent and striking mosqu
e constructed in 1591 has four minarets and is the most recognisable symbol in
the city of Hyderabad.']]
```

In [199]:
```
img_url = []
for i in soup.find_all('img',class_=""):
    img_url.append(i)

# Image URL for the Top 10 monuments :-
img_url
```

Out[199]:
```
[<img height="1" src="https://www.facebook.com/tr?id=120737145081077&amp;ev=Pag
eView
 &amp;noscript=1" width="1"/>,
 <img alt="ABTA Travel with confidence" height="41" src="https://www.puredestin
ations.co.uk/wp-content/themes/puredestinations/images/accreditations/abta.png"
width="107"/>,
 <img alt="Air Travel Organisers' Licensing" height="41" src="https://www.pured
estinations.co.uk/wp-content/themes/puredestinations/images/accreditations/prot
ected.png" width="41"/>,
 <img height="1" src="https://www.facebook.com/tr?id=1063876163715 70&amp;ev=Pag
eView&amp;noscript=1" style="display:none" width="1"/>]
```

In [ ]:
```

```

In [ ]:
```

```

---- -:- ----