In [1]:

```
pip install bs4
```

```
Requirement already satisfied: bs4 in c:\users\admin\anaconda3\lib\site-pack
ages (0.0.1)
Requirement already satisfied: beautifulsoup4 in c:\users\admin\anaconda3\li
b\site-packages (from bs4) (4.11.1)
Requirement already satisfied: soupsieve>1.2 in c:\users\admin\anaconda3\lib
\site-packages (from beautifulsoup4->bs4) (2.3.1)
Note: you may need to restart the kernel to use updated packages.
```

In [2]:

```
from bs4 import BeautifulSoup
```

In [3]:

```
import requests
```

In [4]:

```
page=requests.get("https://www.imdb.com/list/ls092711106/")
```

In [5]:

```
page
```

Out[5]:

```
<Response [200]>
```

In [6]:

```
page.content
```

...

In [7]:

```
soup=BeautifulSoup(page.content)
soup
```

...

In [8]:

```
print(soup.prettify)
```

...

In [9]:

```
first_webseries=soup.find('h3',class_="lister-item-header")
first_webseries
```

Out[9]:

```
<h3 class="lister-item-header">
<span class="lister-item-index unbold text-primary">1.</span>
<a href="/title/tt0903747/">Breaking Bad</a>
<span class="lister-item-year text-muted unbold">(2008–2013)</span>
</h3>
```

In [10]:

```
first_webseries.text
```

Out[10]:

```
'\n1.\nBreaking Bad\n(2008–2013)\n'
```

In [11]:

```
names=soup.find_all('h3',class_="lister-item-header")
names
```

. . .

In [12]:

```
namelist1=[]
for i in names:
    for j in i.find_all('a'):
        namelist1.append(j.text)
namelist1
```

Out[12]:

```
['Breaking Bad',
 'Game of Thrones',
 'Mirzapur',
 'The Boys',
 'Daredevil',
 'The Walking Dead',
 'Sacred Games',
 'Afsos',
 'Breathe',
 'The Family Man',
 'Chernobyl',
 'Dexter',
 'Bates Motel',
 'Asur: Welcome to Your Dark Side',
 'Fosse/Verdon',
 'Euphoria',
 'Killing Eve',
 'Succession',
 'Kingdom',
 'Six Feet Under',
 'Dark Angel',
 'Doom Patrol',
 'Swamp Thing',
 'Chilling Adventures of Sabrina',
 'The Terror',
 'Channel Zero',
 'Hannibal',
 'Dark',
 'The Haunting of Hill House',
 'The Originals',
 'True Blood',
 'Fargo',
 "A Young Doctor's Notebook & Other Stories",
 'Fearless',
 'You',
 'The Vampire Diaries',
 'I Am Not Okay with This',
 'Love, Death & Robots',
 'Lucifer',
 'The Outsider',
 'Sense8',
 'Delhi Crime',
 'Arrested Development',
 'Wu Assassins',
 "Schitt's Creek",
 'Barry',
 'Ninjak vs the Valiant Universe',
 'From Dusk Till Dawn: The Series',
 'The Hollow Crown',
 'American Horror Story',
 'Mad About You',
```

```
 'Murder One',
 'Rectify',
 'The Killing',
 'Black Summer',
 'Locke & Key',
 'The Man in the High Castle',
 'Carnival Row',
 'The Expanse',
 'Titans',
 'The Night Manager',
 'Billions',
 'Unbelievable',
 'Unorthodox',
 'The Morning Show',
 'I Know This Much Is True',
 'Big Little Lies',
 'Fear City: New York vs the Mafia',
 'The Blacklist',
 'Your Honor',
 'Superman & Lois',
 'Warrior',
 'Safe',
 'The Five',
 'The Sinner',
 'The Fall',
 'Broadchurch',
 'The Undoing']
```

In [13]:

```
rank=soup.find_all('span',class_="lister-item-index unbold text-primary")
rank
```

...

In [14]:

```
ranks=[]
for i in rank:
    ranks.append(i.text)
ranks
```

...

In [15]:

```
year=soup.find_all('span',class_="lister-item-year text-muted unbold")
year
```

...

In [16]:

```
years=[]
for i in year:
    years.append(i.text)
years
```

Out[16]:

```
['(2008–2013)',
 '(2011–2019)',
 '(2018– )',
 '(2019– )',
 '(2015–2018)',
 '(2010–2022)',
 '(2018– )',
 '(2020– )',
 '(2018)',
 '(2019– )',
 '(2019)',
 '(2006–2013)',
 '(2013–2017)',
 '(2020– )',
 '(2019)',
 '(2019– )',
 '(2018–2022)',
 '(2018– )',
```

In [17]:

```
print(ranks,years,namelist1)
```

```
['1.', '2.', '3.', '4.', '5.', '6.', '7.', '8.', '9.', '10.', '11.', '1
2.', '13.', '14.', '15.', '16.', '17.', '18.', '19.', '20.', '21.', '22.',
'23.', '24.', '25.', '26.', '27.', '28.', '29.', '30.', '31.', '32.', '3
3.', '34.', '35.', '36.', '37.', '38.', '39.', '40.', '41.', '42.', '43.',
'44.', '45.', '46.', '47.', '48.', '49.', '50.', '51.', '52.', '53.', '5
4.', '55.', '56.', '57.', '58.', '59.', '60.', '61.', '62.', '63.', '64.',
'65.', '66.', '67.', '68.', '69.', '70.', '71.', '72.', '73.', '74.', '7
5.', '76.', '77.', '78.'] ['(2008–2013)', '(2011–2019)', '(2018– )', '(201
9– )', '(2015–2018)', '(2010–2022)', '(2018– )', '(2020– )', '(2018)', '(2
019– )', '(2019)', '(2006–2013)', '(2013–2017)', '(2020– )', '(2019)', '(2
019– )', '(2018–2022)', '(2018– )', '(2019– )', '(2001–2005)', '(2000–200
2)', '(2019– )', '(2019)', '(2018–2020)', '(2018–2019)', '(2016–2018)',
'(2013–2015)', '(2017–2020)', '(2018)', '(2013–2018)', '(2008–2014)', '(20
14– )', '(2012–2013)', '(I) (2017)', '(2018– )', '(2009–2017)', '(2020)',
'(2019– )', '(2016–2021)', '(2020)', '(2015–2018)', '(2019– )', '(2003–201
9)', '(2019)', '(2015–2020)', '(2018– )', '(2018– )', '(2014–2016)', '(201
2–2016)', '(2011– )', '(1992–2019)', '(1995–1997)', '(2013–2016)', '(2011–
2014)', '(2019– )', '(2020–2022)', '(2015–2019)', '(2019– )', '(2015–202
2)', '(I) (2018– )', '(2016)', '(2016– )', '(2019)', '(2020)', '(I) (2019–
```

In [18]:

```
import pandas as pd
web_series=pd.DataFrame({})
web_series['ranks']=ranks
web_series['Name of Series']=namelist1
web_series['year']=years
```

In [19]:

```
web_series
```

Out[19]:

	ranks	Name of Series	year
0	1.	Breaking Bad	(2008–2013)
1	2.	Game of Thrones	(2011–2019)
2	3.	Mirzapur	(2018–)
3	4.	The Boys	(2019–)
4	5.	Daredevil	(2015–2018)
...	...	...	...
73	74.	The Five	(2016)
74	75.	The Sinner	(2017–2021)
75	76.	The Fall	(I) (2013–2016)
76	77.	Broadchurch	(2013–2017)
77	78.	The Undoing	(2020)

78 rows × 3 columns

In [ ]:

```

```