In [ ]:

In [3]:
```
# Your code goes here and below
# First 50 movies
import pandas as pd
import csv
import requests
from bs4 import BeautifulSoup

ranking=[]
movie_names = []
ratings = []

# Set the URL to scrape first 50
url = "https://myanimelist.net/topanime.php"

page = requests.get(url)
print(page)
# Parse the HTML content of the page
soup = BeautifulSoup(page.text, 'html.parser')
#print(soup)
# Find all the <div> elements with the class "lister-item mode-advanced"

table = soup.find("table", class_="top-ranking-table")

for row in table.find_all("tr", class_="ranking-list"):
            rank = row.find("td", class_="rank").text.strip()
            title = row.find("div", class_="detail").find("h3").text.strip()
            score = row.find("td", class_="score").find("span").text.strip()
            ranking.append(rank)
            movie_names.append(title)
            ratings.append(score)


            # Append the data to the rankings list


movie_dic = {"Movie Name" : movie_names, "Rating" : ratings}
df = pd.DataFrame(movie_dic)
df
```

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

Out[3]:

	Movie Name	Rating
0	Sousou no Frieren	9.37
1	Fullmetal Alchemist: Brotherhood	9.09
2	Steins;Gate	9.07
3	Gintama°	9.06
4	Shingeki no Kyojin Season 3 Part 2	9.05
5	Gintama: The Final	9.04
6	Hunter x Hunter (2011)	9.04
7	Gintama'	9.03
8	Bleach: Sennen Kessen-hen	9.02
9	Ginga Eiyuu Densetsu	9.02
10	Gintama': Enchousen	9.02
11	Kaguya-sama wa Kokurasetai: Ultra Romantic	9.01
12	Fruits Basket: The Final	8.98
13	Gintama.	8.98
14	Gintama	8.94
15	Clannad: After Story	8.93
16	Koe no Katachi	8.93
17	3-gatsu no Lion 2nd Season	8.92
18	Code Geass: Hangyaku no Lelouch R2	8.91
19	Gintama Movie 2: Kanketsu-hen - Yorozuya yo Ei...	8.91
20	Kusuriya no Hitorigoto	8.91
21	Shingeki no Kyojin: The Final Season - Kankets...	8.90
22	Gintama.: Shirogane no Tamashii-hen - Kouhan-sen	8.88
23	Monster	8.88
24	Owarimonogatari 2nd Season	8.87
25	Violet Evergarden Movie	8.87
26	Boku no Kokoro no Yabai Yatsu 2nd Season	8.87
27	Kimi no Na wa.	8.84
28	Jujutsu Kaisen 2nd Season	8.83
29	Kingdom 3rd Season	8.82
30	Gintama.: Shirogane no Tamashii-hen	8.81
31	Vinland Saga Season 2	8.81
32	The First Slam Dunk	8.80
33	Shingeki no Kyojin: The Final Season	8.79

	Movie Name	Rating
34	Mob Psycho 100 II	8.79
35	Bocchi the Rock!	8.79
36	Haikyuu!! Karasuno Koukou vs. Shiratorizawa Ga...	8.78
37	Kaguya-sama wa Kokurasetai: First Kiss wa Owar...	8.78
38	Kizumonogatari III: Reiketsu-hen	8.78
39	Sen to Chihiro no Kamikakushi	8.77
40	Hajime no Ippo	8.77
41	Shingeki no Kyojin: The Final Season Part 2	8.76
42	Monogatari Series: Second Season	8.76
43	Kimetsu no Yaiba: Yuukaku-hen	8.75
44	Kingdom 5th Season	8.75
45	Ashita no Joe 2	8.75
46	Cowboy Bebop	8.75
47	Vinland Saga	8.75
48	Kingdom 4th Season	8.74
49	Mushishi Zoku Shou 2nd Season	8.73

In [6]:
```
# All 100 movies
import requests
from bs4 import BeautifulSoup

movie_names = []
ratings = []

# Set the URL to scrape first 50
url_1_50 = "https://myanimelist.net/topanime.php"

# from 51 to 100
url_51_100 = "https://myanimelist.net/topanime.php?limit=50"

page = requests.get(url_1_50)

# Parse the HTML content of the page
soup = BeautifulSoup(page.text, 'html.parser')

# Find all the <div> elements with the class "lister-item mode-advanced"
table = soup.find("table", class_="top-ranking-table")

for row in table.find_all("tr", class_="ranking-list"):
            rank = row.find("td", class_="rank").text.strip()
            title = row.find("div", class_="detail").find("h3").text.strip()
            score = row.find("td", class_="score").find("span").text.strip()
            ranking.append(rank)
            movie_names.append(title)
            ratings.append(score)
            page = requests.get(url_51_100)
```

```
# Parse the HTML content of the page
soup = BeautifulSoup(page.text, 'html.parser')

# Find all the <div> elements with the class "lister-item mode-advanced"
table = soup.find("table", class_="top-ranking-table")

for row in table.find_all("tr", class_="ranking-list"):
            rank = row.find("td", class_="rank").text.strip()
            title = row.find("div", class_="detail").find("h3").text.strip()
            score = row.find("td", class_="score").find("span").text.strip()
            ranking.append(rank)
            movie_names.append(title)
            ratings.append(score)

movie_dic = {"Movie Name" : movie_names, "Rating" : ratings}
df = pd.DataFrame(movie_dic)
df
```

Out[6]:

	Movie Name	Rating
0	Sousou no Frieren	9.37
1	Fullmetal Alchemist: Brotherhood	9.09
2	Steins;Gate	9.07
3	Gintama°	9.06
4	Shingeki no Kyojin Season 3 Part 2	9.05
...	...	...
95	Kizumonogatari II: Nekketsu-hen	8.57
96	Natsume Yuujinchou Go	8.57
97	Natsume Yuujinchou San	8.57
98	Ookami Kodomo no Ame to Yuki	8.57
99	Shouwa Genroku Rakugo Shinjuu	8.57

100 rows × 2 columns

In [4]:

```
# Determine the number of movies by a variable
# Not all values can be used
import requests
from bs4 import BeautifulSoup

movie_names = []
ratings = []
how_many=1000

# Set the URL to scrape first 50
first_url = "https://myanimelist.net/topanime.php?limit="
page_no = 1
next_page = "0"

while page_no < how_many:
    current_page = first_url + next_page
```

```
    print(current_page)
    page = requests.get(current_page)

    # Parse the HTML content of the page
    soup = BeautifulSoup(page.text, 'html.parser')

    # Find all the <div> elements with the class "lister-item mode-advanced"
    table = soup.find("table", class_="top-ranking-table")

    for row in table.find_all("tr", class_="ranking-list"):
            rank = row.find("td", class_="rank").text.strip()
            title = row.find("div", class_="detail").find("h3").text.strip()
            score = row.find("td", class_="score").find("span").text.strip()
            ranking.append(rank)
            movie_names.append(title)
            ratings.append(score)


    page_no += 50
    next_page = str(page_no)


movie_dic = {"Movie Name" : movie_names, "Rating" : ratings}
df = pd.DataFrame(movie_dic)
df
```

```
https://myanimelist.net/topanime.php?limit=0
https://myanimelist.net/topanime.php?limit=51
https://myanimelist.net/topanime.php?limit=101
https://myanimelist.net/topanime.php?limit=151
https://myanimelist.net/topanime.php?limit=201
https://myanimelist.net/topanime.php?limit=251
https://myanimelist.net/topanime.php?limit=301
https://myanimelist.net/topanime.php?limit=351
https://myanimelist.net/topanime.php?limit=401
https://myanimelist.net/topanime.php?limit=451
https://myanimelist.net/topanime.php?limit=501
https://myanimelist.net/topanime.php?limit=551
https://myanimelist.net/topanime.php?limit=601
https://myanimelist.net/topanime.php?limit=651
https://myanimelist.net/topanime.php?limit=701
https://myanimelist.net/topanime.php?limit=751
https://myanimelist.net/topanime.php?limit=801
https://myanimelist.net/topanime.php?limit=851
https://myanimelist.net/topanime.php?limit=901
https://myanimelist.net/topanime.php?limit=951
```

Out[4]:

	Movie Name	Rating
0	Sousou no Frieren	9.37
1	Fullmetal Alchemist: Brotherhood	9.09
2	Steins;Gate	9.07
3	Gintama°	9.06
4	Shingeki no Kyojin Season 3 Part 2	9.05
...	...	...
995	Aishang Ta de Liyou Extra	7.81
996	Baby Steps	7.81
997	Bungou Stray Dogs	7.81
998	Enen no Shouboutai: Ni no Shou	7.81
999	Hidamari Sketch x SP	7.81

1000 rows × 2 columns

In [7]:
```
# Save the movies and ratings in a csv file
df.to_csv('saidinesh_myanimelist_1000.csv')
```

In [ ]:
```
```