In [4]:

```
import requests
import lxml.html as html
import pandas as pd
import numpy as np
from collections import defaultdict
```

CREANDO FUNCIÓN PARA OBTENER LOS DATOS DE LA PAGINA A SCRAPEAR

- **Titulos**
- **Score**
- **Temporada**
- **Portada**

In [1]:

```
def parser_anime_temporada(url):
    content_dic = {}
    r = requests.get(url)
    home = r.content.decode("utf-8")
    parser = html.fromstring(home)

    if r.status_code == 200:

        #OBTENIENDO TITULOS

        titulo_temporada = '//div[2]/div[3]/div[1]//a[@class="link-title"]/text()'
        titulos_anime_streno_temporada = parser.xpath(titulo_temporada)
        content_dic['Anime Season'] = titulos_anime_streno_temporada

        #OBTENIENDO SCORE

        score_temporada = '//div[2]/div[3]/div[1]//div[@title="Score"]/text()'
        score_anime_streno_temporada = parser.xpath(score_temporada)
        score_anime_streno_temporada = [x for x in score_anime_streno_temporada if x !=
'\n            ']
        score_anime_streno_temporada = list(map(lambda x: x.replace('\n        ', ''), s
core_anime_streno_temporada))
        content_dic['Score Anime Season'] = score_anime_streno_temporada

        #OBTENIENDO LA TEMPORADA ACTUAL

        temporada = '//h1//a[@class="on"]/text()'
        temporada_analisis = parser.xpath(temporada)
        temporada_analisis = list(map(lambda x: x.replace('\n', ''), temporada_analisis)
)
        temporada_analisis = [x.strip() for x in temporada_analisis]

    #OBTENIENDO LINK DE PORTADAS
    imagenes_temporada = []
    # tree = html.fromstring(home)
    for link_element in parser.xpath('//div[2]/div[3]/div[1]//div[@class="image"]//a/img'
):
        # href1 = link_element.get('src')
        # print(type(href1))
        if (link_element.get('src') is None):
            href1 = link_element.get('data-src')
            imagenes_temporada.append(href1)
            # print(href1)
        else:
            href1 = link_element.get('src')
            imagenes_temporada.append(href1)
        # print(href1)

    #COLOCANDO URL DE IMAGEN EN DICCIONARIO
    content_dic['Imagenes'] = []
```

```
    content_dic.update({"Imagenes": imagenes_temporada})

    #COLOCANDO TEMPORADA EN DICCIONARIO
    list_temporada = []
    for j in range(len(content_dic['Anime Season'])):

        list_temporada.append(temporada_analisis)

    content_dic['Temporada'] = []
    content_dic.update({"Temporada": list_temporada})

    return content_dic
```

Definimos web a scrapear y ejecutamos la función

In [6]:

```
#WEB A SCRAPEAR
url_padre = "https://myanimelist.net/anime/season"

#CREANDO DICCIONARIO Y AGREGANDO INFORMACIÓN
data = []
data.append(parser_anime_temporada(url_padre))
```

CREANDO DATAFRAME

In [9]:

```
df_anime_list = pd.DataFrame()
for j in data:
    df_uno = pd.DataFrame(j)
    df_anime_list = pd.concat([df_anime_list, df_uno])

df_anime_list.head()
```

Out[9]:

	Anime Season	Score Anime Season	Imagenes	Temporada
0	Spy x Family	9.08	https://cdn.myanimelist.net/images/anime/1441/...	[Spring 2022]
1	Tate no Yuusha no Nariagari Season 2	7.33	https://cdn.myanimelist.net/images/anime/1143/...	[Spring 2022]
2	Kaguya-sama wa Kokurasetai: Ultra Romantic	8.96	https://cdn.myanimelist.net/images/anime/1160/...	[Spring 2022]
3	Kawaii dake ja Nai Shikimori-san	7.23	https://cdn.myanimelist.net/images/anime/1995/...	[Spring 2022]
4	Komi-san wa, Comyushou desu. 2nd Season	8.32	https://cdn.myanimelist.net/images/anime/1108/...	[Spring 2022]

CREANDO FUNCIÓN PARA OBTENER IMG EN DATAFRAME (FORMATEAR A HTML)

In [10]:

```
from IPython.core.display import HTML

def path_html_img(url):
    return '<img src="' + url + '"width = "60">'

df_anime_list['Imagen_vista'] = df_anime_list['Imagenes'].apply(lambda x : path_html_img(
x))
df_anime_list.head()
```

Out[10]:

	Score

	Anime Season	Score Anime Season	Imagenes	Temporada	Imagen_v
0	Spy x Family	9.08	https://cdn.myanimelist.net/images/anime/1441/...	[Spring 2022]	<src="https://cdn.myanimelist.net/images
1	Tate no Yuusha no Nariagari Season 2	7.33	https://cdn.myanimelist.net/images/anime/1143/...	[Spring 2022]	<src="https://cdn.myanimelist.net/images
2	Kaguya-sama wa Kokurasetai: Ultra Romantic	8.96	https://cdn.myanimelist.net/images/anime/1160/...	[Spring 2022]	<src="https://cdn.myanimelist.net/images
3	Kawaii dake ja Nai Shikimori-san	7.23	https://cdn.myanimelist.net/images/anime/1995/...	[Spring 2022]	<src="https://cdn.myanimelist.net/images
4	Komi-san wa, Comyushou desu. 2nd Season	8.32	https://cdn.myanimelist.net/images/anime/1108/...	[Spring 2022]	<src="https://cdn.myanimelist.net/images

Mostramos Dataframe en formato HTML

In [12]:

```
HTML(df_anime_list.to_html(escape=False, formatters=dict(Portada = path_html_img)))
```

Out[12]:

	Anime Season	Score Anime Season	Imagenes	Temporada	Imagen_vista
0	Spy x Family	9.08	https://cdn.myanimelist.net/images/anime/1441/122795.jpg	[Spring 2022]	
1	Tate no Yuusha no Nariagari Season 2	7.33	https://cdn.myanimelist.net/images/anime/1143/121873.jpg	[Spring 2022]	
2	Kaguya-sama wa Kokurasetai: Ultra Romantic	8.96	https://cdn.myanimelist.net/images/anime/1160/122627.jpg	[Spring 2022]	
3	Kawaii dake ja Nai Shikimori-san	7.23	https://cdn.myanimelist.net/images/anime/1995/121695.jpg	[Spring 2022]	
4	Komi-san wa, Comyushou desu. 2nd Season	8.32	https://cdn.myanimelist.net/images/anime/1108/121157.jpg	[Spring 2022]	
5	Aharen-san wa Hakarenai	7.49	https://cdn.myanimelist.net/images/anime/1612/120636.jpg	[Spring 2022]