Grupo GII3

Integrantes:

Galeano, Gerardo Agustín

Guzmán Ma. Lilen

Ingaramo Ma. Eugenia

Proyecto colaborativo de Web Scraping

El proyecto de web scraping extrae datos del resultado generado por un programa codificado en la plataforma de VS code, y subido al repositorio en la plataforma de GitHub.

Esta codificado con python y toma datos de la plataforma https://www.filmaffinity.com/ar/cat_new_netflix.html para almacenarlos en un archivo con extensión ".csv" llamado "webScrapingNetflix.csv"

Desarrollo del código:

Primeramente, se usaron las librerías BeautifulSoup, lxml, Requests y pandas. El programa consta de una primera función que consigue el link de todas las páginas de la web filmaffinity:

```
def linkTodasPginas(pageLink):
    lista=[]
    for i in range(1, 2):#gene
      b=str(i)#paso int a stri
      lista.append(pageLink+b)
    return lista
```

De cada página se obtienen los links de las películas y se los agrega a la lista "listaPeliculas" agregando en esta lista el link de todas las películas y descargando el contenido de la página actual en el momento de ejecución de la línea y se guarda todo en un archivo".lxml" (links).

```
def linkTodasPeliculas(listaPaginas):#de cada pagina obtengo los
    listaPeliculas=[]# en esta lista agrego el link de todas las
    for i in listaPaginas:#de cada pagina obtengo los link
        page = requests.get(i)  # descargo el contenido de la pagi
        soup = BeautifulSoup(page.text, 'lxml')  # genero el texto
        listaLink = soup.find_all('div', class_="movie-title")  #
        linkPeliculas=[x.find('a').get('href')for x in listaLink]

        contador=0
        for j in linkPeliculas:#de la lista obtenida al recorrer
                               #que tiene el link de todas las pe
            listaPeliculas.append(linkPeliculas[contador])
            contador=contador+1
    return listaPeliculas#retorno una lista con los link de pelicu
```

Seguidamente se genera un método que los guarda en la misma lista, se obtienen los datos que se desean como título, genero, etc. Luego se descargan los datos guardándolos en un archivo ".lxml"

Además de guardarlos en los archivos "csv", estos se muestran en la pantalla de la consola

```
[Running] python -u "c:\Users\mingaramo3\OneDrive - DXC Production\EUGE\TRAINING\ISPC - CDIA\MODULO
PROGRAMADOR\_PROYECTO INTEGRADOR\Proyecto Integrador\GitHub_GII3\GII3\grupo GII3 webScrapingNetflix
py"
['', 'Titulo', 'Anio', 'Pais', 'Director', 'Genero']
['0', 'Jurassic World Camp Cretaceous: Hidden Adventure', '2022', 'Estados Unidos', 'Sin Datos',
'Animación']
['1', 'Teletubbies', '1997', 'Reino Unido', 'Andrew Davenport', 'Serie de TV']
['2', 'Stutz', '2022', 'Estados Unidos', 'Jonah Hill', 'Documental']
['3', "My Father's Dragon", '2022', 'Irlanda', 'Nora Twomey', 'Animación']
['4', 'Down to Earth with Zac Efron', '2020', 'Estados Unidos', 'Zac Efron', 'Serie de TV']
['5', 'Is That Black Enough for You?!?', '2022', 'Estados Unidos', 'Elvis Mitchell', 'Documental']
```

```
for i in linkPeliculas:
    lista = list()
    page = requests.get(i)  # descargo el c
    soup = BeautifulSoup(page.text, 'lxml')
    box=soup.find('dl',class_='movie-info')

    lista.append(obtenerTitulos(box))
    lista.append(obtenerAnio(box))
    lista.append(obtenerPais(box))
    lista.append(obtenerDirector(box))
    lista.append(obtenerGenero(box))
    listaTodos.append(lista)
```

Desarrollando todas las funciones se completaría el código completo, se omite ya que es extenso y se deja que la magia suceda en el archivo mismo.

Para elaborar el informe realizamos "preguntas" a los datos y visualizamos las respuestas gráficamente mediante la herramienta Google Data Studio. Lo que hacemos con el web scrapping es generar ese set de datos a partir de la información contenida en la página que elegimos. Como no tenemos tanto poder de cómputo para recorrer todas las páginas de la URL decidimos trabajar con un set de datos de kaggle que tiene más registros.

La gestión se llevó a cabo con la herramienta Trello; aquí el link: https://trello.com/invite/b/7zxXt6Oj/ATTI5a39ff6ae569c4ee71fc716fe89759bbF793CCBD/gii3-kanban .