Utilización de WebDriver para extraer HTML de web con javascript.

In [1]:
```
from selenium import webdriver
from bs4 import BeautifulSoup
from selenium.webdriver.support.ui import Select
from datetime import datetime
from dateutil.relativedelta import relativedelta
import requests
import zipfile
import os
```

In [ ]:
```
%%time
url="https://datos.sfp.gov.py/data/funcionarios/download"

driver = webdriver.Chrome()
driver.get(url)
```

In [3]:
```
%%time
cantidad_registros = 36
select_element = Select(driver.find_element('name','listado_length'))
select_element.select_by_visible_text(f"{cantidad_registros}")

driver.implicitly_wait(10)
```

```
Wall time: 165 ms
```

WebScrapping de html - Automatización de descarga de excels 1

In [4]:
```
%%time
html = driver.page_source
driver.close()

soup = BeautifulSoup(html)
```

```
Wall time: 227 ms
```

In [5]:
```
%%time
prefijo_archivos = '/data/funcionarios_'
prefijo_para_descarga = 'https://datos.sfp.gov.py'
sufijo_archivos = '.csv.zip'

enlaces = [prefijo_para_descarga + link['href']
           for link in soup.find_all('a', href=True)
           if link['href'].startswith(prefijo_archivos) and link['href'][-8:]==sufi

print(f'Cantidad de archivos {sufijo_archivos} ubicados en html: ' + str(len(enlace
```

```
Cantidad de archivos .csv.zip ubicados en html: 36
Wall time: 8 ms
```

In [6]:
```
%%time
fecha_mes_cerrado = datetime.now() - relativedelta(months=1)
fecha_mom = datetime.now() - relativedelta(months=2)
fecha_yoy = datetime.now() - relativedelta(months=13)

#Variables para renombre de archivos
anho_mes_act = str(fecha_mes_cerrado.year)+'_'+str(fecha_mes_cerrado.month)
anho_mes_mom = str(fecha_mom.year)+'_'+str(fecha_mom.month)
```

```
anho_mes_yoy = str(fecha_yoy.year)+'_'+str(fecha_yoy.month)

#Variables que determina cuales archivos (segun el anho_mes) se descargaran del lis
anho_mes = (str(fecha_mes_cerrado.year)+'_'+str(fecha_mes_cerrado.month),
            str(fecha_mom.year)+'_'+str(fecha_mom.month),
            str(fecha_yoy.year)+'_'+str(fecha_yoy.month))
anho_mes_a_descargar = tuple([anhomes + sufijo_archivos for anhomes in anho_mes])
print('Se descargarán los archivos de los meses:')
print(anho_mes)
```

```
Se descargarán los archivos de los meses:
('2023_6', '2023_5', '2022_6')
Wall time: 0 ns
```

In [7]:

```
%%time
ruta_descarga = 'C:\\Users\\HUAWEI\\Desktop\\Python_Projects\\WebScrapping_data\\Se

for enlace in enlaces:
    nombre_archivo = enlace.split('/')[-1]

    if nombre_archivo.endswith(anho_mes_a_descargar):
        respuesta = requests.get(enlace)

        if respuesta.status_code == 200:
            with open(ruta_descarga + nombre_archivo, 'wb') as archivo_local:
                archivo_local.write(respuesta.content)
            print(f"{nombre_archivo} descargado con éxito.")
        else:
            print(f"¡Error en descarga!!! Archivo: {nombre_archivo}. Código de esta
    #else:
        #print(f"No descargado: {nombre_archivo}.")
```

```
funcionarios_2023_6.csv.zip descargado con éxito.
funcionarios_2023_5.csv.zip descargado con éxito.
funcionarios_2022_6.csv.zip descargado con éxito.
Wall time: 13.3 s
```

Descomprimimos los ZIPs para mandarlos a SQL Server proximamente

In [8]:

```
%%time

ruta_destino = 'C:\\Users\\HUAWEI\\Desktop\\Python_Projects\\WebScrapping_data\\Sec
prefijo_archivo = 'funcionarios_'

for archivo_zip in os.listdir(ruta_destino):
    if archivo_zip.endswith('.zip'):
        with zipfile.ZipFile(os.path.join(ruta_destino, archivo_zip), 'r') as zip_r
            # Extraer todos los archivos del ZIP en una lista
            archivos_zip = zip_ref.namelist()

            # Buscar archivos con sufijo funcionarios_aaaa_mm.csv y renombrar si co
            for archivo_csv in archivos_zip:
                if archivo_csv.endswith('.csv'):
                    nombre_archivo_csv = os.path.basename(archivo_csv)
                    anho_mes_csv = "_".join((nombre_archivo_csv.split(".")[0]).spli

                    if anho_mes_csv == anho_mes_act:
                        nuevo_nombre = f'{prefijo_archivo}actual'
                    elif anho_mes_csv == anho_mes_mom:
                        nuevo_nombre = f'{prefijo_archivo}MoM'
                    elif anho_mes_csv == anho_mes_yoy:
                        nuevo_nombre = f'{prefijo_archivo}YoY'
```

```
                    else:
                        # Si no coincide con ninguna variable, no se renombra
                        continue

                    # Renombrar el archivo dentro del ZIP
                    zip_ref.extract(archivo_csv, ruta_destino)
                    os.rename(os.path.join(ruta_destino, archivo_csv), os.path.join
        os.remove(os.path.join(ruta_destino, archivo_zip))
```

```
Wall time: 6.64 s
```