

Comentario: Aquí se importan las bibliotecas necesarias y se establece la ruta de exportación para los archivos. Luego, se crea la carpeta si no existe.

```
In [1]: # Importación de bibliotecas necesarias
        csv
        json
        pandas    pd
        selenium   webdriver
        bs4        BeautifulSoup
        selenium.webdriver.common.by        By
        selenium.webdriver.support.ui        WebDriverWait
        selenium.webdriver.support        expected_conditions    EC
        os

# Ruta de exportación de archivos
ruta_exportacion    r'C:\resultados'

# Crear la carpeta si no existe
os.path.exists(ruta_exportacion):
os.makedirs(ruta_exportacion)
```

Esta función toma el código HTML de un trayecto y extrae la información de precios. Se asume que hay una estructura específica de clases en el HTML y por eso usamos BeautifulSoup.

```
In [2]: def extraer_precios(html_code):
        :
        soup    BeautifulSoup(html_code, 'html.parser')

        # Encuentra el elemento que contiene la información de precios
        precios_element    soup.find('div', class_='tu-clase-de-precios') # Reemplaza 'tu-clase-de-precios' con la c

        # Extrae los precios
        precios    []
        precios_element:
            precios_items    precios_element.find_all('span', class_='tu-clase-de-precio') # Reemplaza 'tu-clase-de-
            precios    [precio.get_text(strip=)    precio    precios_items]

        precios

        Exception    e:
        print(f"Error al extraer precios: {str(e)}")
        []
```

Esta función extrae las fechas de IDA y VUELTA de una URL codificada.

```
In [3]: def obtener_fecha_desde_url(url):
        :
        # Parsea la URL para encontrar las fechas
        partes_url    url.split('%2C')
        fecha_ida    partes_url[2]
        fecha_vuelta    partes_url[1]
        fecha_ida, fecha_vuelta
        Exception    e:
        print(f"Error al obtener fechas desde la URL: {str(e)}")
        ,
```

Esta función toma un elemento de trayecto, tipo de trayecto, precios, número de viaje y fechas, y devuelve un diccionario con la información del viaje.

```
In [4]: def obtener_info_viaje(trayecto_element, trayecto_tipo, precios, numero_viaje, fecha_ida, fecha_vuelta):
        :
        trayecto_html    trayecto_element.get_attribute("outerHTML")
        soup    BeautifulSoup(trayecto_html, 'html.parser')
```

```

hora_salida    soup.find('span', class_='sc-dLMFU gVWdyu JourneyTime__HourFont-sc-1ej0v4k-1 cBmPEj') get_text
ciudad_origen  soup.find('span', class_='sc-dLMFU gVWdyu JourneyTime__CityFont-sc-1ej0v4k-2 jpapGu') get_te

llegada_info_element  soup.find('div', class_='JourneyTime__TimeContainer-sc-1ej0v4k-0 bcTRis')
hora_llegada  llegada_info_element.find('span', class_='sc-dLMFU gVWdyu JourneyTime__HourFont-sc-1ej0v4k-1
ciudad_destino  llegada_info_element.find('span', class_='sc-dLMFU gVWdyu JourneyTime__CityFont-sc-1ej0v4k-

print("=" 35)
print(f'{"Viaje "      str(numero_viaje):^35}')
print(f'{"- Trayecto "  trayecto_tipo upper()    ":"^35}')
print(f"Ciudad de origen: {ciudad_origen} -> Ciudad de destino: {ciudad_destino}")
print(f"Hora de salida: {hora_salida} -> Hora de llegada: {hora_llegada}")

precios_extraidos  extraer_precios(trayecto_html)
i, precio_info  enumerate(precios_extraidos):
print(f"=====")
print(f'{"          Precio "      str(i 1)    ":"^35}')
print(f'{"PrecioMonto:"} {precio_info["PrecioMonto"]}')
print(f'{"PrecioDecimales:"} {precio_info["PrecioDecimales"]}')
print(f'{"PrecioDivisa:"} {precio_info["PrecioDivisa"]}')
print(f'{"=" 35)

precios.extend(precios_extraidos)

info_viaje  {
    "Número de Viaje": numero_viaje,
    "Tipo de Trayecto": trayecto_tipo,
    "Ciudad de Origen": ciudad_origen,
    "Ciudad de Destino": ciudad_destino,
    "Hora de Salida": hora_salida,
    "Hora de Llegada": hora_llegada,
    "Fecha IDA": fecha_ida,
    "Fecha VUELTA": fecha_vuelta,
    "Precios": precios_extraidos
}

    info_viaje

    Exception    e:
print(f"Error al obtener información del trayecto: {str(e)}")

```

En este bloque principal, se configura el navegador, se procesan las dos URLs, y se exportan los resultados a archivos CSV, JSON y Excel. También, se manejan excepciones durante el proceso.

In [9]:

```

procesar_url(url, driver):
driver.get(url)

# Espera hasta que el botón de aceptar cookies esté presente
accept_cookies_button  WebDriverWait(driver, 10) until(
    EC.presence_of_element_located((By.ID, "CybotCookiebotDialogBodyButtonAccept"))
)
accept_cookies_button.click()

# Espera hasta que al menos un trayecto esté presente
trayectos  WebDriverWait(driver, 10) until(
    EC.presence_of_all_elements_located((By.XPATH, '//div[@class="styles__Journey-sc-lornisu-4 gOhLrT"]'))
)

# Lista para almacenar información de viajes y precios
lista_info_viajes  []
lista_precios  []

# Obtiene las fechas desde la URL
fecha_ida, fecha_vuelta  obtener_fecha_desde_url(url)

# Itera sobre los elementos de trayecto y llama a la función obtener_info_viaje intercalando IDA y VUELTA
i, trayecto_element  enumerate(trayectos):
    numero_viaje  i  2  1 # Calcula el número de viaje (división entera)
    i  2  0:
        info_viaje  obtener_info_viaje(trayecto_element, "IDA", lista_precios, numero_viaje, fecha_ida, fecha_v
        info_viaje:
            lista_info_viajes.append(info_viaje)
        :
        info_viaje  obtener_info_viaje(trayecto_element, "VUELTA", lista_precios, numero_viaje, fecha_ida, fech
        info_viaje:
            lista_info_viajes[1]["Precios"].extend(info_viaje["Precios"])

```

```
lista_info_viajes

# Configuración del navegador (en este caso, se utiliza Firefox con GeckoDriver)
firefox_options webdriver FirefoxOptions()
firefox_options add_argument("-private") # Habilita el modo incógnito
driver webdriver Firefox(options firefox_options)

:
# Procesa la primera URL
url_1 "https://www.atrapalo.com/ms/results?journeys=VG0%2CMAD%2C20240228%7CMAD%2CVGO%2C20240229"
resultados_1 procesar_url(url_1, driver)

# Procesa la segunda URL
url_2 "https://www.atrapalo.com/ms/results?journeys=VG0%2CMAD%2C20240229%7CMAD%2CVGO%2C20240301"
resultados_2 procesar_url(url_2, driver)

# Exportar datos a CSV
csv_file_path os path join(ruta_exportacion, "resultadoAtrapaloSelenium.csv")
open(csv_file_path, "w", newline "", encoding "utf-8") csv_file:
csv_writer csv DictWriter(csv_file, fieldnames resultados_1[0] keys())
csv_writer writeheader()
csv_writer writerows(resultados_1 resultados_2)

# Exportar datos a JSON
json_file_path os path join(ruta_exportacion, "resultadoAtrapaloSelenium.json")
open(json_file_path, "w", encoding "utf-8") json_file:
json dump(resultados_1 resultados_2, json_file, ensure_ascii , indent 4)

# Exportar datos a XLS (Excel)
excel_file_path os path join(ruta_exportacion, "resultadoAtrapaloSelenium.xlsx")
df pd DataFrame(resultados_1 resultados_2)
df to_excel(excel_file_path, index )

print("Archivos exportados exitosamente.")
Exception export_error:
print(f"Error al exportar archivos: {str(export_error)}")
:
# Cierra el navegador
driver quit()
```

```
=====
Viaje 1
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 09:20 -> Hora de llegada: 10:35
=====
Viaje 1
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 07:20 -> Hora de llegada: 08:40
=====
Viaje 2
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 06:30 -> Hora de llegada: 07:50
=====
Viaje 2
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 07:20 -> Hora de llegada: 08:40
=====
Viaje 3
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:40 -> Hora de llegada: 19:00
=====
Viaje 3
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 07:20 -> Hora de llegada: 08:40
=====
Viaje 4
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 09:20 -> Hora de llegada: 10:35
=====
Viaje 4
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 5
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 09:20 -> Hora de llegada: 10:35
=====
Viaje 5
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 6
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 09:20 -> Hora de llegada: 10:35
=====
Viaje 6
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 11:30 -> Hora de llegada: 12:50
=====
Viaje 7
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:00 -> Hora de llegada: 18:10
=====
Viaje 7
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 8
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:00 -> Hora de llegada: 18:10
=====
Viaje 8
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 9
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 21:05 -> Hora de llegada: 22:15
```

```
=====
Viaje 9
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 10
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 21:05 -> Hora de llegada: 22:15
=====
Viaje 10
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 11
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 06:30 -> Hora de llegada: 07:50
=====
Viaje 11
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 12
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 06:30 -> Hora de llegada: 07:50
=====
Viaje 12
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 13
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 06:30 -> Hora de llegada: 07:50
=====
Viaje 13
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 11:30 -> Hora de llegada: 12:50
=====
Viaje 14
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:40 -> Hora de llegada: 19:00
=====
Viaje 14
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 15
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:40 -> Hora de llegada: 19:00
=====
Viaje 15
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 16
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:40 -> Hora de llegada: 19:00
=====
Viaje 16
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 11:30 -> Hora de llegada: 12:50
=====
Viaje 17
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 13:30 -> Hora de llegada: 14:45
=====
Viaje 17
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 07:20 -> Hora de llegada: 08:40
```

```
=====
Viaje 18
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 08:40 -> Hora de llegada: 09:50
=====
Viaje 18
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 19
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 08:40 -> Hora de llegada: 09:50
=====
Viaje 19
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 20
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 12:35 -> Hora de llegada: 13:45
=====
Viaje 20
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 06:45 -> Hora de llegada: 07:50
=====
Viaje 21
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 12:35 -> Hora de llegada: 13:45
=====
Viaje 21
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 10:40 -> Hora de llegada: 11:45
=====
Viaje 22
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 17:00 -> Hora de llegada: 18:10
=====
Viaje 22
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 07:20 -> Hora de llegada: 08:40
=====
Viaje 23
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 21:05 -> Hora de llegada: 22:15
=====
Viaje 23
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 07:20 -> Hora de llegada: 08:40
=====
Viaje 24
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 09:20 -> Hora de llegada: 10:35
=====
Viaje 24
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 22:15 -> Hora de llegada: 23:35
=====
Viaje 25
- Trayecto IDA:
Ciudad de origen: VGO -> Ciudad de destino: MAD
Hora de salida: 06:30 -> Hora de llegada: 07:50
=====
Viaje 25
- Trayecto VUELTA:
Ciudad de origen: MAD -> Ciudad de destino: VGO
Hora de salida: 22:15 -> Hora de llegada: 23:35
Error al exportar archivos: Message:
Stacktrace:
RemoteError@chrome://remote/content/shared/RemoteError.sys.mjs:8:8
WebDriverError@chrome://remote/content/shared/webdriver/Errors.sys.mjs:191:5
NoSuchElementError@chrome://remote/content/shared/webdriver/Errors.sys.mjs:509:5
```

```
dom.find("</<@chrome://remote/content/shared/DOM.sys.mjs:136:16
```

In []: