import requests

from bs4 import BeautifulSoup

# Función para extraer datos de una sola página

def scrape\_page(url):

    # Realiza una solicitud GET a la página

    response = requests.get(url)

    if response.status\_code != 200:

        print(f"Error al acceder a la página {url}: {response.status\_code}")

        return [], [], [], []

    # Parsear el contenido HTML de la página

    soup = BeautifulSoup(response.text, 'html.parser')

    # Listas para almacenar los datos de esta página

    dates = []

    buy\_prices = []

    sell\_prices = []

    variations = []

    # Encuentra todos los elementos <td> con la clase "text-right"

    text\_right\_elements = soup.find\_all('td', {'class': 'text-right'})

    # Encuentra todos los elementos <td> que contienen un enlace <a> con href="/dolar-blue/cotizacion/"

    td\_elements\_with\_link = soup.find\_all('td')

    # Asume que los primeros elementos "text-right" corresponden a precios de compra, los siguientes a venta.

    for i in range(0, len(text\_right\_elements), 3):

        buy\_prices.append(text\_right\_elements[i].text.strip())

        sell\_prices.append(text\_right\_elements[i + 1].text.strip())

        variations.append(text\_right\_elements[i + 2].text.strip())

    # Asume que las fechas están en los <td> que contienen los enlaces, de ser así, obtén las fechas:

    for td in td\_elements\_with\_link:

        a\_tag = td.find('a', href=True)

        if a\_tag and '/dolar-oficial/cotizacion/' in a\_tag['href']:

            dates.append(a\_tag.text.strip())

    return dates, buy\_prices, sell\_prices, variations

# URLs base y lista de resultados

base\_url = 'https://dolarhistorico.com/cotizacion-dolar-oficial/mes/'

all\_dates = []

all\_buy\_prices = []

all\_sell\_prices = []

all\_variations = []

# Años y meses que quieres recorrer

years = ['2021','2022','2023','2024']

months = [

    'enero', 'febrero', 'marzo', 'abril', 'mayo',

    'junio', 'julio', 'agosto', 'septiembre', 'octubre',

    'noviembre', 'diciembre'

]

# Bucle sobre cada combinación de año y mes para construir la URL y recoger los datos

for year in years:

    for month in months:

        url = f'{base\_url}{month}-{year}'

        print(f"Scrapeando datos de: {month}-{year}")

        dates, buy\_prices, sell\_prices, variations = scrape\_page(url)

        # Agregar datos a las listas globales

        all\_dates.extend(dates)

        all\_buy\_prices.extend(buy\_prices)

        all\_sell\_prices.extend(sell\_prices)

        all\_variations.extend(variations)

# Formatear y mostrar todos los datos recogidos

print(f"\n{'Fecha':<12} {'Compra':<10} {'Venta':<10} {'Variación':<10}")

print("-" \* 45)

for date, buy, sell, variation in zip(all\_dates, all\_buy\_prices, all\_sell\_prices, all\_variations):

    print(f"{date:<12} {buy:<10} {sell:<10} {variation:<10}")