import pygetwindow as gw

import pandas as pd

import time

import re

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.support.ui import WebDriverWait

from selenium.webdriver.support import expected\_conditions as EC

from selenium.webdriver.firefox.service import Service

from webdriver\_manager.firefox import GeckoDriverManager

import pyperclip

import pyautogui

def acessar\_url(driver, url):

driver.get(url)

print("URL acessada.")

def copiar\_nosso\_numero(driver, nosso\_numero):

try:

iframes = driver.find\_elements(By.TAG\_NAME, "iframe")

for iframe in iframes:

driver.switch\_to.frame(iframe)

try:

processo\_input = driver.find\_element(By.ID, "ProcessoNumero")

if processo\_input:

break

except:

driver.switch\_to.default\_content()

continue

processo\_input = WebDriverWait(driver, 2).until(

EC.element\_to\_be\_clickable((By.ID, "ProcessoNumero"))

)

processo\_input.clear()

processo\_input.send\_keys(nosso\_numero)

print(f"Colou o número do processo: {nosso\_numero}")

except Exception as e:

print(f"Erro ao colar o número do processo: {e}")

finally:

driver.switch\_to.default\_content()

def clicar\_botao\_buscar(driver):

try:

botao\_buscar = WebDriverWait(driver, 1).until(

EC.element\_to\_be\_clickable((By.XPATH, '/html/body/div[3]/form/div/fieldset/div[5]/input[1]'))

)

botao\_buscar.click()

print("Botão 'Buscar' clicado.")

except Exception as e:

print(f"Erro ao clicar no botão 'buscar': {e}")

def focar\_janela\_navegador(title\_part):

try:

windows = [win for win in gw.getAllTitles() if title\_part in win]

if windows:

window = gw.getWindowsWithTitle(windows[0])[0]

if window.isMinimized:

window.restore()

window.activate()

print(f"Janela do navegador '{title\_part}' em foco.")

else:

print(f"Janela com título contendo '{title\_part}' não encontrada.")

except Exception as e:

print(f"Erro ao focar a janela do navegador: {e}")

def buscar\_numero\_guia(driver, numero\_guia):

try:

pyautogui.hotkey('ctrl', 'f')

time.sleep(1)

pyautogui.hotkey('ctrl', 'a')

time.sleep(0.5)

pyautogui.press('backspace')

time.sleep(0.5)

pyautogui.write(str(numero\_guia))

pyautogui.press('enter')

print(f"Número da guia '{numero\_guia}' digitado no navegador.")

except Exception as e:

print(f"Erro ao digitar número guia: {e}")

def localizar\_e\_copiar\_valores\_por\_texto(driver):

try:

# Obtenha o texto completo da página

page\_source = driver.page\_source

print("Página capturada") # Print de confirmação

# Expressões regulares ajustadas para capturar os valores

patterns = {

"valor\_original": r"<b>Valor Originário da Guia</b>.\*?<td align=\"right\"><b>(.\*?)</b></td>",

"juros": r"JUROS FUNDESP-PJ S F\(Reg\.JURO\)</td>\s\*<td align=\"center\">.\*?</td>\s\*<td align=\"center\">.\*?</td>\s\*<td align=\"right\">(.\*?)</td>",

"multa": r"MULTA FUNDESP/PJ\(Reg\.M\.FUNDESP\)</td>\s\*<td align=\"center\">.\*?</td>\s\*<td align=\"center\">.\*?</td>\s\*<td align=\"right\">(.\*?)</td>",

"total\_guia": r"Total Guia Atualizado \(\d{2}/\d{2}/\d{4}\)</b>.\*?<td align=\"right\"><b>(.\*?)</b></td>"

}

# Dicionário para armazenar os valores encontrados

valores = {}

for key, pattern in patterns.items():

match = re.search(pattern, page\_source, re.DOTALL)

if match:

valores[key] = match.group(1).strip()

print(f"{key} encontrado: {valores[key]}")

else:

print(f"Não foi possível localizar {key} na página.")

return valores

except Exception as e:

print(f"Erro ao copiar os valores: {e}")

return None

def focar\_janela\_excel():

try:

excel\_windows = [win for win in gw.getAllTitles() if 'Excel' in win]

if excel\_windows:

excel\_window = gw.getWindowsWithTitle(excel\_windows[0])[0]

if excel\_window.isMinimized:

excel\_window.restore()

excel\_window.activate()

print("Janela do Excel em foco.")

else:

print("Janela do Excel não encontrada.")

except Exception as e:

print(f"Erro ao focar a janela do Excel: {e}")

def colar\_dados\_na\_planilha(linha, coluna, valor):

try:

focar\_janela\_excel() # Assegura que a janela do Excel está em foco

time.sleep(0.5) # Pequena pausa para mudança de foco

pyautogui.hotkey('ctrl', 'g')

time.sleep(0.5)

pyautogui.write(f"{coluna}{linha + 1}")

pyautogui.press('enter')

time.sleep(0.5)

pyperclip.copy(valor)

pyautogui.hotkey('ctrl', 'v')

print(f"Valor '{valor}' colado na célula {coluna}{linha + 1}.")

except Exception as e:

print(f"Erro ao colar dados na planilha: {e}")

def salvar\_planilha():

try:

focar\_janela\_excel()

pyautogui.hotkey('ctrl', 'b')

time.sleep(1)

print("Planilha salva.")

except Exception as e:

print(f"Erro ao salvar a planilha: {e}")

def main():

caminho\_arquivo = r'C:/Users/mferreirasantos/Documents/demanda DANILO - Atualizar valores.xlsx'

nome\_planilha = 'Planilha1'

tabela\_boleto = pd.read\_excel(caminho\_arquivo, sheet\_name=nome\_planilha, skiprows=0)

url = 'https://projudi.tjgo.jus.br/BuscaProcesso?PaginaAtual=4&TipoConsultaProcesso=24&ServletRedirect=GuiaFinal\_FinalZeroPublica&TituloDaPagina=Guia+Final+e+Final+Zero+%5BAcesso+P%C3%BAblico%5D&hashFluxo=1722628305255'

options = webdriver.FirefoxOptions()

with webdriver.Firefox(service=Service(GeckoDriverManager().install()), options=options) as driver:

acessar\_url(driver, url)

for index, row in tabela\_boleto.iterrows():

nosso\_numero = row['Número Processo']

numero\_guia = row['Número Guia']

copiar\_nosso\_numero(driver, nosso\_numero)

clicar\_botao\_buscar(driver)

focar\_janela\_navegador("Projudi")

buscar\_numero\_guia(driver, numero\_guia)

valores = localizar\_e\_copiar\_valores\_por\_texto(driver)

print(f"Valores extraídos: {valores}")

if valores:

colar\_dados\_na\_planilha(index + 1, 'F', valores.get('valor\_original', ''))

colar\_dados\_na\_planilha(index + 1, 'G', valores.get('juros', ''))

colar\_dados\_na\_planilha(index + 1, 'H', valores.get('multa', ''))

colar\_dados\_na\_planilha(index + 1, 'I', valores.get('total\_guia', ''))

salvar\_planilha()

print("Reiniciando para o próximo número do processo...\n")

time.sleep(1)

acessar\_url(driver, url)

if \_\_name\_\_ == "\_\_main\_\_":

main()

**USAR ESTE**

import pygetwindow as gw

import pandas as pd

import time

import re

from selenium import webdriver

from selenium.webdriver.common.by import By

from selenium.webdriver.support.ui import WebDriverWait

from selenium.webdriver.support import expected\_conditions as EC

from selenium.webdriver.firefox.service import Service

from webdriver\_manager.firefox import GeckoDriverManager

import pyperclip

import pyautogui

from bs4 import BeautifulSoup

def acessar\_url(driver, url):

driver.get(url)

print("URL acessada.")

def copiar\_nosso\_numero(driver, nosso\_numero):

try:

iframes = driver.find\_elements(By.TAG\_NAME, "iframe")

for iframe in iframes:

driver.switch\_to.frame(iframe)

try:

processo\_input = driver.find\_element(By.ID, "ProcessoNumero")

if processo\_input:

break

except:

driver.switch\_to.default\_content()

continue

processo\_input = WebDriverWait(driver, 2).until(

EC.element\_to\_be\_clickable((By.ID, "ProcessoNumero"))

)

processo\_input.clear()

processo\_input.send\_keys(nosso\_numero)

print(f"Colou o número do processo: {nosso\_numero}")

except Exception as e:

print(f"Erro ao colar o número do processo: {e}")

finally:

driver.switch\_to.default\_content()

def clicar\_botao\_buscar(driver):

try:

botao\_buscar = WebDriverWait(driver, 1).until(

EC.element\_to\_be\_clickable((By.XPATH, '/html/body/div[3]/form/div/fieldset/div[5]/input[1]'))

)

botao\_buscar.click()

print("Botão 'Buscar' clicado.")

except Exception as e:

print(f"Erro ao clicar no botão 'buscar': {e}")

def focar\_janela\_navegador(title\_part):

try:

windows = [win for win in gw.getAllTitles() if title\_part in win]

if windows:

window = gw.getWindowsWithTitle(windows[0])[0]

if window.isMinimized:

window.restore()

window.activate()

print(f"Janela do navegador '{title\_part}' em foco.")

else:

print(f"Janela com título contendo '{title\_part}' não encontrada.")

except Exception as e:

print(f"Erro ao focar a janela do navegador: {e}")

def buscar\_numero\_guia(driver, numero\_guia):

try:

pyautogui.hotkey('ctrl', 'f')

time.sleep(1)

pyautogui.hotkey('ctrl', 'a')

time.sleep(0.5)

pyautogui.press('backspace')

time.sleep(0.5)

pyautogui.write(str(numero\_guia))

pyautogui.press('enter')

print(f"Número da guia '{numero\_guia}' digitado no navegador.")

except Exception as e:

print(f"Erro ao digitar número guia: {e}")

def localizar\_e\_copiar\_valores\_por\_texto(driver):

try:

# Obtenha o texto completo da página

page\_source = driver.page\_source

print("Página capturada")

# Analise o HTML com BeautifulSoup

soup = BeautifulSoup(page\_source, 'html.parser')

valores = {}

# Valor Original

valor\_original = soup.find(text="Valor Originário da Guia")

if valor\_original:

valor\_original\_td = valor\_original.find\_next('td', align="right")

if valor\_original\_td:

valores["valor\_original"] = valor\_original\_td.text.strip()

# JUROS FUNDESP-PJ S F

juros\_row = soup.find('td', text="JUROS FUNDESP-PJ S F(Reg.JURO)")

if juros\_row:

juros\_td = juros\_row.find\_next('td', align="right")

if juros\_td:

valores["juros"] = juros\_td.text.strip()

# MULTA FUNDESP/PJ

multa\_row = soup.find('td', text="MULTA FUNDESP/PJ(Reg.M.FUNDESP)")

if multa\_row:

multa\_td = multa\_row.find\_next('td', align="right")

if multa\_td:

valores["multa"] = multa\_td.text.strip()

# Total Guia Atualizado

total\_guia\_text = soup.find(text=re.compile("Total Guia Atualizado"))

if total\_guia\_text:

total\_guia\_td = total\_guia\_text.find\_next('td', align="right")

if total\_guia\_td:

valores["total\_guia"] = total\_guia\_td.text.strip()

for key, value in valores.items():

print(f"{key} encontrado: {value}")

return valores

except Exception as e:

print(f"Erro ao copiar os valores: {e}")

return None

def focar\_janela\_excel():

try:

excel\_windows = [win for win in gw.getAllTitles() if 'Excel' in win]

if excel\_windows:

excel\_window = gw.getWindowsWithTitle(excel\_windows[0])[0]

if excel\_window.isMinimized:

excel\_window.restore()

excel\_window.activate()

print("Janela do Excel em foco.")

else:

print("Janela do Excel não encontrada.")

except Exception as e:

print(f"Erro ao focar a janela do Excel: {e}")

def colar\_dados\_na\_planilha(linha, coluna, valor):

try:

focar\_janela\_excel() # Assegura que a janela do Excel está em foco

time.sleep(0.5) # Pequena pausa para mudança de foco

pyautogui.hotkey('ctrl', 'g')

time.sleep(0.5)

pyautogui.write(f"{coluna}{linha + 1}")

pyautogui.press('enter')

time.sleep(0.5)

pyperclip.copy(valor)

pyautogui.hotkey('ctrl', 'v')

print(f"Valor '{valor}' colado na célula {coluna}{linha + 1}.")

except Exception as e:

print(f"Erro ao colar dados na planilha: {e}")

def salvar\_planilha():

try:

focar\_janela\_excel()

pyautogui.hotkey('ctrl', 'b')

time.sleep(1)

print("Planilha salva.")

except Exception as e:

print(f"Erro ao salvar a planilha: {e}")

def main():

caminho\_arquivo = r'C:/Users/mferreirasantos/Documents/demanda DANILO - Atualizar valores.xlsx'

nome\_planilha = 'Planilha1'

tabela\_boleto = pd.read\_excel(caminho\_arquivo, sheet\_name=nome\_planilha, skiprows=0)

url = 'https://projudi.tjgo.jus.br/BuscaProcesso?PaginaAtual=4&TipoConsultaProcesso=24&ServletRedirect=GuiaFinal\_FinalZeroPublica&TituloDaPagina=Guia+Final+e+Final+Zero+%5BAcesso+P%C3%BAblico%5D&hashFluxo=1722628305255'

options = webdriver.FirefoxOptions()

with webdriver.Firefox(service=Service(GeckoDriverManager().install()), options=options) as driver:

acessar\_url(driver, url)

for index, row in tabela\_boleto.iterrows():

nosso\_numero = row['Número Processo']

numero\_guia = row['Número Guia']

copiar\_nosso\_numero(driver, nosso\_numero)

clicar\_botao\_buscar(driver)

focar\_janela\_navegador("Projudi")

buscar\_numero\_guia(driver, numero\_guia)

valores = localizar\_e\_copiar\_valores\_por\_texto(driver)

print(f"Valores extraídos: {valores}")

if valores:

colar\_dados\_na\_planilha(index + 1, 'F', valores.get('valor\_original', ''))

colar\_dados\_na\_planilha(index + 1, 'G', valores.get('juros', ''))

colar\_dados\_na\_planilha(index + 1, 'H', valores.get('multa', ''))

colar\_dados\_na\_planilha(index + 1, 'I', valores.get('total\_guia', ''))

salvar\_planilha()

print("Reiniciando para o próximo número do processo...\n")

time.sleep(1)

acessar\_url(driver, url)

if \_\_name\_\_ == "\_\_main\_\_":

main()