import re

from selenium import webdriver

from time import sleep

from selenium.webdriver.chrome.options import Options

import time

from datetime import datetime

import pandas as pd

from konlpy.tag import Okt

import bs4

from selenium.common.exceptions import TimeoutException

query = input("저장이름 입력")

source = "네이버 블로그"

keyword = input('검색어떤걸로 함')

t = Okt()

now = datetime.now() # 파일이름 현 시간으로 저장하기

RESULT\_PATH = 'C:/Users/wnstj/PycharmProjects/BigData/Crawling/' # 결과 저장할 경로/파일명 (본인 경로로 수정)

outputFileName = query + '%s-%s-%s %s시 %s분.csv' % (

now.year, now.month, now.day, now.hour, now.minute)

driver = webdriver.Chrome('./chromedriver')

driver.implicitly\_wait(3)

def clean\_text(text\_in\_file):

text\_in\_file\_1th = re.sub('[a-zA-Z]', '', text\_in\_file)

text\_in\_file\_2th = re.sub('[\{\}\[\]\/?.,;:|\)\*~`!^\-\_+<>@\#$%&\\\=\(\'\"]', '', text\_in\_file\_1th)

return text\_in\_file\_2th

def save\_csv(RR):

try:

df = pd.DataFrame(RR) # df로 변환

df.columns = ["Keyword", "Contents", "Words", "Source", "URL"]

df.to\_csv(RESULT\_PATH + outputFileName, mode='w')

except:

df = pd.DataFrame(RR) # df로 변환

df.columns = ["URL"]

df.to\_csv(RESULT\_PATH + '블로그 URL 리스트' + outputFileName, mode='w')

def read\_url(filename):

df = pd.read\_csv(filename, encoding='utf-8')

df2 = df['0']

driver.implicitly\_wait(3)

all = []

pn = 1

try:

for url in df2:

print(pn)

pn = pn + 1

time.sleep(1)

body\_list = [keyword]

driver.get(url)

current\_url = driver.current\_url

html = driver.page\_source

soup = bs4.BeautifulSoup(html, 'html.parser')

time.sleep(2)

try:

driver.switch\_to.frame('mainFrame')

try:

tbody = driver.find\_elements\_by\_id('postViewArea')

for txt in tbody:

c\_txt = txt.text.replace('\n', '')

print(txt.text.replace('\n', ''))

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

try:

tbody = driver.find\_elements\_by\_class\_name('se-main-container')

for txt in tbody:

c\_txt = txt.text.replace('\n', '')

print(txt.text.replace('\n', ''))

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

try:

tbody = driver.find\_elements\_by\_class\_name('se\_component\_wrap')

ttt = tbody[1]

c\_txt = ttt.text.replace("\n", "")

print(c\_txt)

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

except:

pass

try:

driver.switch\_to.frame('hiddenFrame')

try:

tbody = driver.find\_elements\_by\_id('postViewArea')

for txt in tbody:

c\_txt = txt.text.replace('\n', '')

print(txt.text.replace('\n', ''))

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

try:

tbody = driver.find\_elements\_by\_class\_name('se-main-container')

for txt in tbody:

c\_txt = txt.text.replace('\n', '')

print(txt.text.replace('\n', ''))

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

try:

tbody = driver.find\_elements\_by\_class\_name('se\_component\_wrap')

ttt = tbody[1]

c\_txt = ttt.text.replace("\n", "")

print(c\_txt)

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

except:

pass

try:

frame = soup.find\_all('frame', {'id':'screenFrame'})

for i in frame:

screen\_src = i['src']

# print(screen\_src)

driver.get(screen\_src)

driver.switch\_to.frame('mainFrame')

try:

tbody = driver.find\_elements\_by\_id('postViewArea')

for txt in tbody:

c\_txt = txt.text.replace('\n', '')

print(txt.text.replace('\n', ''))

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

try:

tbody = driver.find\_elements\_by\_class\_name('se-main-container')

for txt in tbody:

c\_txt = txt.text.replace('\n', '')

print(txt.text.replace('\n', ''))

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

try:

tbody = driver.find\_elements\_by\_class\_name('se\_component\_wrap')

ttt = tbody[1]

c\_txt = ttt.text.replace("\n", "")

print(c\_txt)

word = t.nouns(c\_txt)

word = str(word).strip('[]')

print(word)

body\_list.append(c\_txt)

body\_list.append(word)

body\_list.append(source)

body\_list.append(current\_url)

except:

pass

except:

pass

if(len(body\_list) == 5):

all.append(body\_list)

except TimeoutException as e:

print('오류 다음으로 넘어갑니다', format(type(e)))

save\_csv(all)

if \_\_name\_\_ == '\_\_main\_\_':

read\_url('세제2019-8-19 14시 37분.csv')