검색 및 크롤링 flow

웹 크롤링을 검색 - 파이썬 웹 크롤링 방법
- beautifulSoup / selenium 키워드 발견
- 장단점과 웹 크롤링에 대한 정보 검색
- 쉽고 복잡한 무한스크롤방한 사이트에서 원하는 키워드 같을 크롤링하는 것이었기에) 무한스크롤 웹 크롤링 검색
- python 으로 구현하는 코드들을 발견
- anaconda 등을 세팅 할거다 or 그 python 코드를 짜는 것은 상당 힘들겠기에 결국이긴 VSCode 를 사용하기로 결정
- vscode selenium 크롤링 원하 키워드로 검색
- selenium 설치 및 사용법 숙지
- 무한스크롤 → 스크롤을 내리는 방법에 대한 정보 습득 - 여러가지 적용 및 연관된 테스트
- 상당 양은 정보를 손실 없이 끌어올 수 있는 방법 체택
- 테스트 구조
ex) 1번 방법 : 100개 중 40개 크롤, 60개 손실
/ 2번 방법 : 100개 중 70개 크롤, 30 손실, 30 중복
/ 3번 방법 : 100개 중 60개 크롤, 40 손실, 20중복

→ 2번 방법 체택 (한계는 적은 손실로 많은 양의 데이터가 필요, 중복은 수학적으로 (역설 또는 또 다른 프로그램, 코딩 등 이용) 체택시는 상고 중복데이터도 사용 가능한 상황)
- 테스트 코드 버전으로 도구인생 놀림
- 중복 및 에러 대처 체가
- 이동

selenium

1. Selenium 패키지란?

selenium 패키지는 chromedriver를 제어하거나 원하는 정보를 얻기 위해 사용합니다.

크롤링을 하기보단 무엇인가 입력하거나 특정 버튼을 눌러야 하는 상황이 발생합니다. 사람이 그러한 행동을 하는 대신 컴퓨터가 할 수 있도록 해주는 패키지가 selenium입니다.

selenium 패키지도 웹 정보를 크롤링하는 것이 가능하기 때문에 저는 웹 크롤링 간 가끔 selenium을 패키지를 사용하는 편입니다.

이제고 피피고 알아 있지만 읽어두면 좋음.

출처 : 위키독스 - 시험날 볼래 하는 파이썬 입문자들의 - 오픈웹크롤링 한 부터 웹 파싱 기 간단한 주한 내용

49 동적 웹크롤링 - selenium 소개 및 기초사용법
49. Selenium 패키지와 selenium 패키지는 chromedriver을...
wikidocs.net

selenium 설치법

0. Python 설치
1. VSCode 설치 (대는 Anaconda, Spider, Jupyter Notebook 등등등등 또는 그냥 CLI 에서 실행하면 설치안해도 됨)
2. Python 가상 환경 설정
3. 크롤 드라이버 다운

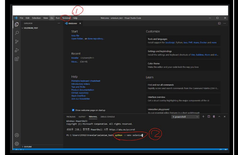
0. Python 설치
- https://www.python.org/downloads/ 에서 운영체제에 맞는 파이썬 설치

1. VSCode 설치 (대는 Anaconda, Spider, Jupyter Notebook 등등등등 또는 그냥 CLI 에서 실행하면 설치안해도 됨)
- https://code.visualstudio.com/ 에서 운영체제에 맞는 VSCode 설치

2. Python 가상 환경 설정
- VSCode 에서 Open Folder 로 작업할 폴더 열기
- VSCode 에서 Terminal 열기
- Terminal 에서

python -m venv selenium

입력



이후

cd selenium/Script

입력 해서 selenium/Scripts 폴더로 이동, 이동 후

\activate

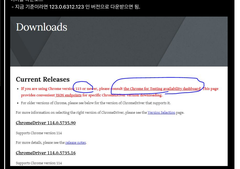
입력

```
PS C:\Users\ithit\travelar\selenium_test> cd selenium\Scripts\
PS C:\Users\ithit\travelar\selenium_test\selenium\Scripts> .\activate
(selenium) PS C:\Users\ithit\travelar\selenium_test\selenium\Scripts>
```

3. 크롤 드라이버 다운
- 크롤 드라이버 다운 전 크롬의 버전을 확인해야함

- 크롬 열기
- 우측 상단 점 3개
- 도움말 (?)
- Chrome 정보 (G)

- 대충 업데이트 끝나면 현재 버전 알아둠
- https://chromedriver.chromium.org/downloads 에 들어가서 크롬 버전과 같은 크롬 드라이버를 다운로드
- 지금 기준이라면 123.0.6312.123 인 버전으로 다운받으면 됨.



- 즉, 115 이상이라 저 빨간 link 들어가서 버전 맞게 다운 하는 방법도 있음.

- 다운 후 압축을 풀어 chromedriver.exe 를 selenium 폴더 안으로 이동.
(자동 실행 및 py 파일과 chromedriver.exe 가 같이 있어야 함)

관련 내용 정보
(위단 링크 중 첫 링크 (history) 에서 기술한 오류관련 링크는 현재 없는 주소, 오류내용은 우측 section 에 정리해 둠)

[Python]Python selenium을 이용하여...
파이썬을 이용하여 여러 크롤링 후 다운로드하는 것...
tae-hui.tistory.com

selenium 사용하여 웹 페이지 크롤링하기(...
간단해 있는 본격한 정보에 적용하여 주세요!...
oshig.tt

다른 및 selenium test 실행하지는 거의 이 블로그 다녀감.

mac 버전 / brew 를 이용한 다운방법 (brew 가 필요대요)

[Python] VS Code에서 셀레니움 설치...
VS Code 에서 selenium 실행하는 방법은 간단해요...
dolang-gogume.tistory.com

[mac] m1 웹 크롤링 Selenium, Chromedriv...
크롤링을 위한 Selenium, Chromedriver을...
dongmin98.tistory.com

selenium 실행 시 권한단 오류

```
(activate
---
CategoryInfo : 보안 오류: (.):[], PSSecurityException
FullyQualifiedErrorId : UnauthorizedAccess
```

문제는 activate.ps1 Scripts 안에 위치한 python.exe를 실행시켜야하는데, 오류를 뿌려대니나 보안 문제로 인해 activate를 할 수 없는는 것

권한 오류 시지제발시) VSCode 의 기본 terminal 이 PowerShell 로 되어 있는데 PowerShell 을 관리자 권한으로 열거나 PowerShell 대신 CMD 로 열었든 뭐든 상관 없음.
→ VSCode 내부서 위일 방법 찾기만하시 terminal 로 작업하면 좋으니 기본 설정을 변경하기로 함.

관련 내용 정보

[Python] VSCode 오류 해결 - 가상환경 set...
문제상황 파이썬 인터프 Selenium 가상환경에 대해 제...
effort.tistory.com

vscode 터미널 powershell을 cmd로 변경하...
최초 vscode에서 터미널을 열면 powershell이 기본값...
hanjoon.dev.tistory.com

무한스크롤인 사이트를 크롤링가 위한 방법 검색

일단 무한스크롤인 사이트를 크롤링하면

1. 현재 사이트의 내용 크롤링
2. 사이트의 스크롤 내리기
3. 내린 후 로딩된 데이터 크롤링
4. 반복

의 순서로 진행되야야 한다고 생각.

스크롤 방법에는 간단하게 3가지 방법이 있다고 한다.

1. 페이지의 높이를 계산해서 최하단으로 이동하는 방법.
2. 키보드 키 (PAGE_DOWN/END) 나 END, 아래 화살표 키를 반복적으로 눌러 페이지를 내려가는 방법
3. ActionChains.move_to_element() 를 이용하여 해당 요소로 이동하는 방법

3가지 방법을 일일 수 사용해보고지만
현재 포함한 내용 코드로는

개인은 테스트코드를 작성해서 실행해 보니 "무한스크롤" 인 사이트 기준으로 "최하단" 까 기준점이 애만 바뀌게 되고, 그게 아마 광 일하는 데이터의 양이 정확실해하고, 본 코드는 "하드코딩 data 가 로딩된 후 크롤링하는 방법의 마지막기 때문에 손실되는 data 도 많았다.

3번은 내가 크롤링할 사이트가 소팅을 이용도, 그 안에 상품이스트 중에서 이름, 가격, 사진 data 등을 가져오려 했었고, 그 때문 상당 번거롭기도 한 것이어, 일정한 element가 반복되고, 그중 한을 요청단한 특정 element 를 찾지 못하기도 했고, 이 스크로르 후 적당해보인다고 한 element 를 찾아내도 내 작인 문제가 있었어서 배제했다. (한에 다시 한다면 잘 할수도)

또 다른 방법은 특정 scroll 값을 주어서 그만큼 내려가게 하는 방법이 있었는데, 내 요약한 코드 내에서 방하값을 2 scroll height 한번 내려서 재대로 작동을 해서 있어서 포기했다.

그래서 찌질히 스크롤다운하는 방법 배로 정할 수 있고, 사진처럼도 주면서 실행을 하는 방법이고 2번의 방법을 택했다.

참고 사이트들에 여러 사람들이 각 방법과 장다른 방법들을 다양한 형식으로 해온 내용들이 있으니 참고할 만 함.

참고사이트

[Selenium] 웹크지를 스크롤하여 무한...
이번에는 웹사이 스크롤 기능을 이용해서 데이터를 가져...
dailyprogrammer.tistory.com

스크롤 방법에 관한 정보와 웹사이다양에 대한 이야기라면 있음

velog

[selenium]셀레니움(selenium)으로 크롤링하기...
무한스크롤 처럼하기 크롤링 하기 웹사이트를 공부하시 ...
velog.io

특정 상태라면 스크롤하는 내용과 사용한 웹사이트에 대한 간략한 설명 참고

[PYTHON] 파이썬 웹 크롤링 (무한 스크롤...
안녕하세요 파이썬과 관련한 여러 주제들로 찾아오는 ...
tourzia.tistory.com

사이트 전체 놓아인용 스크롤하는 방법과 소스 코드 시간 체크내용 참고

파이썬 Selenium으로 카카오페이지 접속...
이번에 셀레니움을 이용해서 카카오 페이지 로 접속 다운 받는...
korapy.blogspot.com

xPath 기준으로 검색하는 내용 참고

Selenium으로 웹 크롤링 자동화 구현하기
이 게시 작업 시 Selenium 패키지 설치 pip install sele...
mangchoi96.tistory.com

selenium 설치 및 class name 기준 검색, data parsing, df 통해 행렬인도는 내용 참고

[Crawling] 네이버 자동로그인
➡ 이는 및 크롤링 이해에 도움 되실제이지라야 하실...
velog.io

이건 별개되나 내용이지만 selenium을 이용한 네이버 자동 로그인 코드 내용

셀레니움(Selenium) 사용하기 - 네이버...
* 개발 환경 및 소프트웨어 설치 (버전 확인 필수) 1. 웹...
greenhouse.tistory.com

역시 적용로그인 후 크롤링 내용과 data csv 저장 내용, 개 좋고그 저장하게 mac 에서는 utf-8 도 인쇄안다고 함.

selenium 크롤링 - 무한 스크롤, 클릭
안녕 하고있어서 스타일에이 정 설명...
velog.io

역시 여러가지 방법 놓아서 첫 크롤링하는 방법, 무한스크롤 해 대한 내용, csv 저장 방법 참고

Pandas 데이터프레임을 CSV로 내보내는 방법
Pandas 데이터프레임을 표형식인 텍스트 파일 (보통은 쉼표 CSV 파일)...
docs.kanaries.net

pandas 를 통해 csv 파일 저장 하는 방법

그 외에도 참고하거나 거쳐간 사이트가 많은데 다시 찾는데 기억이 안나... ㅠ

실행한 코드_img

shop_furniture.py

```
from selenium import webdriver # 크롬 드라이버를 사용하기 위한 import
from selenium.webdriver.common.by import By # 크롤드라이버로 선택지를 찾기 위한 import
from selenium.webdriver.common.keys import Keys # 키보드 키를 누르는 역할을 위한 import
import time # data 로딩을 위해 시간지연을 사용하기 위한 import
import pandas as pd # 형/열을 만들어 data 처리를 하기 위한 import, pandas 는 별도의 설치가 필요함 ( pip istall pandas 등 을 통해)

driver = webdriver.Chrome() # 크롬 드라이버 가져오기
url = "https://ohou.se/store/category?category_id=10000000" # 크롤링 할 url
driver.get(url) # 크롤드라이버에 url 입력

time.sleep(5) # 시간지연, 초 기준

names = [] # 크롤링 한 내용을 담을 배열
prices = [] # 크롤링 한 내용을 담을 배열

actions = driver.find_element(By.CSS_SELECTOR, 'body') # 키보드 키를 누르는 영역을 설정

last_height = driver.execute_script("return document.body.scrollHeight") # 현재 최대 스크롤값 측정 -> last_height

while (last_height <= 2000000): # 최대 스크롤값이 2 이하인 동안 반복 => 크롤링할 데이터량 조절

    time.sleep(1) # 데이터 로딩 을 위한 시간지연

    name = driver.find_elements(By.CLASS_NAME, "production-item__header__name") # Class Name 기준으로 element 가져옴
    price = driver.find_elements(By.CLASS_NAME, "production-item-price__price") # Class Name 기준으로 element 가져옴

    for i in name:
        names.append(i.text) # name 에 해당하는 text 를 names 배열에 담음
    for i in price:
        prices.append(i.text) # price 에 해당하는 text 를 prices 배열에 담음

    time.sleep(1) # 아무튼 시간지연

    actions.send_keys(Keys.PAGE_DOWN) # 바디 기준으로 'page_down' 키를 한번 누름 ( end, enter, spacebar 등 여러 키 로 변경해서 응용 사용 가능 )
    time.sleep(0.1) # 연속적인 키 입력시간 고려해서 시간지연 다음 0.1초로 해줌 그래도 1번 연속이 잘 안됨, 0.5 초 일때 정확히 원하는 동수만큼 가능했었음.
    actions.send_keys(Keys.PAGE_DOWN)
    time.sleep(0.1)
    actions.send_keys(Keys.PAGE_DOWN)
    time.sleep(0.1)
    actions.send_keys(Keys.PAGE_DOWN)
    time.sleep(0.1)
    actions.send_keys(Keys.PAGE_DOWN)
    # 총 5번 page_down 했는데, 원하는 갯수만큼 조절 가능, 1회 크롤링 후 임시로 2차 테스트했을때는 3번도 충분했었을것 같기도..
    # 이 역시 크롤링 및 파지지와 데이터 로딩 주변에 따라 변동됨.

    time.sleep(1) # 웹사 로드시 스크롤 다음 한 주제도 데이터 로딩을 위한 시간지연

    current_height = driver.execute_script("return document.body.scrollHeight") # 스크롤 후 현재 최대 스크롤값을 측정
    print(current_height) # 스크롤량이 잘되지만 현재 어디쯤이시 모르니, 콘솔을 통해 진행률 확인 (현재 스크롤값 확인)

    if current_height == last_height: # 위에서 정한 last_height 과 현재 스크롤 값이 같으면 탈출 // 지금 생각해보니 이미 while 문에서 조건을 걸어놔서 필요없을수도.
        break

    last_height = current_height # 정한 범위 안이면 현재 스크롤 값 = last_height 로 치환 // 이것도 상동. 다시 보니 원래 처면 다른 방법의 코드에서 방법을 바꾸며 지우다 남은 코드인듯함.

print("\n")
print("가져온 상품 수: ", len(names)) # 크롤링 한 총 data 량 알고싶어서 콘솔로 출력
print("\n")
df = pd.DataFrame({'상품 이름':names, '가격':prices}) # pandas 의 형/열 정렬 시스템을 이용해서 상품이름/가격 column 에 각각 names/prices 입력
print(df) # pandas가 제대로 먹히는지 확인하고싶어서 콘솔로 추력
df.to_csv("01_shop_furniture.csv", index = False, encoding='cp949')
# DataFrame 인 기둥 중 하나로, 정렬로 된 data를 csv 파일로 저장 가능, 이 때 UTF-8 로 하면 깨짐, cp949 로 인코딩 후 oracle sql 에 넣을 때 다시 utf-8 로 인코딩 후 import 하면 됨.
# 또는 다른 좋은 방법이 있다면 그것도 추천

driver.quit() # 크롬드라이버 실행 중지
print("\n")
print('save completed') # 모든 기능이 끝났을 알기 위해 콘솔로 출력
```

실행 code 그대로 figma 에 이렇게 넣으면 모양... ㅠ

실행한 코드_txt

```
from selenium import webdriver # 크롬 드라이버를 사용하기 위한 import
from selenium.webdriver.by import By # 크롤드라이버로 선택지를 찾기 위한 import
from selenium.webdriver.common.keys import Keys # 키보드 키를 누르는 역할을 위한 import
import time # data 로딩을 위해 시간지연을 사용하기 위한 import
import pandas as pd # 형/열을 만들어 data 처리를 하기 위한 import, pandas 는 별도의 설치가 필요함 ( pip istall pandas 등 을 통해)

time.sleep(5) # 시간지연, 초 기준

names = [] # 크롤링 한 내용을 담을 배열
prices = [] # 크롤링 한 내용을 담을 배열

actions = driver.find_element(By.CSS_SELECTOR, 'body') # 키보드 키를 누르는 영역을 설정

last_height = driver.execute_script("return document.body.scrollHeight") # 현재 최대 스크롤값 측정 -> last_height

while (last_height <= 2000000): # 최대 스크롤값이 2 이하인 동안 반복 => 크롤링할 데이터량 조절

    time.sleep(1) # 데이터 로딩 을 위한 시간지연

    name = driver.find_elements(By.CLASS_NAME, "production-item__header__name") # Class Name 기준으로 element 가져옴
    price = driver.find_elements(By.CLASS_NAME, "production-item-price__price") # Class Name 기준으로 element 가져옴

    for i in name:
        names.append(i.text) # name 에 해당하는 text 를 names 배열에 담음
    for i in price:
        prices.append(i.text) # price 에 해당하는 text 를 prices 배열에 담음

    time.sleep(1) # 아무튼 시간지연

    actions.send_keys(Keys.PAGE_DOWN) # 바디 기준으로 'page_down' 키를 한번 누름 ( end, enter, spacebar 등 여러 키 로 변경해서 응용 사용 가능 )
    time.sleep(0.1) # 연속적인 키 입력시간 고려해서 시간지연 다음 0.1초로 해줌 그래도 1번 연속이 잘 안됨, 0.5 초 일때 정확히 원하는 동수만큼 가능했었음.
    actions.send_keys(Keys.PAGE_DOWN)
    time.sleep(0.1)
    actions.send_keys(Keys.PAGE_DOWN)
    time.sleep(0.1)
    actions.send_keys(Keys.PAGE_DOWN)
    time.sleep(0.1)
    actions.send_keys(Keys.PAGE_DOWN)
    # 총 5번 page_down 했는데, 원하는 갯수만큼 조절 가능, 1회 크롤링 후 임시로 2차 테스트했을때는 3번도 충분했었을것 같기도..
    # 이 역시 크롤링 및 파지지와 데이터 로딩 주변에 따라 변동됨.

    time.sleep(1) # 웹사 로드시 스크롤 다음 한 주제도 데이터 로딩을 위한 시간지연

    current_height = driver.execute_script("return document.body.scrollHeight") # 스크롤 후 현재 최대 스크롤값을 측정
    print(current_height) # 스크롤량이 잘되지만 현재 어디쯤이시 모르니, 콘솔을 통해 진행률 확인 (현재 스크롤값 확인)

    if current_height == last_height: # 위에서 정한 last_height 과 현재 스크롤 값이 같으면 탈출 // 지금 생각해보니 이미 while 문에서 조건을 걸어놔서 필요없을수도.
        break

    last_height = current_height # 정한 범위 안이면 현재 스크롤 값 = last_height 로 치환 // 이것도 상동. 다시 보니 원래 처면 다른 방법의 코드에서 방법을 바꾸며 지우다 남은 코드인듯함.

print("\n")
print("가져온 상품 수: ", len(names)) # 크롤링 한 총 data 량 알고싶어서 콘솔로 출력
print("\n")
df = pd.DataFrame({'상품 이름':names, '가격':prices}) # pandas 의 형/열 정렬 시스템을 이용해서 상품이름/가격 column 에 각각 names/prices 입력
print(df) # pandas가 제대로 먹히는지 확인하고싶어서 콘솔로 추력
df.to_csv("01_shop_furniture.csv", index = False, encoding='cp949')
# DataFrame 인 기둥 중 하나로, 정렬로 된 data를 csv 파일로 저장 가능, 이 때 UTF-8 로 하면 깨짐, cp949 로 인코딩 후 oracle sql 에 넣을 때 다시 utf-8 로 인코딩 후 import 하면 됨.
# 또는 다른 좋은 방법이 있다면 그것도 추천

driver.quit() # 크롬드라이버 실행 중지
print("\n")
print('save completed') # 모든 기능이 끝났을 알기 위해 콘솔로 출력
```

코드 실행 시 권한단 오류

문제해결 방법은 간단해요 ! 실행하기 전에 사전 입니다.

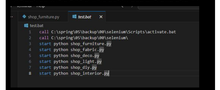
1. 관리자로 바탕로 가장 후 cmd 명령을 만드는데, 꼭 한 한개의 같은 폰에 할 때만 고려가 나며 그 화면만 cmd 가 다 남기고, ==== 그냥 여러대에 했어하도, 상관 없음 ㅎ

2. 정해진 scrol height 범위지지 기지 방법는는 먹지 않한이 굉상, 이유는 첫 첫줄.

3. 맞 가동에다가 갖다가 개발, 이유는 첫 첫줄.

4. 위정보에서 여러 개 필요한 bat 파일로 같이 돌린다니 6개 중 3개가 일하도 도와만 일하도 도와만 일하도 하다가 5개중 1개 중간에 혼자 하지고 이러저 ... 그래서 그냥 하나씩 정작하게 돌릴게 ㅋㅋ

이러니 경험해 면은 bat 파일 (배치파일) 로 자동 실행하기 / 실행한 뱃그나 편하는 실행법 ㅋㅋ



파일 명시문 (이름).bat

call C:\~selenium\Scripts\activate.bat
: Scripts 경로로 이동하여 selenium 실행하기

call C:\~selenium
: py 파일들이 있는 selenium 폴더로 이동
(1.단 selenium 폴더에 py 파일과 chromedriver.exe 파일을 같이 두었음.)

start python ~.py
: ~.py 를 python 으로 실행하기

bat 파일 이름, 등등 하면 여러 키입문 일도 좋을 수 있을것 같기도 하고, 해있는 것도 만들 수 있을 듯