



분석용 데이터

획득 전략

웹 데이터 확보를 위한 환경설정

학습 목표

+ + +

학습 목표

- 웹 크롤링의 원리를 이해하고 설명할 수 있다.
- 웹 크롤링을 진행하기 위한 환경 설정을 할 수 있다.
- 웹 페이지 접속과 검색을 자동으로 구현할 수 있다.

학습 내용

- 웹 크롤링의 원리 및 환경 설정
- 웹 페이지 자동 접속 및 자동 검색 구현

웹 크롤링의 원리 이해

1) 웹 크롤링이란?



- 변화는 환경 속에 **원리**를 정확히 인지해야 함
- 셀레니움(Selenium)
 - 파이썬(Python), 자바(Java), 알(R)이나 이런 프로그램을 이용해서 이렇게 명령어를 사용
 - 셀레니움이 사용하는 웹 브라우저가 있어야 됨
- 웹 드라이버(Web Driver)
 - 셀레니움이 사용하는 웹 브라우저

웹 크롤링의 원리 이해

1) 웹 크롤링이란?



■ 뷰티풀 스프(Beautiful Soup)

- 셀레니움이 가져온 전체데이터에서 필요한데이터를 뽑아내는 모듈
- 텍스트 파일(Text File), 엑셀 파일(Excel File) 등 여러 가지 형식으로 데이터 저장

SELENIUM

사람을 대신해서 실제 작업 수행 모듈

Web Driver

Selenium 이 사용하는 웹 브라우저

Beautiful Soup

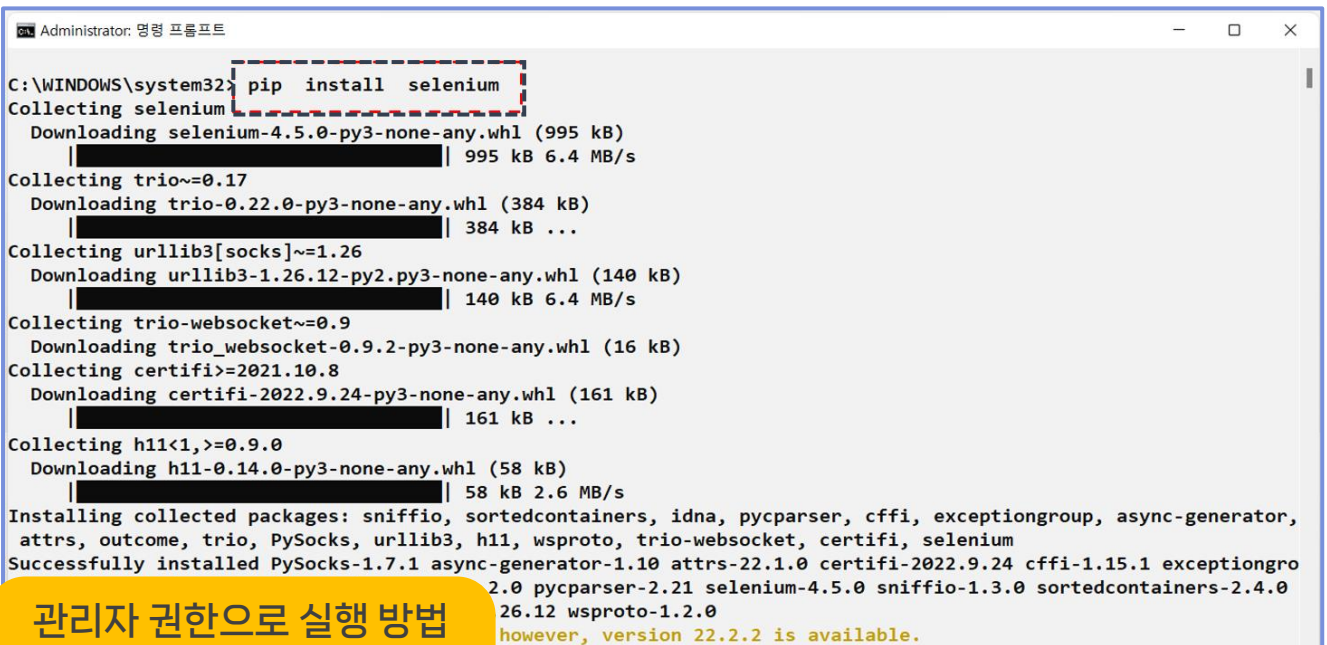
사람이 원하는 데이터만 추출하는 모듈

웹 크롤링을 위한 환경 설정

1) 웹 크롤링 환경 설정

(1) Selenium 모듈 설치

- Selenium 설치 시 권한 에러가 발생할 수 있으므로 cmd(명령 프롬프트)창을 **관리자 권한**으로 실행해야 함



```
Administrator: 명령 프롬프트

C:\WINDOWS\system32> pip install selenium
Collecting selenium
  Downloading selenium-4.5.0-py3-none-any.whl (995 kB)
    |#####| 995 kB 6.4 MB/s
Collecting trio~=0.17
  Downloading trio-0.22.0-py3-none-any.whl (384 kB)
    |#####| 384 kB ...
Collecting urllib3[socks]~=1.26
  Downloading urllib3-1.26.12-py2.py3-none-any.whl (140 kB)
    |#####| 140 kB 6.4 MB/s
Collecting trio-websocket~=0.9
  Downloading trio_websocket-0.9.2-py3-none-any.whl (16 kB)
Collecting certifi>=2021.10.8
  Downloading certifi-2022.9.24-py3-none-any.whl (161 kB)
    |#####| 161 kB ...
Collecting h11<1,>=0.9.0
  Downloading h11-0.14.0-py3-none-any.whl (58 kB)
    |#####| 58 kB 2.6 MB/s
Installing collected packages: sniffio, sortedcontainers, idna, pycparser, cffi, exceptiongroup, async-generator,
attrs, outcome, trio, PySocks, urllib3, h11, wsproto, trio-websocket, certifi, selenium
Successfully installed PySocks-1.7.1 async-generator-1.10 attrs-22.1.0 certifi-2022.9.24 cffi-1.15.1 exceptiongro
2.0 pycparser-2.21 selenium-4.5.0 sniffio-1.3.0 sortedcontainers-2.4.0
26.12 wsproto-1.2.0
however, version 22.2.2 is available.
```

관리자 권한으로 실행 방법

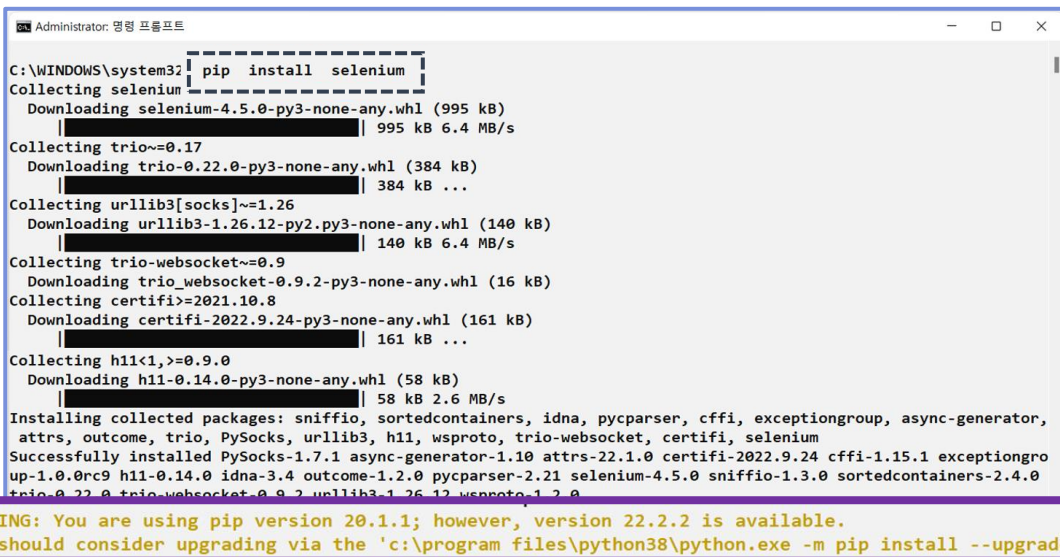
마우스 오른쪽 버튼을 눌러 실행

웹 크롤링을 위한 환경 설정

1) 웹 크롤링 환경 설정

(1) Selenium 모듈 설치

- **빨간색** 메시지 - 문제 발생
- **노란색** 메시지 - 경고



```
Administrator: 명령 프롬프트
C:\WINDOWS\system32 [pip install selenium]
Collecting selenium
  Downloading selenium-4.5.0-py3-none-any.whl (995 kB)
    | 995 kB 6.4 MB/s
Collecting trio~=0.17
  Downloading trio-0.22.0-py3-none-any.whl (384 kB)
    | 384 kB ...
Collecting urllib3[socks]~=1.26
  Downloading urllib3-1.26.12-py2.py3-none-any.whl (140 kB)
    | 140 kB 6.4 MB/s
Collecting trio-websocket~=0.9
  Downloading trio_websocket-0.9.2-py3-none-any.whl (16 kB)
Collecting certifi>=2021.10.8
  Downloading certifi-2022.9.24-py3-none-any.whl (161 kB)
    | 161 kB ...
Collecting h11<1,>=0.9.0
  Downloading h11-0.14.0-py3-none-any.whl (58 kB)
    | 58 kB 2.6 MB/s
Installing collected packages: sniffio, sortedcontainers, idna, pycparser, cffi, exceptiongroup, async-generator,
attrs, outcome, trio, PySocks, urllib3, h11, wsproto, trio-websocket, certifi, selenium
Successfully installed PySocks-1.7.1 async-generator-1.10 attrs-22.1.0 certifi-2022.9.24 cffi-1.15.1 exceptiongro
up-1.0.0rc9 h11-0.14.0 idna-3.4 outcome-1.2.0 pycparser-2.21 selenium-4.5.0 sniffio-1.3.0 sortedcontainers-2.4.0
trio-0.22.0 trio-websocket-0.9.2 urllib3-1.26.12 wsproto-1.2.0

WARNING: You are using pip version 20.1.1; however, version 22.2.2 is available.
You should consider upgrading via the 'c:\program files\python38\python.exe -m pip install --upgrade pip' command
```

웹 크롤링을 위한 환경 설정

1) 웹 크롤링 환경 설정

(1) Selenium 모듈 설치



```
Administrator: 명령 프롬프트
C:\WINDOWS\system32
Collecting selenium
  Downloading selenium-4.5.0-py3-none-any.whl (995 kB)
    | 995 kB 6.4 MB/s
Collecting trio~=0.17
  Downloading trio-0.22.0-py3-none-any.whl (384 kB)
    | 384 kB ...
Collecting urllib3[socks]~=1.26
  Downloading urllib3-1.26.12-py2.py3-none-any.whl (140 kB)
    | 140 kB 6.4 MB/s
Collecting trio-websocket~=0.9
  Downloading trio_websocket-0.9.2-py3-none-any.whl (16 kB)
Collecting certifi>=2021.10.8
  Downloading certifi-2022.9.24-py3-none-any.whl (161 kB)
    | 161 kB ...
Collecting h11<1,>=0.9.0
  Downloading h11-0.14.0-py3-none-any.whl (58 kB)
    | 58 kB 2.6 MB/s
Installing collected packages: sniffio, sortedcontainers, idna, pycparser, cffi, exceptiongroup, async-generator,
  attrs, outcome, trio, PySocks, urllib3, h11, wsproto, trio-websocket, certifi, selenium
Successfully installed PySocks-1.7.1 async-generator-1.10 attrs-22.1.0 certifi-2022.9.24 cffi-1.15.1 exceptiongro
up-1.0.0rc9 h11-0.14.0 idna-3.4 outcome-1.2.0 pycparser-2.21 selenium-4.5.0 sniffio-1.3.0 sortedcontainers-2.4.0
trio-0.22.0 trio-websocket-0.9.2 urllib3-1.26.12 wsproto-1.2.0
WARNING: You are using pip version 20.1.1; however, version 22.2.2 is available.
```

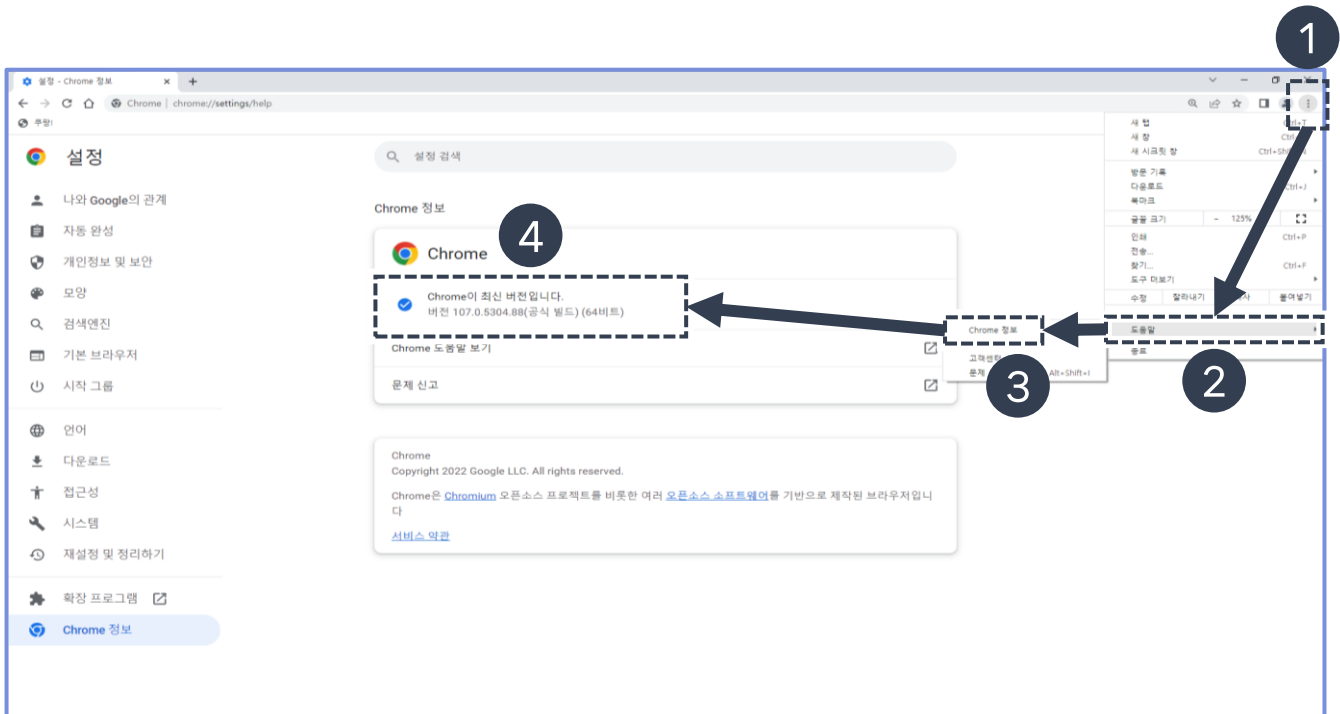
아나콘다(Anaconda) 사용자는 아나콘다 프롬프트를 이용

웹 크롤링을 위한 환경 설정

1) 웹 크롤링 환경 설정

(2) Web Driver 설치

- Web Driver는 사람이 사용하는 크롬, 에지, 사파리, 파이어폭스 등을 다 사용할 수 있음
- Chrome driver를 사용하기 위해서는 먼저 컴퓨터에 사람이 사용하는 Google Chrome 브라우저가 설치되어 있어야 함
- 사람이 사용하는 Chrome 버전과 Chrome driver 버전이 동일해야 함



강의를 듣는 시점에 따라 버전이 다를 수 있습니다.

웹 크롤링을 위한 환경 설정

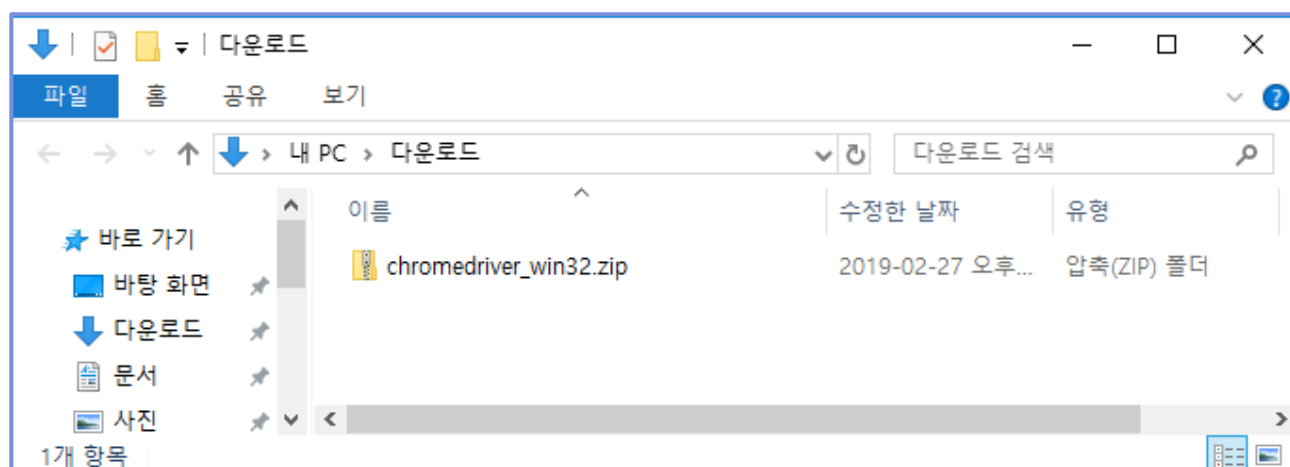
1) 웹 크롤링 환경 설정

(2) Web Driver 설치

- linux64.zip – 리눅스 OS
- mac64.zip, mac arm64.zip – 맥 OS
- win32.zip – 윈도우 OS

Index of /107.0.5304.62/

Name	Last modified	Size	ETag
Parent Directory	-	-	-
chromedriver_linux64.zip	2-10-25 12:20:56	7.26MB	90d3353f17fcbd755626d528e94a1d9a
chromedriver_mac64.zip	2-10-25 12:20:57	8.41MB	652c969a3b8d47e7fa9518d90b411fba
chromedriver_mac_arm64.zip	2-10-25 12:20:59	7.72MB	dba9920d41a8ec9fb847326ae0f68200
chromedriver_win32.zip	2-10-25 12:21:00	6.46MB	a5040d2731fe174c9a7b026edb3fe271
notes.txt	2-10-25 12:21:05	0.00MB	936b74dab32b11addaffb0a624d9894a

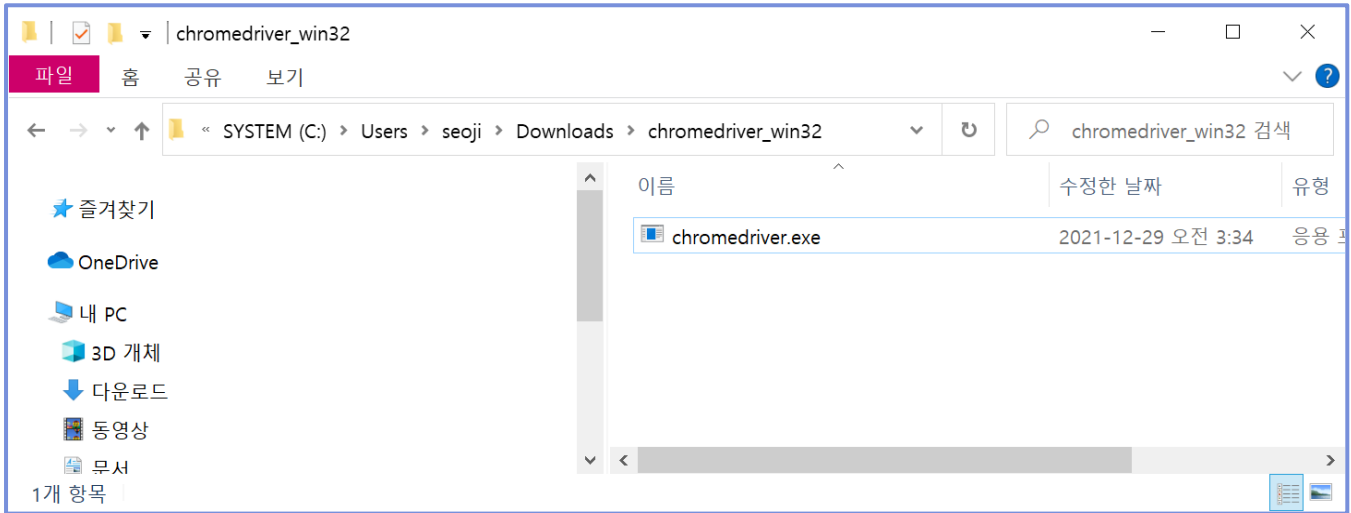


- 위 파일을 적당한 곳에 압축을 풀

웹 크롤링을 위한 환경 설정

1) 웹 크롤링 환경 설정

(2) Web Driver 설치

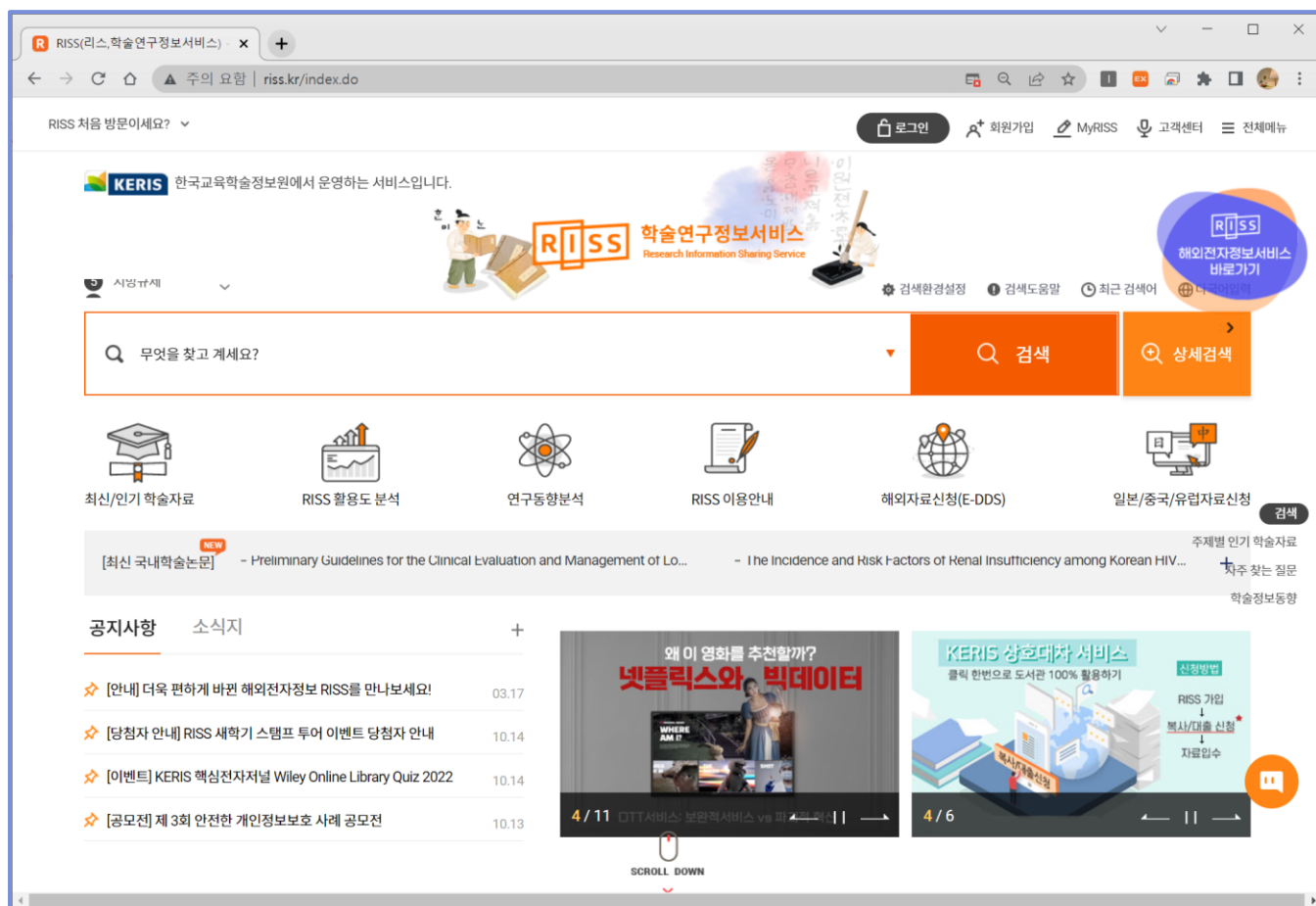


- 실습의 편의상 압축을 풀어서 생긴 chromedriver.exe 파일을 c:\py_temp 폴더를 생성한 후 그 폴더 안으로 복사

웹 페이지 자동 접속 및 자동 검색 구현

1) 웹 페이지 자동 접속

- 학술연구정보서비스 사이트를 이용하여 웹 크롤링 실습



웹 페이지 자동 접속 및 자동 검색 구현

1) 웹 페이지 자동 접속

```
File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel) C
[+] [x] [f] [p] [u] [d] [Run] [Stop] [C] [Code] [v] [m]

In [ ]: 1 #Step 1. 필요한 모듈을 로딩합니다
2 from selenium import webdriver
3 from selenium.webdriver.common.by import By
4 from selenium.webdriver.common.keys import Keys
5 from selenium.webdriver.chrome.service import Service
6 import time
7
8 #Step 2. 사용자에게 검색 관련 정보들을 입력 받습니다.
9 print("-" * 100)
10 print(" 이 크롤러는 riss 사이트의 논문 자료 수집용 웹크롤러입니다.")
11 print("-" * 100)
12 query_txt = input('1. 수집할 자료의 키워드는 무엇입니까?(예: 해양자원): ')
13 print("\n")
14
15 #Step 3. 크롬 드라이버 설정 및 웹 페이지 열기
16 # 크롬 드라이버 설정하는 최신 문법
17 s = Service("c:/py_temp/chromedriver.exe")
18 driver = webdriver.Chrome(service=s)
19
20 url = 'https://www.riss.kr/'
21 driver.get(url)
22 time.sleep(5)
23 driver.maximize_window()
24
25 #Step 4. 자동으로 검색어 입력 후 조회하기
26 element = driver.find_element(By.ID, 'query')
27 driver.find_element(By.ID, 'query').click()
28 element.send_keys(query_txt)
29 element.send_keys("\n")
```

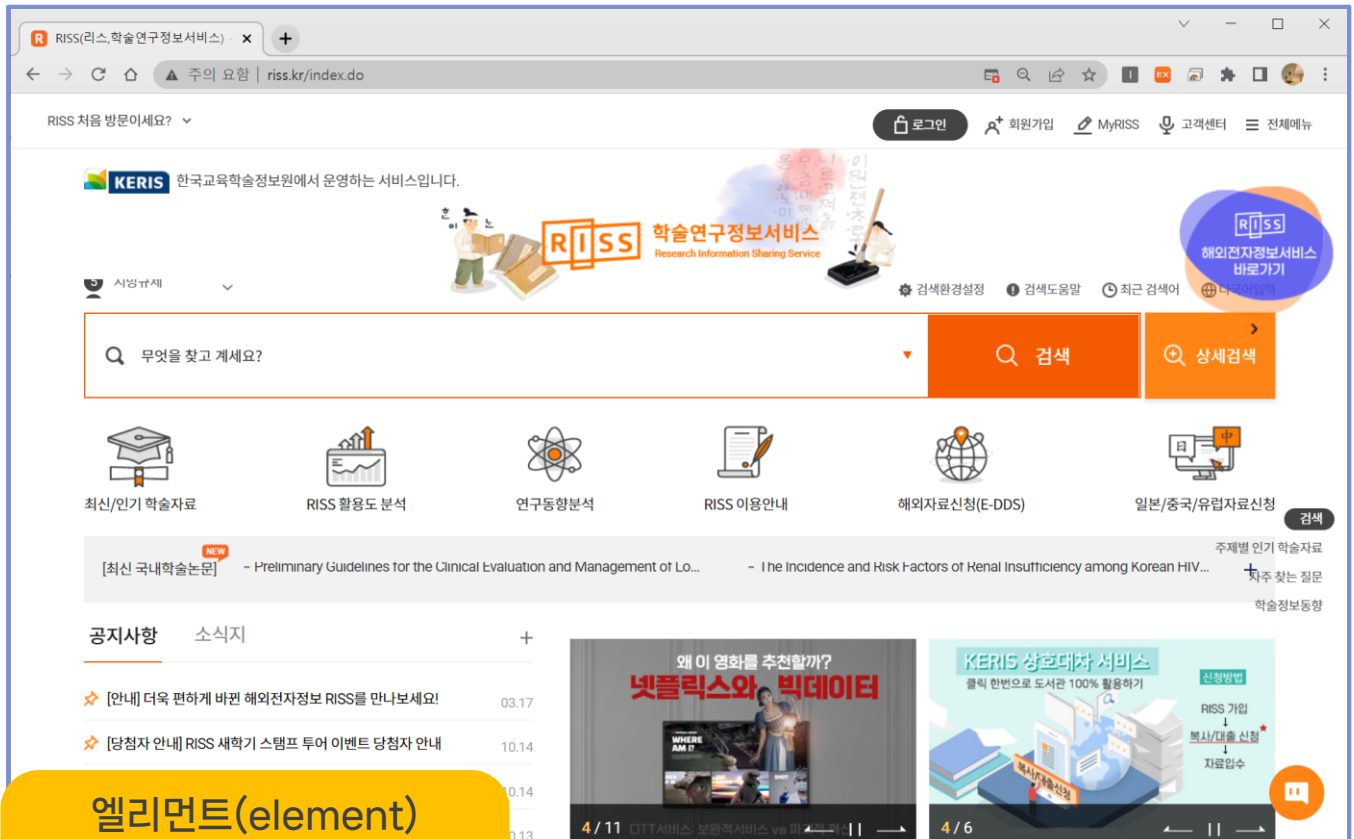
주피터 노트북(Jupyter Notebook)을 이용

```
In [ ]: 1 #Step 1. 필요한 모듈을 로딩합니다
2 from selenium import webdriver
3 from selenium.webdriver.common.by import By
4 from selenium.webdriver.common.keys import Keys
5 from selenium.webdriver.chrome.service import Service
6 import time
7
8 #Step 2. 사용자에게 검색 관련 정보들을 입력 받습니다.
9 print("=" * 100)
10 print(" 이 크롤러는 riss 사이트의 논문 자료 수집용 웹크롤러입니다.")
11 print("=" * 100)
12 query_txt = input('1. 수집할 자료의 키워드는 무엇입니까?(예: 해양자원): ')
13 print("\n")
14
15 #Step 3. 크롬 드라이버 설정 및 웹 페이지 열기
16 # 크롬 드라이버를 실행하는 확실한 방법
17 s = Service("c:/py_temp/chromedriver.exe")
18 driver = webdriver.Chrome(service=s)
19
20 url = 'https://www.riss.kr/'
21 driver.get(url)
22 time.sleep(5)
23 driver.maximize_window()
24
25 #Step 4. 자동으로 검색어 입력 후 조회하기
26 element = driver.find_element(By.ID, 'query')
```

chromedriver.exe 파일 경로를 적음

웹 페이지 자동 접속 및 자동 검색 구현

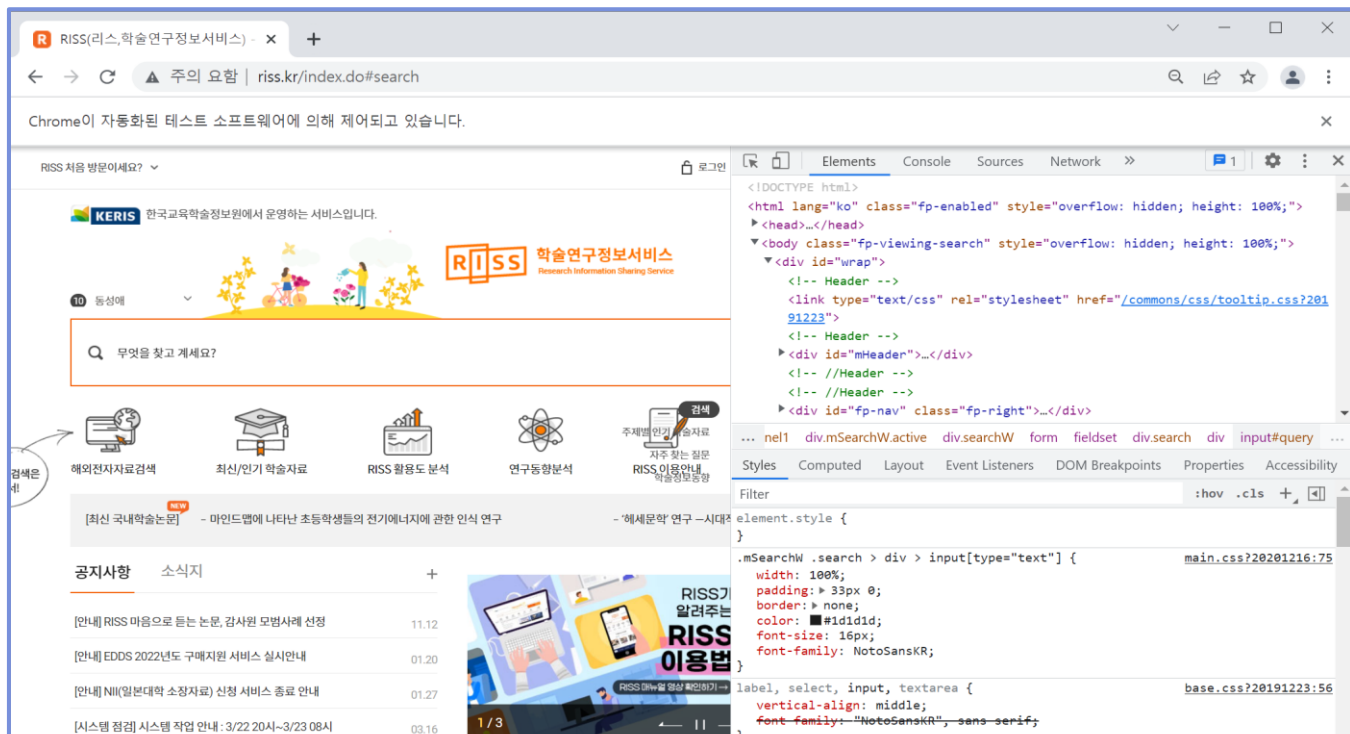
2) 자동 검색 구현



웹사이트를 구성하는 하나하나의 요소

웹 페이지 자동 접속 및 자동 검색 구현

2) 자동 검색 구현



Chrome 개발자 도구 F12

웹 페이지 자동 접속 및 자동 검색 구현

2) 자동 검색 구현

1 클릭

2 클릭

3

```
<input type="hidden" name="oldQuery" value>
<!-- //통학검색 form 해외자료 검색 용-->
<input type="hidden" name="sflag" value="1">
<input type="hidden" name="fsearchMethod" value="sea
h">
<input type="hidden" name="isFDetailSearch" value="N
">
<input type="hidden" name="searchQuery" value>
<input type="hidden" name="kbid">
<input type="hidden" name="pageNumber" value="1">
<fieldset>
  <legend>검색</legend>
  <ul class="searchInfo globalMenu">...</ul>
  <div class="search">
    <div>
      <input type="text" id="query" name="query"
        placeholder="무엇을 찾고 계세요?" title="검색"
        onkeydown="javascript:setFrameEvent(event);"
        onkeyup="javascript:getAutoQuery(this.value,ev
t);" onblur="javascript:onBlurCheck();document
etElementById('lastFocusName').value='query';"
        onkeypress="javascript:trick();" tabindex="22">
```

id = "query" 또는 name = "query" 둘 중 하나를 이용

2) 자동 검색 구현

