

# 공공빅데이터분석PBL

## [ 웹크롤링 ]

Personal Information Protection Theory

## 셀레니움

### ▪ 자동화 툴 이해하기

- ✓ 자동화 툴은 사람을 대신하여 반복적이고 연속적인 작업을 수행하는 프로그램

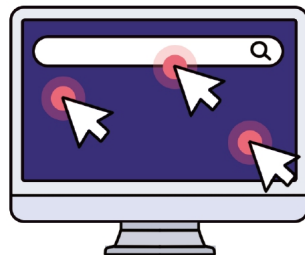


그림 4-1 자동화 툴

- 셀레니움(Selenium)은 사람을 대신하여 자동으로 웹 브라우저에서 동작을 수행하는 프레임워크.
- 셀레니움으로 수행할 수 있는 동작은 다양함.

## 셀레니움

### ■ 셀레니움 기본 사용법

#### ✓ 동적인 웹페이지와 셀레니움

- 정적인 페이지(Static web page)는 서버에 저장된 데이터를 그대로 보여주어 시간이 지나도 모습이 변하지 않음.



그림 4-2 정적인 페이지

## 셀레니움

### ■ 셀레니움 기본 사용법

#### ✓ 동적인 웹페이지와 셀레니움

- 동적인 페이지(Dynamic web page)는 사용자가 클릭하거나 텍스트를 입력하여 페이지의 상태를 바꿀 수 있음.

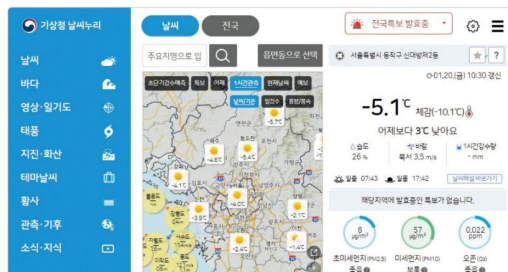


그림 4-3 동적인 페이지

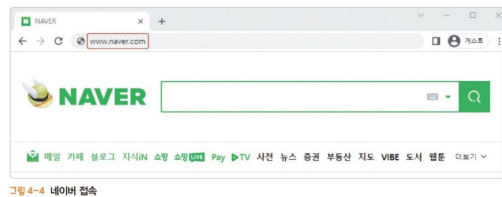
- 셀레니움을 이용하면 동적인 페이지의 데이터를 수집하는 데 시간과 노동력을 절감할 수 있음.

## 셀레니움

### ■ 셀레니움 기본 사용법

#### ✓ 셀레니움 명령어

- 웹페이지 접속: 웹 브라우저 주소창에 `www.naver.com`을 입력하고 Enter를 입력하는 동작



- » 셀레니움에서는 이 동작을 코드 단 한 줄로 표현.

```
driver.get([URL])
```

- » `get()` 함수는 특정 웹페이지에 접속하는 함수.

## 셀레니움

### ■ 셀레니움 기본 사용법

#### ✓ 셀레니움 명령어

- 특정 위치의 텍스트를 찾아서 수집: '대한민국 정책브리핑' 페이지에 접속하여 주요 뉴스 제목을 수집.



- » `find_element()`는 화면 상의 버튼, 텍스트, 리스트 등 객체를 찾음.

```
driver.find_element('xpath', '[실제 XPath 값]').text
```

## 셀레니움

### ■ 셀레니움 기본 사용법

#### ✓ 셀레니움 명령어

- 특정 위치 클릭: 도정이 웹페이지에서 버튼을 클릭하는 동작

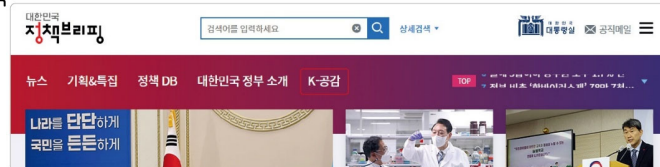


그림 4-6 동적인 웹페이지에서 버튼 클릭

```
driver.find_element('xpath', '[실제 XPath 값]').click( )
» 클릭( ) 함수를 호출하기 위해 클릭할 위치를 입력.
```

## 셀레니움

### ■ 셀레니움 기본 사용법

#### ✓ 셀레니움 명령어

- 특정 위치에 텍스트 입력: 검색어 입력란에 텍스트를 입력하는 동작

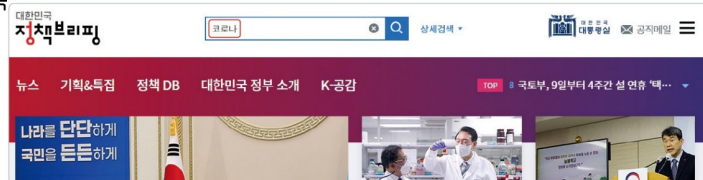


그림 4-7 웹페이지에서 특정 위치에 키 입력

```
driver.find_element('xpath', '[실제 XPath 값]').send_keys('텍스트')
```

- » 문자열을 입력하는 send\_keys( ) 함수를 호출
- » send\_keys('코로나')와 같이 명령.

## 셀레니움 웹 크롤링

### ■ 셀레니움 설치와 실행

- ✓ 셀레니움을 실행하려면 셀레니움이 조작할 가상의 웹 브라우저를 연동해야 함.

크롤링: 소프트웨어를 통해 인터넷 웹페이지를 돌아다니며 정보를 수집하는 일.

크롤러(Crawler): 이러한 작업을 수행하는 소프트웨어.

## 셀레니움 웹 크롤링

### ■ 셀레니움 설치와 실행

- ✓ 셀레니움을 실행하려면 셀레니움이 조작할 가상의 웹 브라우저를 연동해야 함.

[코드 4-1] files 라이브러리

```
import sys

!sudo add-apt-repository ppa:saiancot895/chromium-beta
#실행 결과에서 Enter 입력
!sudo apt remove chromium-browser
!sudo snap remove chromium
!sudo apt install chromium-browser

!pip3 install selenium
!apt-get update
!apt install chromium-chromedriver
!cp /usr/lib/chromium-browser/chromedriver /usr/bin

sys.path.insert(0, '/usr/lib/chromium-browser/chromedriver')
```

— 3행에서 설치가 중지됨. 실행 결과 맨 아래의 입력 양식에서 Enter를 눌러 진행.

## 셀레니움 웹 크롤링

### ■ 셀레니움 설치와 실행

- ✓ 셀레니움을 실행하려면 셀레니움이 조작할 가상의 웹 브라우저를 연동해야 함.  
[코드 4-2] 관련 라이브러리 가져오기

```
from selenium import webdriver
from selenium.webdriver.common.keys import Keys
```

- import 명령어를 사용하여 모듈을 이 코드에서 사용할 수 있도록 가져옴.

[코드 4-3] 구글 Colab 환경에 맞게 셀레니움 사용 설정

```
options = webdriver.Chrome.Options()
options.add_argument('--headless') #창이 나타나지 않도록 Headless 설정하기
options.add_argument('--no-sandbox')
options.add_argument('--disable-dev-shm-usage')
driver = webdriver.Chrome(options=options)
```

- 마지막 행에서는 가상의 웹 브라우저에서 창을 열고 이 객체를 변수 driver에 할당.

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

- ✓ 셀레니움으로 웹페이지에 접속하기

- 특정 URL에 접속(코로나 현황 웹페이지 → 기상청 날씨누리 [www.weather.go.kr](http://www.weather.go.kr))



그림 4-9 질병관리청 코로나바이러스감염증-19 현황 웹페이지

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

- ✓ 셀레니움으로 웹페이지에 접속하기

[코드 4-4] 변수 url에 웹페이지의 URL 대입

```
url = 'http://ncov.kdca.go.kr/'
driver.get(url)
```

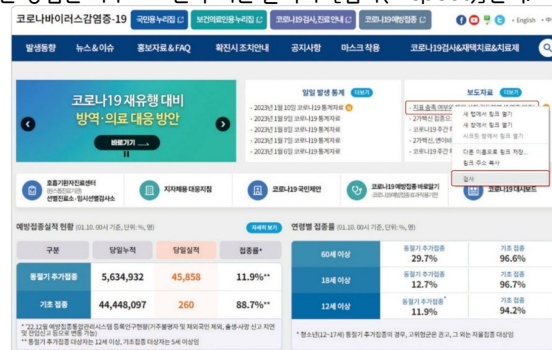
- 변수 url에 코로나 현황 페이지의 주소를 문자열로 대입.
- 변수 url을 driver.get() 함수의 인자로 넣어 해당 웹페이지에 접속

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

- ✓ 최신 뉴스 수집하기

- 특정 위치의 텍스트를 수집할 때 XPath를 사용.
- XPath(XML Path Language)란 웹페이지를 설계할 때 사용한 언어 구조로 위치를 특정하는 방식, 알아내는 방법은 마우스 오른쪽 버튼 클릭 후 [검사(Inspect)]선택.



구분	당일누적	당일상회	당일사망
누적	5,634,932	45,858	11.9%*
당일	44,448,097	260	88.7%*

\* 2020년 12월 21일 기준 누적치로, 2020년 12월 21일 기준 누적치로, 2020년 12월 21일 기준 누적치로

그림 4-10 XPath 알아내기 (1)

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 최신 뉴스 수집하기

- 개발자 도구에는 웹페이지를 구성하고 개발할 때 사용한 소스 코드가 나타남.

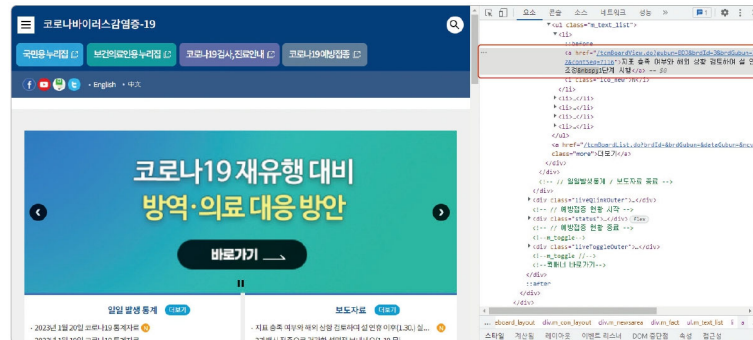


그림 4-11 XPath 알아내기 (2)

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 최신 뉴스 수집하기

- 음영 표시된 코드를 마우스 오른쪽 버튼으로 클릭하고, 메뉴에서 [복사]를 선택하여 XPath를 복사.

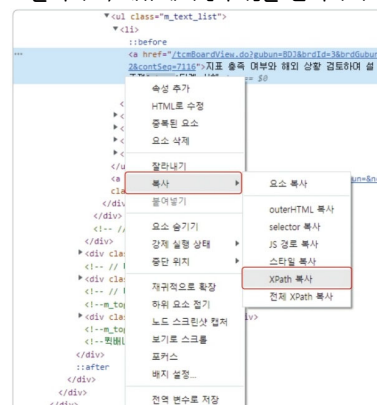


그림 4-12 XPath 알아내기 (3)



## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 최신 뉴스 수집하기

- 메모장에 붙여넣어 확인해 보면 XPath는 다음과 같은 모습.

```
//*[@id="content"]/div/div/div/div[1]/div[2]/div/ul/li[1]/a
```

- 셀레니움에서 복사한 XPath를 그대로 입력하기만 하면 해당 위치에 있는 텍스트를 수집할 수 있음.

```
driver.find_element('xpath', '[실제 XPath 값]')
```

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 최신 뉴스 수집하기

[코드 4-5] 기사 제목 객체

```
topnews = driver.find_element('xpath', '//*[@id="content"]/div/div/div/div[1]/\n\n                div[2]/div/ul/li[1]/a')\n\nprint(topnews.text)
```

[코드 4-5] 실행결과

실외 마스크 착용 자율 전환 및 전국단위 코로나19 항체양성률 조사 결과 발표

- find\_element() 함수가 XPath에 해당하는 객체를 반환하면 객체를 topnews 변수에 넣기.
- topnews 변수에 담은 객체의 텍스트를 가져오기 위해 text 속성을 호출. 그리고 반환되는 값을 print() 함수로 출력하면 뉴스 제목이 출력됨.

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

✓ 여러 개의 최신 뉴스를 한꺼번에 수집하기

- [보도자료] 구역에 뉴스가 총 5개 노출됨.

객체 5개의 XPath 값을 각각 복사하고 메모장에 붙여넣기.

```
//*[@id="content"]/div/div/div/div[1]/div[2]/div/ul/li[1]/a
//*[id="content"]/div/div/div/div[1]/div[2]/div/ul/li[2]/a
//*[id="content"]/div/div/div/div[1]/div[2]/div/ul/li[3]/a
//*[id="content"]/div/div/div/div[1]/div[2]/div/ul/li[4]/a
//*[id="content"]/div/div/div/div[1]/div[2]/div/ul/li[5]/a
```

- ul까지 구문이 모두 일치하는 점을 이용하면 객체 여러 개의 텍스트를 한꺼번에 수집할 수 있음.

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

✓ 여러 개의 최신 뉴스를 한꺼번에 수집하기

[코드 4-6] 여러 개의 텍스트를 한꺼번에 수집

```
topnews = driver.find_elements('xpath', '//*[id="content"]/div/div/div/div[1]/div[2]/div/ul')

#여러 개의 텍스트를 리스트 topnews에 정리하기
topnews = [topnew.text for topnew in topnews]

print(topnews)
```

[코드 4-6] 실행결과

```
[ '[카드뉴스] 신속한 치료 지원을 위해 일반병상 지속 확보'\n「실내 마스크 의무 조정 등 향후 코로나19 대응 방향」 관련 전문가 토론회 개최 (12.15.목)\n동절기 접종률 제고를 위한 일선 의료현장 소통 강화(12.15.목)\n코로나19 주간 확진자 전주 대비\n11.2%\n증가(12.14.)\n12월\n12일부터\n12-17세 청소년 대상 동절기 추가접종 시작 ' ]
```

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 버튼 클릭하기

- 코로나 현황 페이지에서 뉴스를 검색하려면 먼저 돋보기 버튼을 클릭하여 검색어 입력란을 나타내야 함.



그림 4-13 돋보기 버튼

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 버튼 클릭하기

- 셀레니움에서도 돋보기 버튼을 찾아 클릭하는 작업을 자동화할 수 있음. 개발자 도구에서 돋보기 버튼의 XPath를 복사.

```
//*[@id="header"]/div/div[2]/a[1]
```

[코드 4-7] 버튼 객체 클릭

```
button = driver.find_element('xpath', '//*[@id="header"]/div/div[2]/a[1]')
print(button.text)
button.click()
```

[코드 4-7] 실행결과

- 통합검색
- XPath를 기준으로 찾은 돋보기 버튼 객체를 변수 button에 담기.
- 돋보기 버튼 객체에 click() 함수를 덧붙여 객체를 클릭.

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 텍스트 입력하기

- 코로나 현황 페이지 첫 화면에서 돋보기 버튼을 클릭하면 다음 그림처럼 검색어 입력란과 검색 버튼이 나란히 나타남.

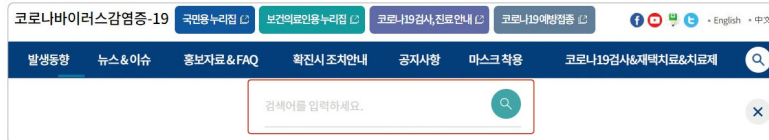


그림 4-14 검색어 입력란과 검색 버튼

- 클릭해야 하는 위치, 즉 검색어 입력란의 XPath를 개발자 모드에서 복사하여 진행.

```
//*[@id="searchTermMobile123"]a
```

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 텍스트 입력하기

[코드 4-8] 검색어 입력

```
driver.find_element('xpath', '//*[@id="searchTermMobile123"]').click()
driver.find_element('xpath', '//*[@id="searchTermMobile123"]').send_keys('서울')

driver.find_element('xpath', '//*[@id="searchTermMobile123"]').send_keys(Keys.ENTER)
```

- 객체에 무언가를 입력할 때는 send\_keys() 함수를 사용.  
Keys.ENTER를 send\_keys() 함수의 인자로 넣으면 [Enter]를 입력하라는 명령.
- [코드 4-7]과 [코드 4-8]의 검색 작업은 돋보기 버튼을 클릭하고 검색어 입력란을 클릭한 다음 검색어를 입력하고 [Enter]를 입력하는 네 단계로 구분.

```
driver.find_element('xpath', '//*[@id="header"]/div/div[2]/a[1]').click()
driver.find_element('xpath', '//*[@id="searchTermMobile123"]').click()
driver.find_element('xpath', '//*[@id="searchTermMobile123"]').send_keys('서울')
driver.find_element('xpath', '//*[@id="searchTermMobile123"]').send_keys(Keys.ENTER)
```

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 텍스트 입력하기

- 코드에서 반복되는 부분을 변수 serch\_for에 담으면 코드를 보다 간결하게 줄일 수 있음.

```
driver.find_element('xpath', '//*[@id="header"]/div/div[2]/a[1]').click()

search_for =
driver.find_element('xpath', '//*[@id="searchTermMobile123"]')
search_for.click()
search_for.send_keys('서울')
search_for.send_keys(Keys.ENTER)
```

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

#### ✓ 텍스트 입력하기

- 여기까지 실행하면 '서울' 검색 결과 페이지로 이동

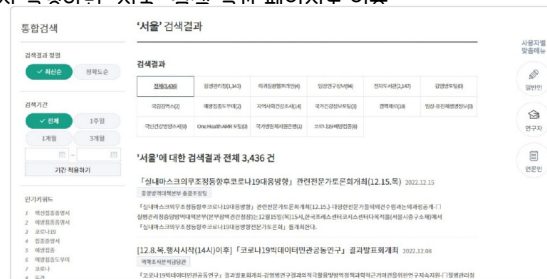


그림 4-15 검색 결과

- page\_source 속성을 출력하면 소스 코드를 보고 간접적으로 현재 페이지 확인.

#### [코드 4-9] 페이지 소스 코드 확인

```
print(driver.page_source)
```

## 셀레니움 웹 크롤링

### ■ 셀레니움 웹 크롤링

✓ 하나 더 알기: 셀레니움 함수

기능	형식
뒤로	driver.back()
앞으로	driver.forward()
새로고침	driver.refresh()
탭 닫기	driver.close()
창 닫기	driver.quit()
창 최대화	driver.maximize_window()
창 최소화	driver.minimize_window()
브라우저 HTML 정보 출력	print(driver.page_source)

## 코로나 발생현황 데이터 수집

### [문제]

코로나 현황 페이지에는 발생현황이 정리되어 있습니다. 2023년 7월 기준으로 발생현황 데이터는 일주일마다 한 번씩 7일간의 일평균이 업데이트됩니다. 확진자 수 데이터만 수집하는 웹 크롤러를 작성해 봅시다.

발생현황 (6.19. 00시 기준, '20.1.3. 이후 누계, 단위: 명)

자세히보기

구분	사망	재원 위중증	확진
최근 7일간 일평균	9	111	16,271
(누적)사망 34,960	(누적)확진 32,018,486		다운로드

발생현황과 다운로드 통계는 역학조사 결과의 세부 내용에 따라 일시적으로 다를 수 있음

## 코로나 발생현황 데이터 수집

### [해결]

지금까지 배운 명령어를 활용하여 코로나 현황 페이지에서 데이터를 수집하고, 미리 만들어둔 템플릿에 대입하겠습니다.

```
기준일자 :
일평균 사망자 수 :
일평균 재원 위중증 환자 수 :
일평균 확진자 수 :
```

1. 발생현황 기준일자, 사망자 수, 재원 위중증 환자 수, 확진자 수를 수집할 웹페이지 URL을 입력하여 접속하는 것으로 시작.

```
url = 'https://ncov.kdca.go.kr/'
driver.get(url)
```

## 코로나 발생현황 데이터 수집

### [해결]

2. 객체 4개에 있는 4가지 수를 수집하려면 코드도 4개 필요. 객체의 XPath를 구하여 찾고, 차례로 변수 first\_blank, second\_blank, third\_blank, fourth\_blank에 담기.

```
#기준일자
first_blank = driver.find_element('xpath', '//*[@id="content"]/div/div/div/\
div[3]/div/div[1]/div[1]/h2/span').text

#일평균 사망자 수
second_blank = driver.find_element('xpath', '//*[@id="content"]/div/div/div/\
div[3]/div/div[1]/div[1]/div[1]/table\
tbody/tr/td[1]/span').text

#일평균 재원 위중증 환자 수
third_blank = driver.find_element('xpath', '//*[@id="content"]/div/div/div/\
div[3]/div/div[1]/div[1]/div[1]/table/\
tbody/tr/td[2]/span').text

#일평균 확진자 수
fourth_blank = driver.find_element('xpath', '//*[@id="content"]/div/div/div/\
div[3]/div/div[1]/div[1]/div[1]/table/\
tbody/tr/td[3]/span').text
```

## 코로나 발생현황 데이터 수집

[해결]

3. 변수에 담은 텍스트를 출력하여 확인.

```
print('기준일자: ', first_blank, '\n일평균 사망자 수: ', second_blank, \
      '\n일평균 재원 위중증 환자 수: ', third_blank, '\n일평균 확진자 수: ', \
      fourth_blank)
```

기준일자 : (6.19. 00시 기준, '20.1.3. 이후 누계, 단위: 명)

일평균 사망자 수: 14,284

일평균 재원 위중증 환자 수: 111

일평균 확진자 수: 16,271