

크롬브라우저를 이용한 웹 크롤링

# 학습 내용

- ▶ 셀레니움(selenium)에 대해 알아본다.
- ▶ 셀레니움을 이용하여 웹 브라우저 자동화에 대해 알아본다.

# 목차

- ▶ 01. 셀레니움(Selenium)이란?
- ▶ 02. 설치
- ▶ 03. 웹 드라이버(webdriver)
- ▶ 04. 주피터 노트북 실행
- ▶ 05. 웹 페이지 자동 접속하기
- ▶ 06. 정보를 가져오는 방법
- ▶ 07. Xpath에 대해 알아보기

# 01. 셀레니움(Selenium)이란?

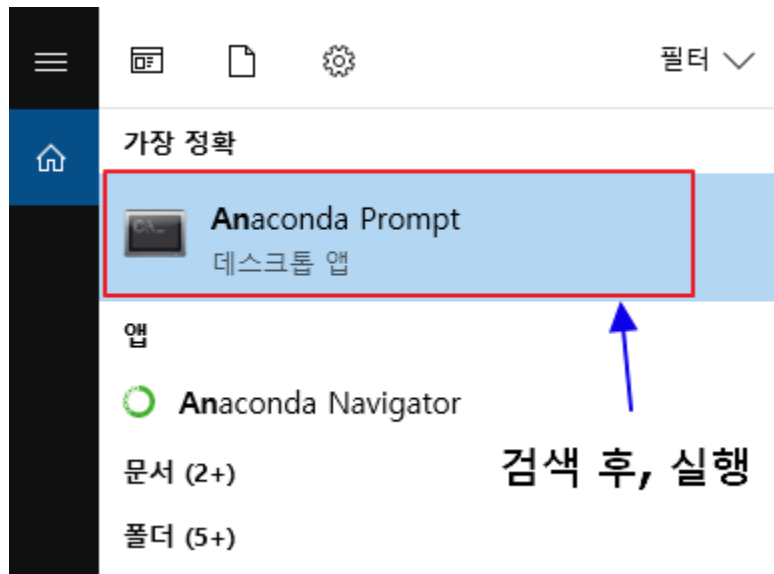
- ▶ 브라우저를 자동화하는 프로그램
- ▶ 주로 개발의 테스트 목적으로 웹 애플리케이션의 자동화를 위한 것이었지만 크롤링등의 자동화 가능해짐.

## 02. 셀레니움(selenium) 설치

### ▶ 설치 진행

(가) 검색 - 'anaconda Prompt' 입력 프로그램 선택 후, 설치 진행.

(나) pip install selenium



검색 후, 실행

A screenshot of the Anaconda Prompt terminal window. The title bar says '선택 Anaconda Prompt'. The command prompt shows the command `>pip install selenium` being entered. The output shows the process of collecting selenium and urllib3, downloading the wheels, and successfully installing them. A red arrow points from the text 'pip install selenium' to the command in the terminal.

```
(section2) C:\Users\WITHJ>pip install selenium
Collecting selenium
  Downloading https://files.pythonhosted.org/packages/80/d6/4294f0b4bce4de0abf13e17190289f9d0613b0a44e5dd6a7f5ca98459853/selenium-3.141.0-py2.py3-none-any.whl (904kB)
    100% |#####| 911kB 1.9MB/s
Collecting urllib3 (from selenium)
  Downloading https://files.pythonhosted.org/packages/62/00/ee1d7de624db8ba7090d1226aebefab96a2c71cd5cfa7629d6ad3f61b79e/urllib3-1.24.1-py2.py3-none-any.whl (118kB)
    100% |#####| 122kB 4.1MB/s
Installing collected packages: urllib3, selenium
Successfully installed selenium-3.141.0 urllib3-1.24.1
You are using pip version 10.0.1, however version 18.1 is available.
You should consider upgrading via the 'python -m pip install --upgrade pip' command.

(section2) C:\Users\WITHJ>
```

pip install selenium

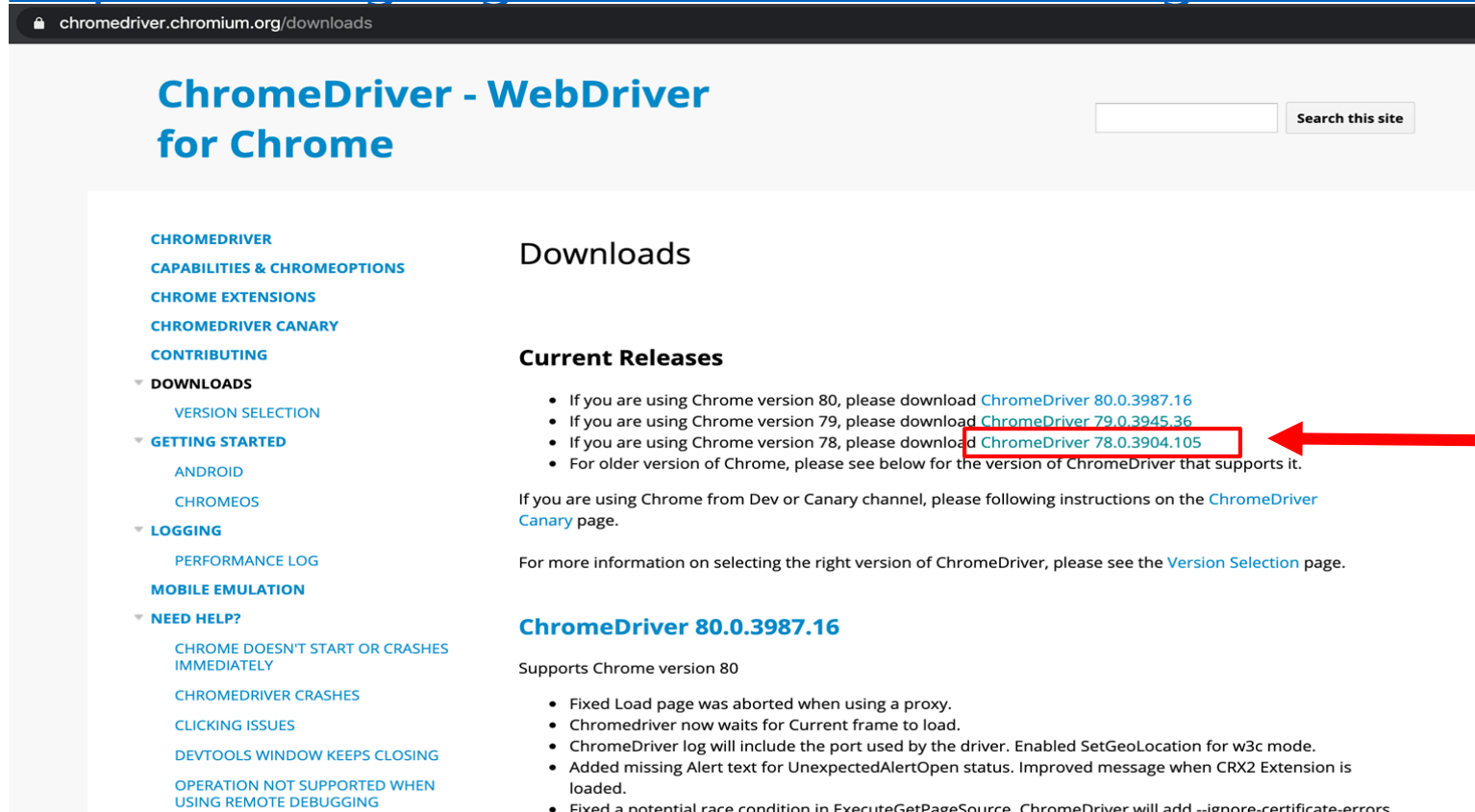
## 03. 웹 드라이버(webdriver)

- ▶ selenium은 webdriver 라는 것을 통해 **컴퓨터(디바이스)에 설치된 브라우저의 제어**가 가능
- ▶ 여러가지 브라우저 중에 일반적으로 **크롬 브라우저**를 많이 이용.  
크롬을 사용하기 위해서는 크롬드라이버(ChromeDriver)를 이용.  
(기타 : Firefox, InternetExplorer 등)
- ▶ 사전에 내 컴퓨터(로컬)에 크롬(Chrome)가 설치되어 있어야 함.  
<https://sites.google.com/a/chromium.org/chromedriver/downloads>

# 03. webdriver

Download URL

<https://sites.google.com/a/chromium.org/chromedriver/downloads>

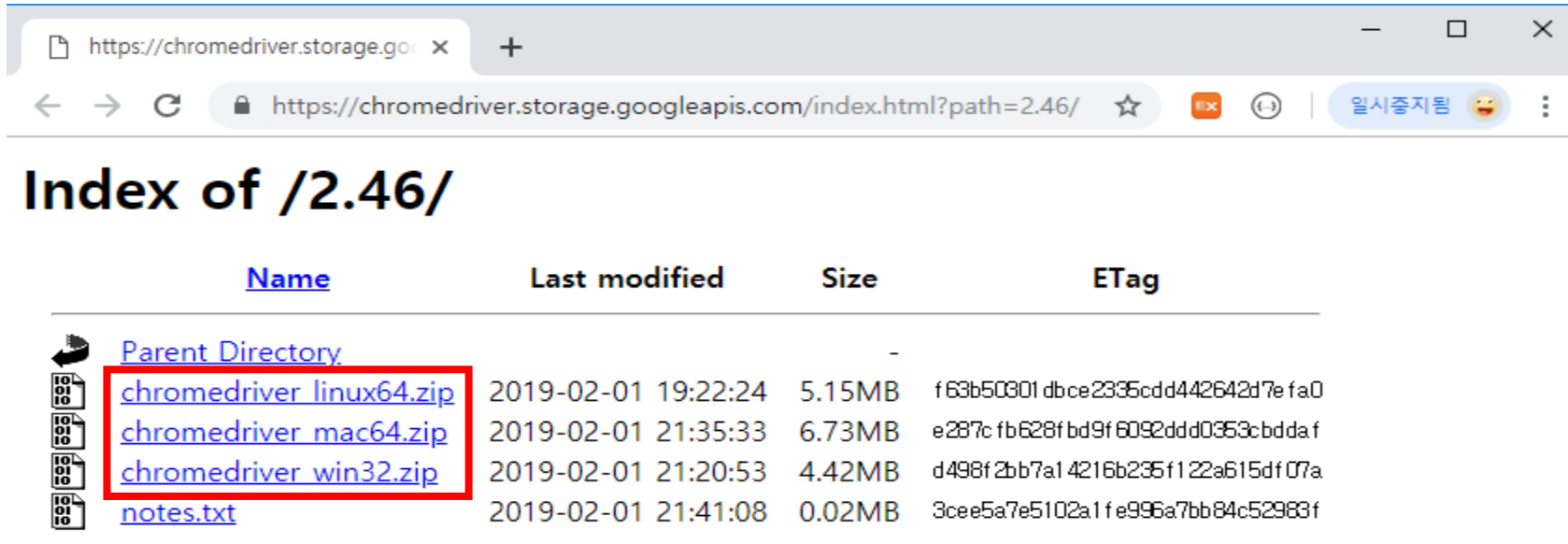


여러가지 프로그램 버전

- \* 최신 버전을 설치 해도 되지만, 여기서는 안정화 된 버전을 선택 후, 다운로드
- \* 크롬 브라우저의 설치된 것과 webdriver의 프로그램이 맞지 않을 경우는, 좀 더 낮은 버전을 찾아 선택 진행

# 03. webdriver

## ▶ 크롬 드라이버(Chrome Driver)

A screenshot of a web browser showing the ChromeDriver storage page. The address bar shows the URL https://chromedriver.storage.googleapis.com/index.html?path=/2.46/. The page title is "Index of /2.46/". Below the title is a table with four columns: Name, Last modified, Size, and ETag. The table lists four items: Parent Directory, chromedriver\_linux64.zip, chromedriver\_mac64.zip, and chromedriver\_win32.zip. The three zip files are highlighted with a red box. The notes.txt file is also listed.

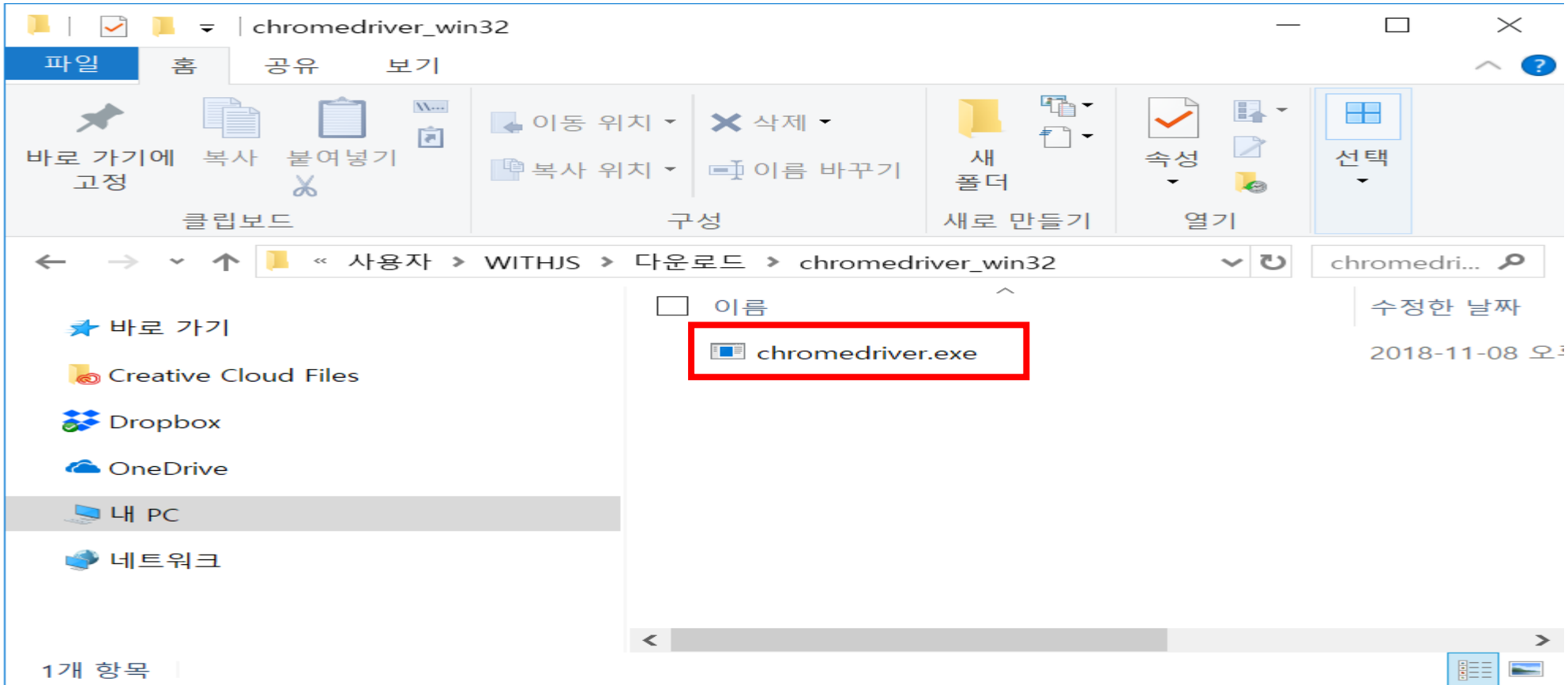
Name	Last modified	Size	ETag
<a href="#">Parent Directory</a>	-	-	-
<a href="#">chromedriver_linux64.zip</a>	2019-02-01 19:22:24	5.15MB	f63b50301dbce2335cdd442642d7efa0
<a href="#">chromedriver_mac64.zip</a>	2019-02-01 21:35:33	6.73MB	e287c fb628fbd9f6092ddd0353cbdda.f
<a href="#">chromedriver_win32.zip</a>	2019-02-01 21:20:53	4.42MB	d498f2bb7a14216b235f122a615df07a
<a href="#">notes.txt</a>	2019-02-01 21:41:08	0.02MB	3cee5a7e5102a1fe996a7bb84c52983f

Window의 경우, chromedriver\_win32.zip를 다운로드 후, 압축을 풀어준다.  
Mac의 경우, chromedriver\_mac64.zip를 다운로드 후, 압축을 풀어준다.  
Linux의 경우, chromedriver\_linux64.zip를 다운로드 후, 압축을 풀어준다.



# 03. webdriver

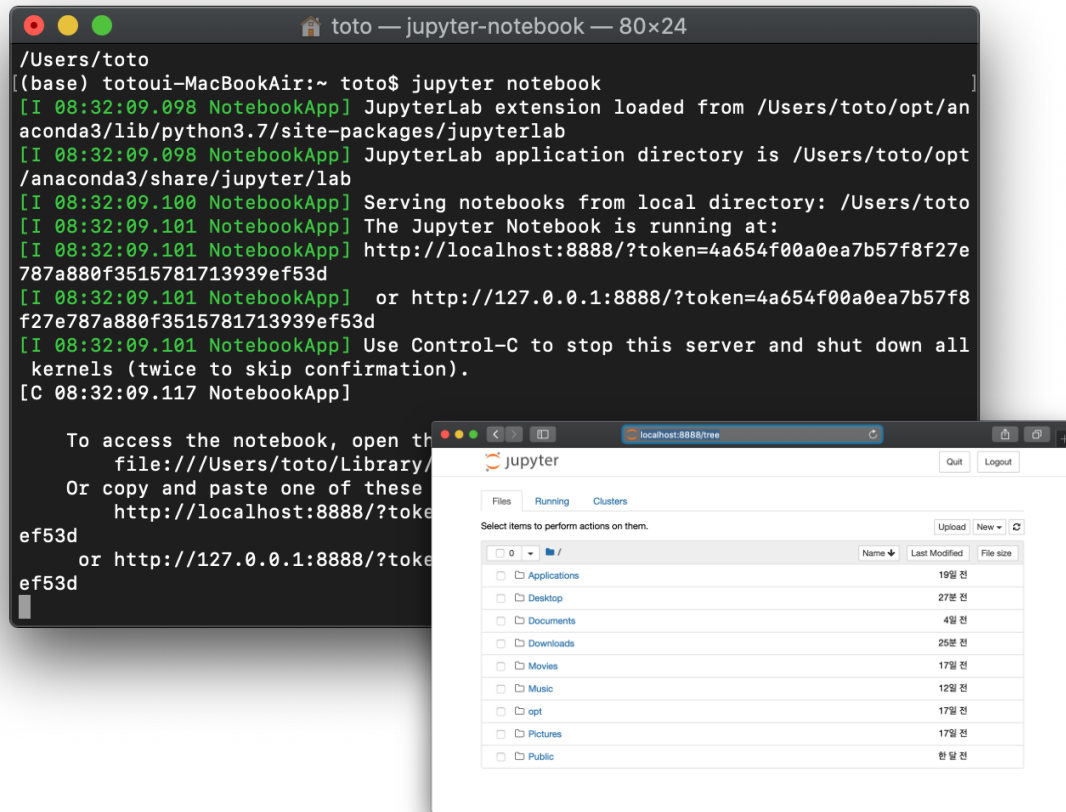
## ▶ 다운로드 압축 해제 후, 작업 위치로 이동



\* 보통 내가 작업하는 파일(ipynb, py)이 위치하고 있는 곳에 해당 파일을 위치시킨다.

## 04. 주피터 노트북 실행

### ▶ Mac(터미널) 에서 Jupyter notebook를 실행

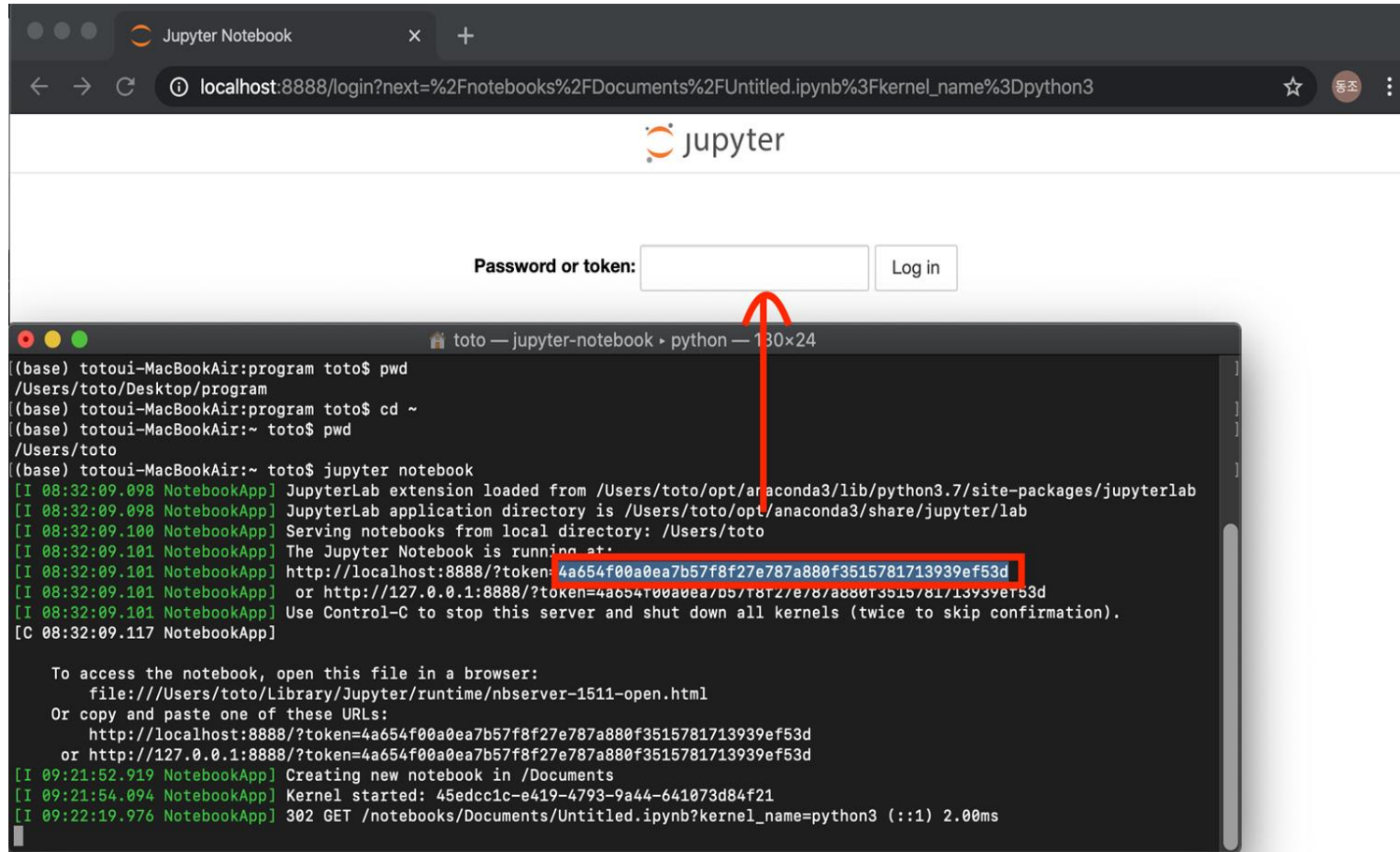
The image shows a terminal window and a JupyterLab web interface. The terminal window, titled 'toto — jupyter-notebook — 80x24', displays the command 'jupyter notebook' and its output, including the URL 'http://localhost:8888/?token=4a654f00a0ea7b57f8f27e787a880f3515781713939ef53d'. The JupyterLab interface, titled 'localhost:8888/tree', shows a file browser with a list of directories and files, including 'Applications', 'Desktop', 'Documents', 'Downloads', 'Movies', 'Music', 'opt', 'Pictures', and 'Public'.

- A. 터미널을 띄우기
- B. jupyter notebook 입력
- C. 주피터 노트북에서 새로운 노트북 생성.

\* 주피터 노트북은 python을 실행할 수 있는 하나의 개발 환경 중의 하나이다.

## 04. 주피터 노트북 실행

▶ 기본 브라우저가 Chrome으로 되어 있지 않을 경우,



The image shows a Jupyter Notebook web interface in a browser window and a terminal window below it. The browser window displays the Jupyter login page with a 'Password or token:' input field and a 'Log in' button. The terminal window shows the command sequence to start Jupyter Notebook. A red box highlights the token '4a654f00a0ea7b57f8f27e787a880f3515781713939ef53d' in the terminal output, and a red arrow points from this token to the 'Password or token:' input field in the browser window.

```
(base) totoui-MacBookAir:program toto$ pwd
/Users/toto/Desktop/program
(base) totoui-MacBookAir:program toto$ cd ~
(base) totoui-MacBookAir:~ toto$ pwd
/Users/toto
(base) totoui-MacBookAir:~ toto$ jupyter notebook
[I 08:32:09.098 NotebookApp] JupyterLab extension loaded from /Users/toto/opt/anaconda3/lib/python3.7/site-packages/jupyterlab
[I 08:32:09.098 NotebookApp] JupyterLab application directory is /Users/toto/opt/anaconda3/share/jupyter/lab
[I 08:32:09.100 NotebookApp] Serving notebooks from local directory: /Users/toto
[I 08:32:09.101 NotebookApp] The Jupyter Notebook is running at:
[I 08:32:09.101 NotebookApp] http://localhost:8888/?token=4a654f00a0ea7b57f8f27e787a880f3515781713939ef53d
[I 08:32:09.101 NotebookApp] or http://127.0.0.1:8888/?token=4a654f00a0ea7b57f8f27e787a880f3515781713939ef53d
[I 08:32:09.101 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 08:32:09.117 NotebookApp]

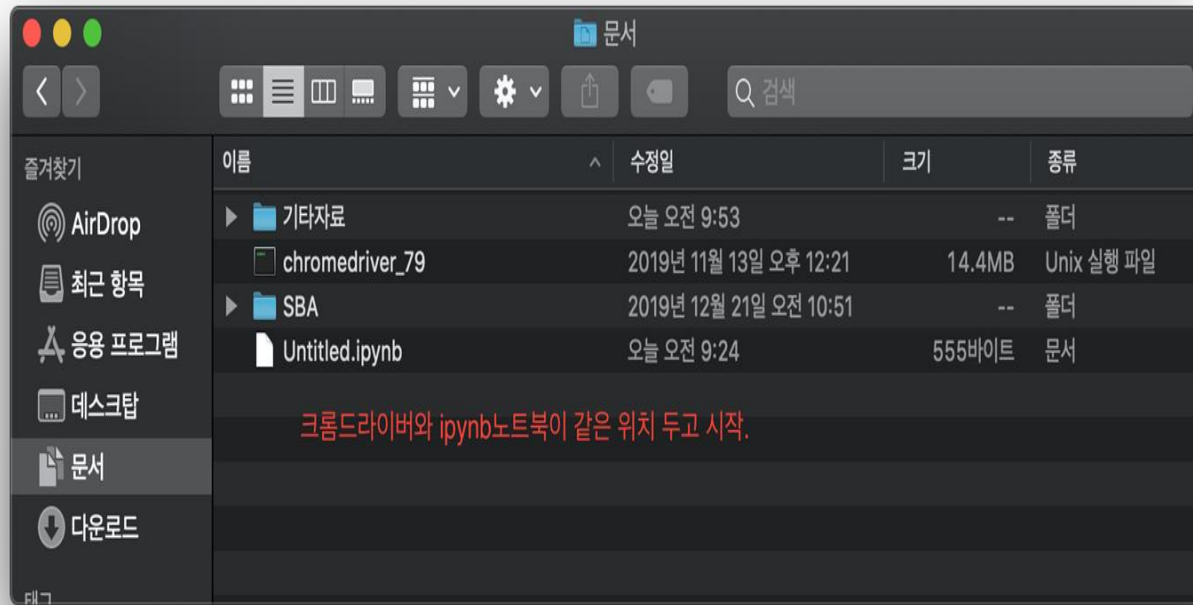
To access the notebook, open this file in a browser:
file:///Users/toto/Library/Jupyter/runtime/nbserver-1511-open.html
Or copy and paste one of these URLs:
http://localhost:8888/?token=4a654f00a0ea7b57f8f27e787a880f3515781713939ef53d
or http://127.0.0.1:8888/?token=4a654f00a0ea7b57f8f27e787a880f3515781713939ef53d
[I 09:21:52.919 NotebookApp] Creating new notebook in /Documents
[I 09:21:54.094 NotebookApp] Kernel started: 45edcc1c-e419-4793-9a44-641073d84f21
[I 09:22:19.976 NotebookApp] 302 GET /notebooks/Documents/Untitled.ipynb?kernel_name=python3 (::1) 2.00ms
```

만약 safari, Internet Explorer 등으로 접속할 경우, 여기서는 크롬을 사용하기 때문에 브라우저를 변경해야 함.

- 크롬 브라우저를 띄운후,
- URL 을 복사 후, 크롬 브라우저에 복사 후, 해당 웹 페이지 접속
- 이때 Password or token 을 물어보면 우리가 처음에 실행시킨 Console창에서 token을 찾아 이를 복사 후, 붙여넣기하면 주피터 노트북의 사용이 가능함.

# 05. 웹 페이지 자동적으로 띄우기

## ▶ 간단하게 파일지정을 위해 같은 위치에 두고 시작



**OS가 Window의 경우,**  
Chromedriver와 ipynb노트북 위치를 같은 위치에 둔다.  
상대경로를 이용하기 위함.  
만약 다른 위치에 있을 경우, 해당되는 전체 경로를 지정해야 함.  
**OS가 Mac의 경우는 지정 경로를 전체 적어주어야 함.**

- **상대 경로**는 내가 현재 위치하고 있는 위치를 기준으로 위의 폴더, 아래 폴더로 이동하는 상대적인 경로를 말함.
- **절대 경로**는 지금 현재 위치하고 있는 위치와 상관없이 **절대적인 주소**로 보면 된다.

내 현재 위치 C:/toto/lim 에 위치하고 있다.

C:/toto/abc 폴더로 이동하려고 한다면 상대경로를 이용하면 -> **../abc** 절대경로를 이용하면 **C:/toto/abc**가 된다.  
상대경로 지정시, **".."**은 현재 폴더의 상위 폴더로 이동을 하는 것을 의미.

## 05. 웹 페이지 자동적으로 띄우기

### ▶ [주피터 노트북 환경] 자동으로 웹 페이지 띄우기

```
# -*- coding: utf-8 -*-  
from selenium import webdriver  
driver = webdriver.Chrome('chromedriver')
```

\* webdriver.Chrome는 ChromeDriver를 사용할 경우를 가르킨다.

\* webdriver.Chrome('경로와파일이름') 실행 파일과 Chromedriver가 같은 위치에 있을 경우, chromedriver만으로 지정 가능.

### ▶ URL 지정 후, 웹 페이지로 이동

```
# -*- coding: utf-8 -*-  
from selenium import webdriver  
driver = webdriver.Chrome('chromedriver')  
url = 'https://ldjwj.github.io/00_SBA01_BigData/05_HTML/idx_lec_list'  
driver.get(url) # url 접속
```

## 05. 웹 페이지 자동적으로 띄우기

### ▶ **ModuleNotFoundError**: No module named 'selenium'

selenium 프로그램이 설치되어 있지 않기에 아래의 명령으로 설치 후, 다시 실행한다.

```
toto — -bash — 108x29
Last login: Sat Dec 28 08:15:18 on ttys000

The default interactive shell is now zsh.
To update your account to use zsh, please run `chsh -s /bin/zsh`.
For more details, please visit https://support.apple.com/kb/HT208050.
(base) totoui-MacBookAir:~ toto$ pip install selenium
Collecting selenium
  Downloading https://files.pythonhosted.org/packages/80/d6/4294f0b4bce4de0abf13e17190289f9d0613b0a44e5dd6a7f5ca98459853/selenium-3.141.0-py2.py3-none-any.whl (904kB)
    |████████████████████████████████████████| 911kB 242kB/s
Requirement already satisfied: urllib3 in ./opt/anaconda3/lib/python3.7/site-packages (from selenium) (1.24.2)
Installing collected packages: selenium
Successfully installed selenium-3.141.0
(base) totoui-MacBookAir:~ toto$
```

## 05. 웹 페이지 자동적으로 띄우기

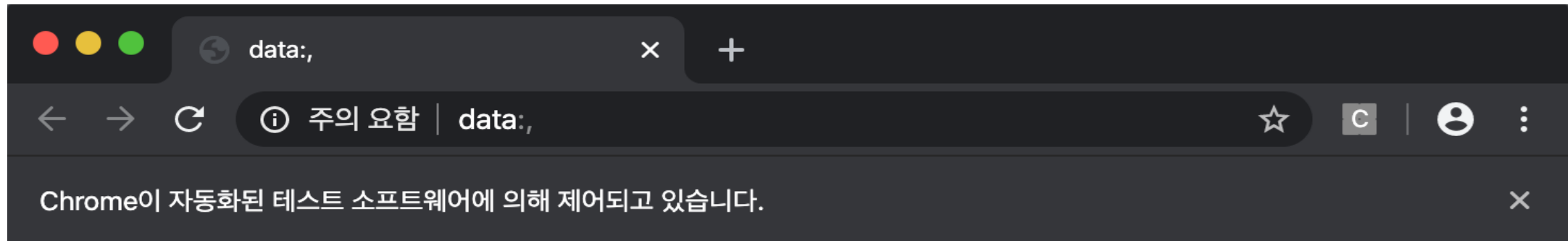
- ▶ MAC의 경우, 경로를 지정해 주어야 함.

```
# -*- coding: utf-8 -*-  
from selenium import webdriver  
driver = webdriver.Chrome('chromedriver')
```

...

Message: 'chromedriver' executable needs to be in PATH. Please see  
<https://sites.google.com/a/chromium.org/chromedriver/home>

```
from selenium import webdriver  
driver = webdriver.Chrome('/Users/toto/Documents/chromedriver_79')
```



## 06. 정보를 가져오는 방법

### ▶ selenium에서 하나 또는 여러개의 정보(객체) 가져오기

`find_element_by_id` : id로 접근 (하나의 정보 가져오기)

`find_elements_by_id` : id로 접근 (여러 개의 정보 가져오기)

#### 지정된 하나의 정보 가져오기

(`find_element_by_[id/name/x..]`)

`find_element_by_id`

`find_element_by_name`

`find_element_by_xpath`

`find_element_by_link_text`

`find_element_by_partial_link_text`

`find_element_by_tag_name`

`find_element_by_class_name`

`find_element_by_css_selector`

#### 여러 개의 정보 가져오기

(`find_elements_by_[id/name/x..]`)

`find_elements_by_name`

`find_elements_by_xpath`

`find_elements_by_link_text`

`find_elements_by_partial_link_text`

`find_elements_by_tag_name`

`find_elements_by_class_name`

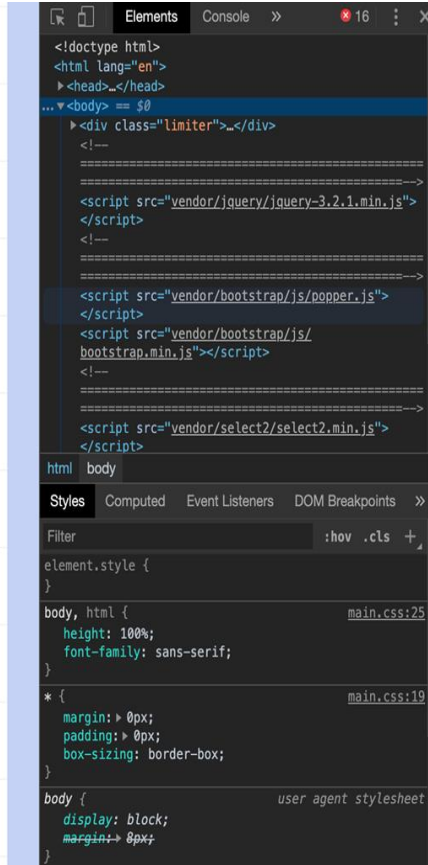
`find_elements_by_css_selector`



# 07. xpath를 이용한 정보 가져오기

## ▶ 웹 페이지로 이동 후, 크롬 개발자 도구 띄우기

통계기본	STAT_BASIC_02	가설검정이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_BASIC_03	회귀분석이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_02	첫번째 모델 만들기	<a href="#">Link</a>
통계기본	STAT_LAB_03	회귀 모델 실습(1)-mtcars	<a href="#">Link</a>
통계기본	STAT_LAB_04	회귀 모델 실습(2)-Boston 집값 예측	<a href="#">Link</a>
통계기본	STAT_LAB_05	회귀 모델 실습(3)-캐글 데이터-집값 예측	<a href="#">Link</a>
통계기본	STAT_LAB_06	로지스틱 회귀 모델 실습(1)-인디언 암 예측	<a href="#">Link</a>
통계기본	STAT_BASIC_04	의사결정트리 기본 이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_07	의사결정트리 실습(1)	<a href="#">Link</a>



- (1) 크롬 브라우저를 띄운다.
  - (2) 크롬 개발자 도구를 띄운다.
- Window의 경우는 F12  
Mac의 경우는 option + command + i

# 07. xpath를 이용한 정보 가져오기

## ▶ xpath를 정보 확인

The screenshot shows a web browser with the URL `ldjwj.github.io/00_SBA01_BigData/05_HTML/idx Lec_list`. The page displays a table with the following data:

통계기본	STAT_BASIC_02	가설검정이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_BASIC_03	회귀분석이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_02	첫번째 모델 만들기	<a href="#">Link</a>
통계기본	STAT_LAB_03	회귀 모델 실습(1)-mtcars	<a href="#">Link</a>

The Chrome DevTools Elements panel is open, showing the DOM tree. The selected element is:

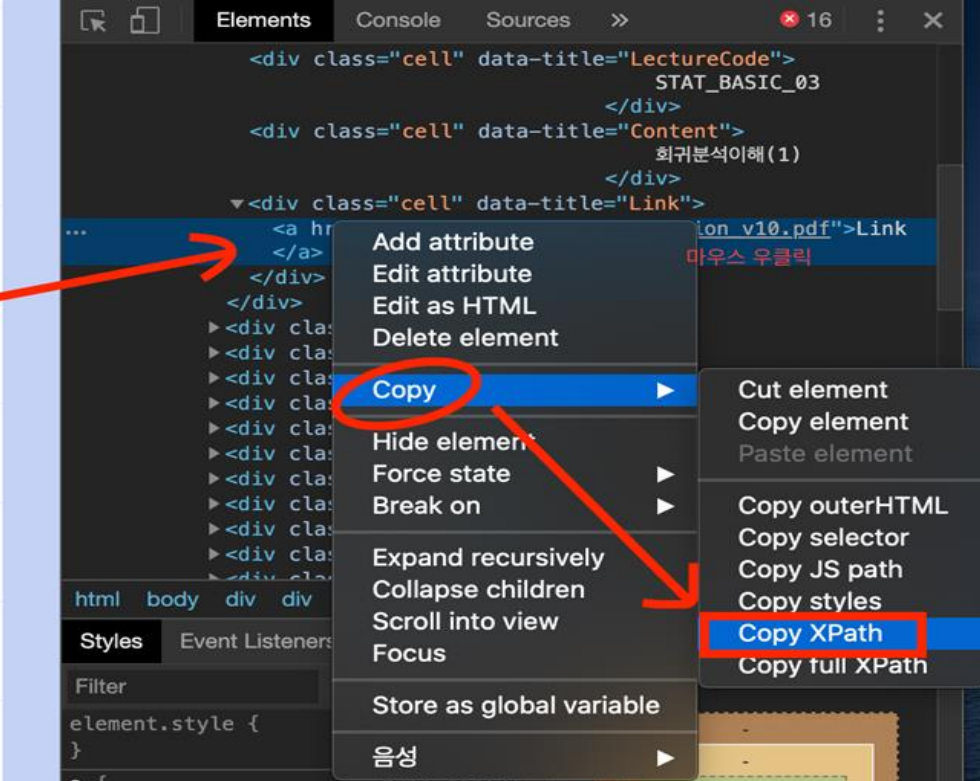
```
<div class="cell" data-title="Link">  
  <a href="01_RClass/Stat02_Regression_v10.pdf">Link</a>  
</div>
```

- (1) 크롬 브라우저를 띄운다.
- (2) 크롬 개발자 도구를 띄운다.  
Window의 경우는 F12  
Mac의 경우는 option + command + i
- (1) 아이콘을 선택한다.
- (2) 내가 원하는 객체를 선택한다.  
여기서는 Link 위치를 선택한다.
- (1) 해당 위치로 이동하는 것을 확인.

# 07. xpath를 이용한 정보 가져오기

## ▶ 웹 페이지로 이동 후, 크롬 개발자 도구 띄우기

통계기본	STAT_BASIC_02	가설검정이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_BASIC_03	회귀분석이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_02	첫번째 모델 만들기	<a href="#">Link</a>
통계기본	STAT_LAB_03	회귀 모델 실습(1)-mtcars	<a href="#">Link</a>
통계기본	STAT_LAB_04	회귀 모델 실습(2)-Boston 집값 예측	<a href="#">Link</a>
통계기본	STAT_LAB_05	회귀 모델 실습(3) - 캐글 데이터-집값 예측	<a href="#">Link</a>

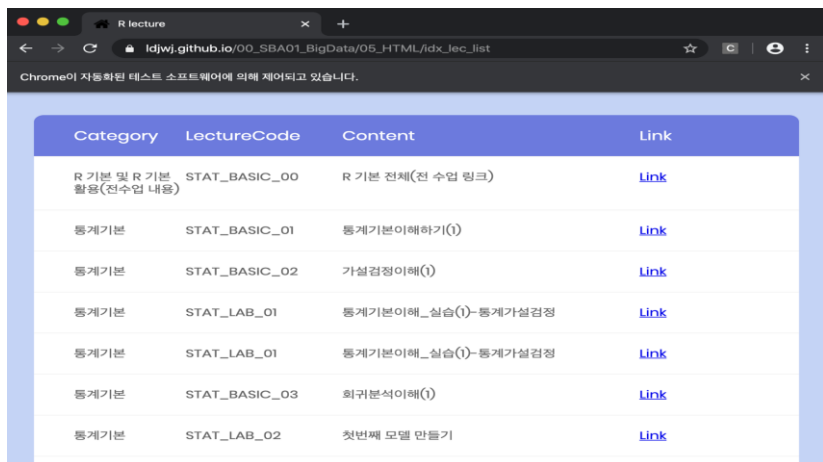


- (1) 해당되는 HTML위치에서 마우스 우클릭을 한 후, Copy를 선택한다.
- (2) Copy XPath를 선택하면 우리는 원하는 위치의 Xpath 정보를 얻을 수 있다.

# 07. xpath를 이용한 정보 가져오기

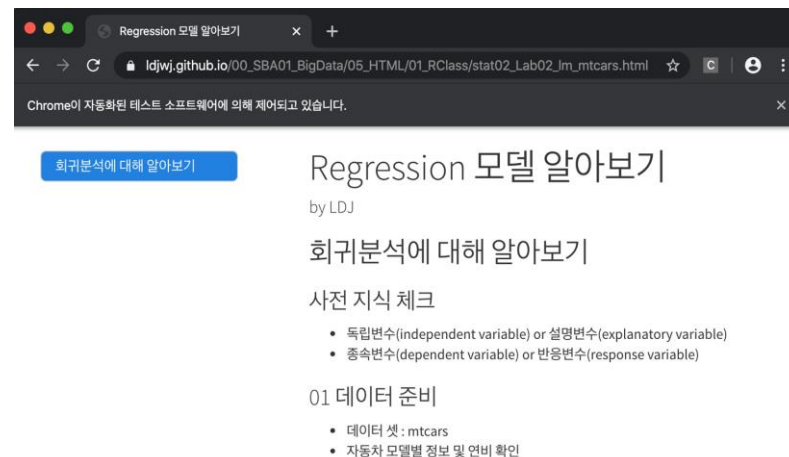
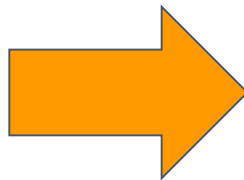
## ▶ URL 지정 후, 웹 페이지로 이동 후, Link를 선택

```
from selenium import webdriver
driver = webdriver.Chrome('chromedriver')
url = 'https://ldjwj.github.io/00_SBA01_BigData/05_HTML/idx Lec list'
driver.get(url) # url 접속
# xpath : /html/body/div/div/div/div/div[9]/div[4]/a
link_obj = driver.find_element_by_xpath('/html/body/div/div/div/div/div[9]/div[4]/a')
link_obj.click()
```



A screenshot of a web browser displaying a table with lecture information. The table has four columns: Category, LectureCode, Content, and Link. The data rows include lecture codes like STAT\_BASIC\_00, STAT\_BASIC\_01, STAT\_BASIC\_02, STAT\_LAB\_01, STAT\_LAB\_02, and STAT\_BASIC\_03, along with their respective content descriptions and links.

Category	LectureCode	Content	Link
R 기본 및 R 기본 활용(전수업 내용)	STAT_BASIC_00	R 기본 전체(전 수업 링크)	<a href="#">Link</a>
통계기본	STAT_BASIC_01	통계기본이해하기(1)	<a href="#">Link</a>
통계기본	STAT_BASIC_02	가설검정이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_LAB_01	통계기본이해_실습(1)-통계가설검정	<a href="#">Link</a>
통계기본	STAT_BASIC_03	회귀분석이해(1)	<a href="#">Link</a>
통계기본	STAT_LAB_02	첫번째 모델 만들기	<a href="#">Link</a>



A screenshot of a web browser displaying a page titled "Regression 모델 알아보기" (Understanding Regression Models) by LDJ. The page includes a button "회귀분석에 대해 알아보기" (Learn about regression analysis) and a section "회귀분석에 대해 알아보기" (Learn about regression analysis) with a sub-section "사전 지식 체크" (Check prior knowledge). The sub-section lists two types of variables: 독립변수 (independent variable) or 설명변수 (explanatory variable), and 종속변수 (dependent variable) or 반응변수 (response variable). Below this, there is a section "01 데이터 준비" (01 Data Preparation) which lists "데이터 셋 : mtcars" (Data set : mtcars) and "자동차 모델별 정보 및 연비 확인" (Check information and mileage by car model).

## 08. Xpath에 대해 알아보기

- ▶ XPath(XML Path Language)는 W3C의 표준이다.
- ▶ XML(Extensible Markup Language)문서의 구조를 통해 경로(Path)위에 지정한 구문을 사용하여 항목을 배치하고 처리하는 방법을 기술한 언어이다.
- ▶ Xpath는 XML문서를 트리 구조로 모델링하여 처리한다.
- ▶ XML 문서의 계층구조, 즉 트리 구조에서 Node들을 식별, 선택, 조작하기 위해 Xpath 표현식을 사용

참조 url : <http://www.w3.org/TR/xpath20/>