훈련교사 : 전 은 석
email : euns_jun@naver.com

데이터 수집

훈 련 교 사 : 전 은 석

훈련교사 : 전 은 석
email : euns_jun@naver.com

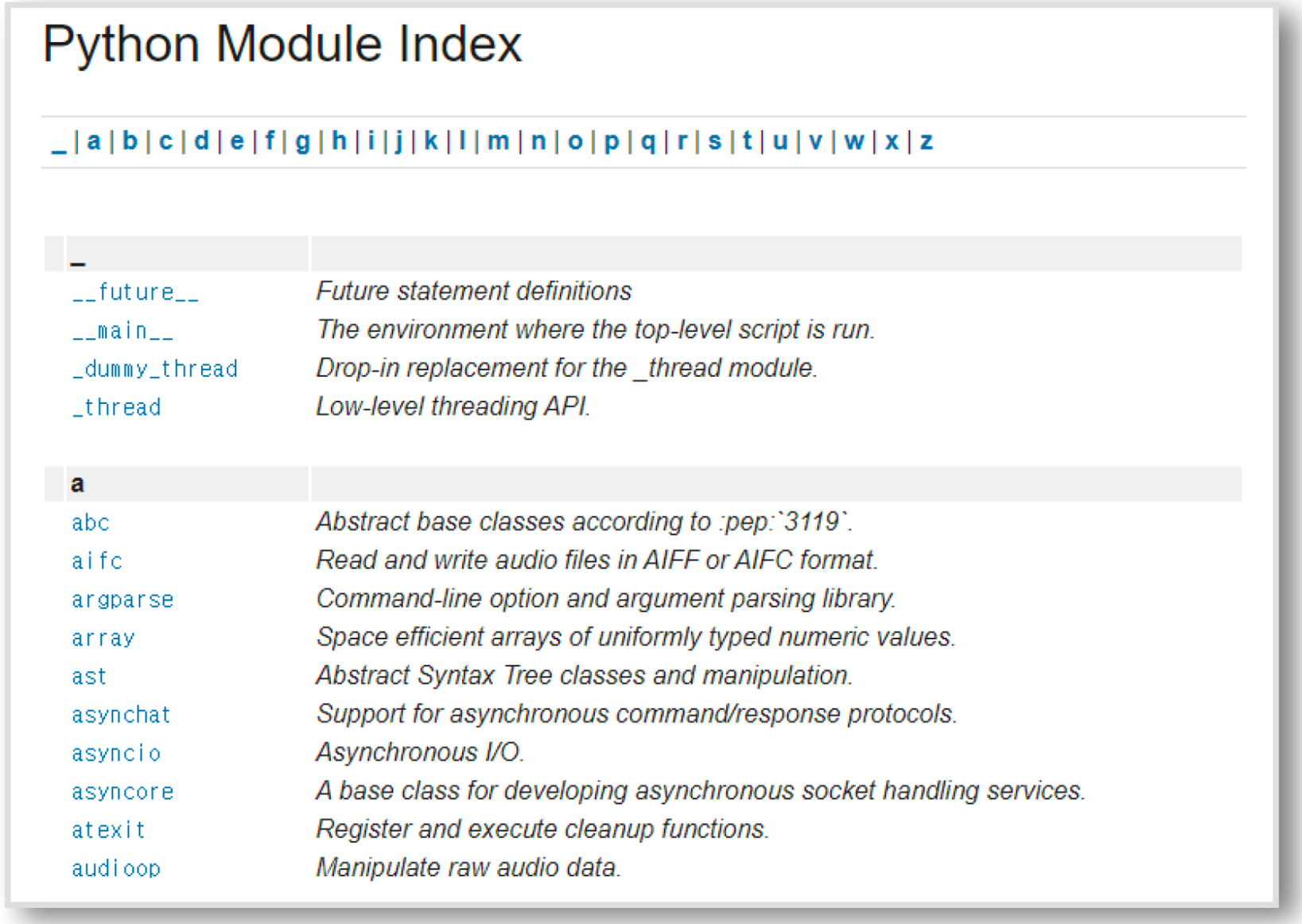
Python Modules

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ Python 모듈

- Python이 제공하는 표준 라이브러리 모듈 확인방법

  https://docs.python.org/3/py-modindex.html



Python Module Index

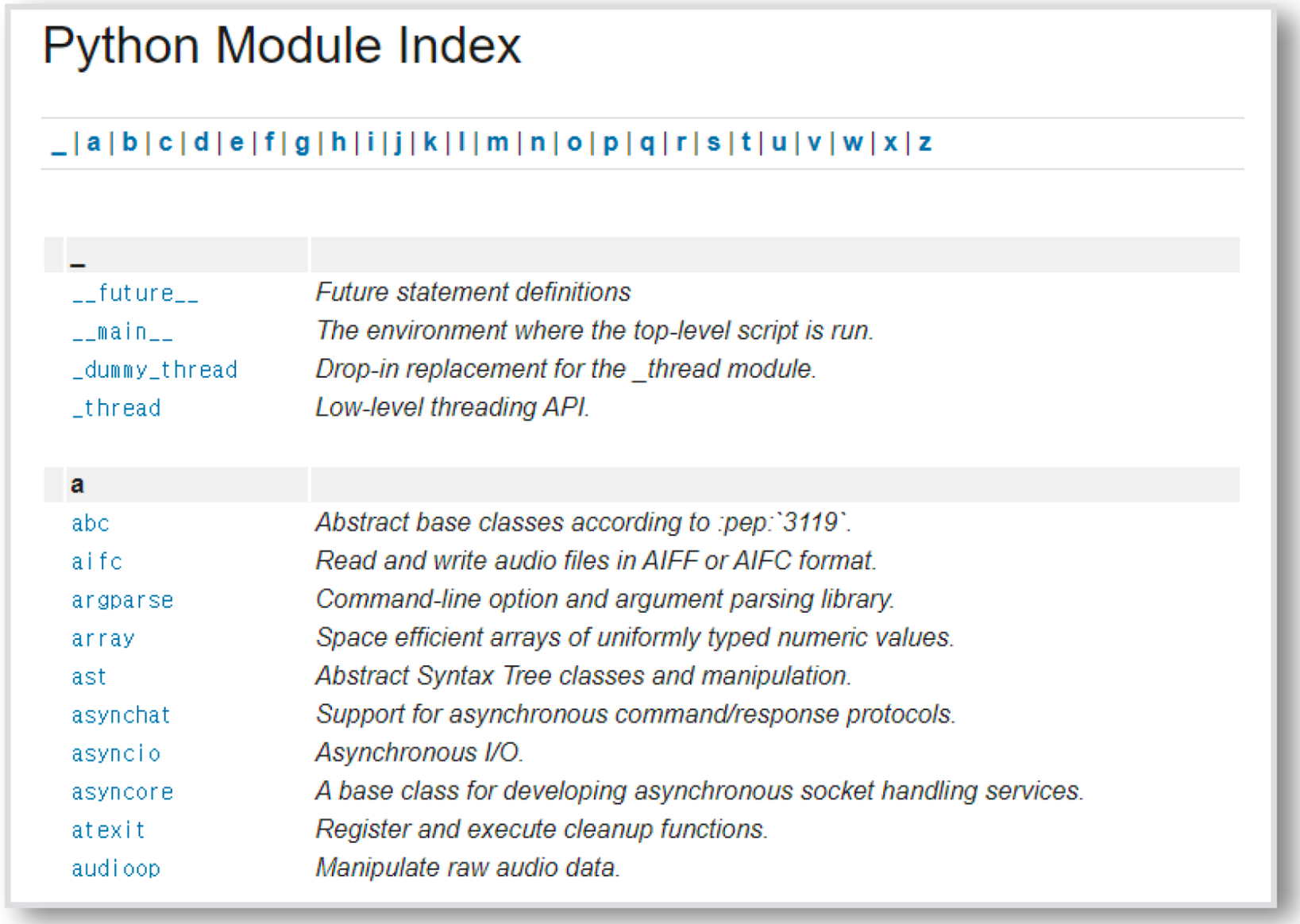
_ | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | z

_	
__future__	*Future statement definitions*
__main__	*The environment where the top-level script is run.*
_dummy_thread	*Drop-in replacement for the _thread module.*
_thread	*Low-level threading API.*
a	
abc	*Abstract base classes according to :pep:`3119`.*
aifc	*Read and write audio files in AIFF or AIFC format.*
argparse	*Command-line option and argument parsing library.*
array	*Space efficient arrays of uniformly typed numeric values.*
ast	*Abstract Syntax Tree classes and manipulation.*
asynchat	*Support for asynchronous command/response protocols.*
asyncio	*Asynchronous I/O.*
asyncore	*A base class for developing asynchronous socket handling services.*
atexit	*Register and execute cleanup functions.*
audioop	*Manipulate raw audio data.*

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ Python 모듈

- Python이 제공하는 표준 라이브러리 모듈 확인방법
  https://docs.python.org/3/py-modindex.html



Python Module Index

_ | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o | p | q | r | s | t | u | v | w | x | z

_	
__future__	*Future statement definitions*
__main__	*The environment where the top-level script is run.*
_dummy_thread	*Drop-in replacement for the _thread module.*
_thread	*Low-level threading API.*
a	
abc	*Abstract base classes according to :pep:`3119`.*
aifc	*Read and write audio files in AIFF or AIFC format.*
argparse	*Command-line option and argument parsing library.*
array	*Space efficient arrays of uniformly typed numeric values.*
ast	*Abstract Syntax Tree classes and manipulation.*
asynchat	*Support for asynchronous command/response protocols.*
asyncio	*Asynchronous I/O.*
asyncore	*A base class for developing asynchronous socket handling services.*
atexit	*Register and execute cleanup functions.*
audioop	*Manipulate raw audio data.*

- help('modules') 명령어로 현재 설치되어 있는 모듈 확인 가능

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ Python 모듈

- os 모듈 : 운영체제와 상호작용하기 위한 여러 함수들을 제공
  os.getcwd() : 현재 작업 경로 반환
  os.chdir('경로') : 작업 경로를 입력된 경로로 변경

```
>>> import os
>>> os.getcwd()
'/Volumes/data/dataProj/crawling/src'
>>> os.chdir('../../python/src')
>>> os.getcwd()
'/Volumes/data/dataProj/python/src'
>>>
```

- time 모듈 : 시간과 관련된 함수들을 제공

```
>>> import time
>>> time.localtime()
time.struct_time(tm_year=2023, tm_mon=4, tm_mday=30, tm_hour=15,
tm_min=5, tm_sec=13, tm_wday=6, tm_yday=120, tm_isdst=0)
>>> time.sleep(1)
>>> time.localtime()
time.struct_time(tm_year=2023, tm_mon=4, tm_mday=30, tm_hour=15,
tm_min=5, tm_sec=19, tm_wday=6, tm_yday=120, tm_isdst=0)
>>>
```

훈련교사 : 전 은 석
email : euns_jun@naver.com

requests

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- Kenneth Reitz에 의해 개발된 파이썬 라이브러리
- HTTP 프로토콜과 관련된 기능 지원
- 특징]
  - 딕셔너리(dict) 형태로 데이터 전송
  - 요청 메서드(GET, POST)를 명시하여 요청
- 공식 홈페이지 : https://requests.readthedocs.io/en/latest/

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- 콘솔 환경(CMD, terminal)에서 pip show requests 명령으로 설치확인

```
euns_macmini@euns-Macmini src % pip show requests
Name: requests
Version: 2.28.2
Summary: Python HTTP for Humans.
Home-page: https://requests.readthedocs.io
Author: Kenneth Reitz
Author-email: me@kennethreitz.org
License: Apache 2.0
Location: /Library/Frameworks/Python.framework/Versions/3.10/lib/pyt
hon3.10/site-packages
Requires: certifi, charset-normalizer, idna, urllib3
Required-by: requests-oauthlib, tensorboard
euns_macmini@euns-Macmini src %
```

- 설치 : pip install requests

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- HTTP 요청 구조

  사용자가 서버에 요청을 할 때는 네 가지로 구분하여 요청 가능

  - URL로 요청하면 메서드를 변경하여 기능 구분

메서드	설 명
GET	정보를 가져오기 위해 요청
POST	새로운 정보를 보내기 위해 요청
PUT	수정할 정보를 보내기 위해 요청
DELETE	정보를 삭제하기 위해 요청
HEAD	콘텐츠 없이 요청 헤더만을 받아오는 방식

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- HTTP 응답 구조

  서버가 사용자에게 요청에 대한 응답을 보낼 때 크게 5가지의 경우 존재

  - 응답 코드로 요청의 진행 상황과 서버의 상태를 예측 가능

응답 코드	설 명
1XX	요청을 받았고, 작업 진행 중
2XX	사용자의 요청이 성공적으로 수행 됨
3XX	요청은 완료 되었으나, 리다이렉션이 필요
4XX	사용자의 요청이 잘못됨
5XX	서버에 오류가 발생함

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- GET 방식 요청

```
requests.request('GET', url, **kwargs)
```

```
requests.get(url, **kwargs)
```

- kwargs :

  params : 요청 시 전달할 문자열(파라미터)을 지정 - 딕셔너리(dict)
  선택적으로 사용

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- POST 방식 요청

```
requests.request('POST', url, **kwargs)
```

```
requests.post(url, **kwargs)
```

- kwargs :
  - data : 요청 시 전달할 문자열(파라미터)을 지정
    - 선택적으로 사용
    - 딕셔너리, 튜플, 리스트, 바이트열(bytes) 형식
  - headers : 요청 헤더 직접 설정 --> 인증 토큰의 경우 유용
  - json : 선택적으로 요청 시 전달할 데이터를 JSON 타입의 객체를 지정
    - JSON 형식

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ requests 패키지

- 응답 객체 속성

  - status_code : 응답코드
  - headers : 응답헤더(dict 형태)
  - context : binary 원문
  - text : UTF-8로 인코딩된 응답문서의 내용
  - json() : 응답 데이터가 JSON 포멧의 경우 dict 객체 반환

훈련교사 : 전 은 석
email : euns_jun@naver.com

BeautifulSoup

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- 홈페이지 내 데이터를 쉽게 추출 할 수 있도록 도와주는 파이썬 외부 라이브러리
- 웹 문서 내 수많은 HTML 태그들을 파서(parser)를 활용해 사용하기 편한 파이썬 객체로 만들어 제공
  - html, xml 파서 제공
- 웹 문서 구조를 알고 있다면, 편하게 원하는 데이터를 추출 활용 가능

- HTML 문서를 태그를 기반으로 구조화해서 태그로 원하는 데이터를 추출

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- 콘솔 환경(CMD, terminal)에서

  pip install beautifulsoup4 명령으로 설치

```
pip install beautifulsoup4
```

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

· BeautifulSoup 모듈 사용방법

```
from bs4 import BeautifulSoup
```

- 모듈 내 BeautifulSoup() 에 HTML 문서와 파서(parser)를 전달하여 분석 결과를 객체에 저장
- 파서에 따라 HTML을 분석할 때, 태그를 추가, 무시, 강제 변경 등의 작업 수행

```
BeautifulSoup("<a></p>", "html.parser")
```

```
<a></a>
```

```
BeautifulSoup("<a></p>", "lxml")
```

```
<html><body><a></a></body></html>
```

```
BeautifulSoup("<a></p>", "html5lib")
```

```
<html><head></head><body><a><p></p></a></body></html>
```

```
BeautifulSoup("<a><b /></a>", "xml")
```

```
<?xml version="1.0" encoding="utf-8"?>
<a><b/></a>
```

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- BeautifulSoup 모듈 사용방법

  - BeautifulSoup은 HTML을 파싱하여 구조화하는 모듈로 urllib, requests 모듈 등과 함께 사용

  - requests 모듈로 웹 문서를 텍스트로 가져온 뒤 BeautifulSoup 모듈로 분석

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- BeautifulSoup 모듈 사용방법

  - 태그 : HTML의 해당 태그에 대한 첫번째 정보를 가져옴
    태그['속성'] : HTML해당 태그의 속성에 대한 첫 번째 정보를 가져옴

  - find() : HTML의 해당 태그에 대한 첫 번째 정보를 가져옴
    find(속성='값') : HTML 해당 속성과 일치하는 값에 대한 첫 번째 정보를 가져옴
  - find_all() : HTML의 해당 태그에 대한 모든 정보를 리스트 형식으로 가져옴
    limits 옵션으로 개수 지정 가능

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- BeautifulSoup 모듈 사용방법

- 태그 : HTML의 해당 태그에 대한 첫번째 정보를 가져옴
  태그['속성'] : HTML해당 태그의 속성에 대한 첫 번째 정보를 가져옴

- find() : HTML의 해당 태그에 대한 첫 번째 정보를 가져옴
  find(속성='값') : HTML 해당 속성과 일치하는 값에 대한 첫 번째 정보를 가져옴

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- BeautifulSoup 모듈 사용방법

- find_all() : HTML의 해당 태그에 대한 모든 정보를 리스트 형식으로 가져옴

limits 옵션으로 개수 지정 가능

CSS 속성으로 필터링

(class = '클래스이름' 또는 attrs 속성에 {'속성': '값'})

string 으로 검색

: 해당 값이 있는지 검사 할 때 활용, 정규 표현식과 함께 활용

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ BeautifulSoup 모듈

- BeautifulSoup 모듈 사용방법

  - select_one(), select()
    : CSS 선택자를 활용하여 원하는 정보를 가져옴
    (find, find_all과 비슷)

  - get_text() : 검색 결과에서 태그를 제외한 텍스트만 출력
    get('속성') : 해당 속성의 값을 출력

  - string : 검색 결과에서 태그 안에 또 다른 태그가 없는 경우
    해당 내용 출력

훈련교사 : 전 은 석
email : euns_jun@naver.com

selenium

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈

* 웹 어플리케이션 테스트를 위한 프레임워크
  - 주로 제작한 홈페이지를 테스트하기 위해 사용
* 다양한 언어에서 지원(C++, Java, Python 등)
  - 사용자가 아닌 프로그램이 웹 브라우저를 제어할 수 있도록 지원
* 웹 브라우저마다 웹 브라우저와 프로그램 간 통신 목적으로 클라이언트 프로그램(Web Driver)이 별도로 필요
* 크롤링보다는 웹을 제어하는 목적이 더 크다.

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈 설치

* 설치 : pip install selenium

❖ Web Driver 설치

* 컴퓨터 운영체제와 웹 브라우저 종류에 따라 개별 설치
  - 브라우저의 버젼, PC 환경 등의 변수가 많아 Chrome에서 실습 권장
* 자신에게 맞는 종류를 다운로드 받아 적당한 위치에 압축해제
  - 압축 해제한 파일명은 수정하지 말 것

웹 브라우저	다운로드 경로
google chrome	https://chromedriver.chromium.org/downloads
microsoft edge	https://docs.microsoft.com/ko-kr/microsoft-edge/webdriver
firefox	https://github.com/mozilla/geckodriver/releases
InternetExplorer	https://www.microsoft.com/en-us/download/detail.asps?id=44069

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈 사용 방법

* selenium 모듈 호출 후 설치한 Web driver 경로 지정
* 원하는 홈페이지를 해당 웹 드라이버로 실행

```
import selenium
from selenium import webdriver

path = '압축해제한 폴더내의 웹 드라이버 파일 경로'
driver = webdriver.Chrome(path)
driver.get('방문할 사이트 주소')
```

* 페이지 소스코드 추출 : driver.page_source
* 태그 내용 읽기 : 태그선택.text

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈 사용 방법

* 태그 접근 함수

단일 객체	find_element(By.속성, '속성 값')
복수 객체 (리스트 형태)	find_elements(By.속성, '속성 값')

By.속성	설명
By.ID	태그의 id값으로 추출
By.NAME	태그의 name값으로 추출
By.XPATH	태그의 경로로 추출
By.LINK_TEXT	링크 텍스트값으로 추출
By.PARTIAL_LINK_TEXT	링크 텍스트의 자식 텍스트 값을 추출
By.TAG_NAME	태그 이름으로 추출
By.CLASS_NAME	태그의 클래스명으로 추출
By.CSS_SELECTOR	css선택자로 추출

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈 사용 방법

* xpath 문법

표현식	설 명
/	절대경로
//	문서 내에서 검색
//@href	href 속성이 있는 모든 태그 선택
//a[@href='http://google.com']	a 태그의 href 속성값이 'http://google.com' 인 태그 모두 선택
(//a)[3]	문서의 세번째 a 태그 선택
(//table)[last()]	문서의 테이블 태그중 마지막 테이블 태그 선택
(//a)[position() < 3]	문서의 처음 두 링크 선택(위치가 3 미만인 a 태그 선택)
//table/tr/*	모든 테이블 태그의 모든 자식 tr 태그 선택
//div[@*]	속성이 하나라도 있는 div 태그 선택

* **selenium 사용시 selenium으로 실행한 브라우저가 종료되면 세션이 끊어져서 코드 실행이 안된다.**

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈 사용 방법

* 이벤트로 제어하기

이벤트	메 소 드
마우스 클릭	click()
키보드 입력	send_keys()
자바스크립트 삽입	execute_script()
입력 폼 전송	submit()
스크린샷	screenshot(파일이름)
글자 지움	clear()
뒤로 가기	back()
앞으로 가기	forward()

* 태그를 선택한 후 이벤트 메소드를 호출해서 제어
* 드라이버를 직접 실행시켜서 제어하기 때문에 로딩이 오래걸린다.
* 많은 작업을 한번에 처리하는 경우 오류가 발생할 수 있다.
  ==> time 모듈의 sleep() 함수를 활용해서 진행시간을
  지연시키는 방법이 있다.

훈련교사 : 전 은 석
email : euns_jun@naver.com

❖ selenium 모듈 사용 방법

* 특수 키 입력

- 특수 키를 입력하기 위해서는 추가 모듈이 필요하다.

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

* Keys.상수

```
ARROW_DOWN, ARROW_LEFT, ARROW_RIGHT, ARROW_UP,
BACKSPACE, DELETE, HOME, END, INSERT, ALT, COMMAND,
CONTROL, SHIFT, ENTER, ESCAPE, SPACE, TAB, F1, F2, ..., F12
```