

# 빅데이터시스템

TECHNOLOGY ANALYTICS

파이썬크롤링 라이브러리이용



# 목 차

## 01 라이브러리 크롤링

Selenium

## 02 KoGPT를 활용한 챗봇 환경 구축

KoGPT, 디렉토리구성, 패키지설치, 프로그램코딩, 실행, 접속

# \*. 중간평가 리뷰

1

```
1 x 2 = 2    1 x 4 = 4    1 x 6 = 6    1 x 8 = 8
2 x 2 = 4    2 x 4 = 8    2 x 6 = 12   2 x 8 = 16
3 x 2 = 6    3 x 4 = 12   3 x 6 = 18   3 x 8 = 24
4 x 2 = 8    4 x 4 = 16   4 x 6 = 24   4 x 8 = 32
5 x 2 = 10   5 x 4 = 20   5 x 6 = 30   5 x 8 = 40
6 x 2 = 12   6 x 4 = 24   6 x 6 = 36   6 x 8 = 48
7 x 2 = 14   7 x 4 = 28   7 x 6 = 42   7 x 8 = 56
8 x 2 = 16   8 x 4 = 32   8 x 6 = 48   8 x 8 = 64
9 x 2 = 18   9 x 4 = 36   9 x 6 = 54   9 x 8 = 72
```

```
for y in range(1, 10):
    for x in range(2, 9, 2):
        print(f" { y } x { x } = { x * y :2d}", end=" ")
    print()
print()
```

# \*. 중간평가 리뷰

2

## ⌚ 빅데이터 플랫폼

### ■ 데이터 플랫폼의 발전

- 데이터 플랫폼은 정형화된 형태로 데이터를 저장하는 파일 시스템으로 시작
- 다수가 동시에 사용할 수 있는 데이터베이스와 데이터 웨어하우스(DW)로 발전
- 폭발적으로 증가하는 데이터를 저장 및 유통하기 위한 빅데이터 플랫폼으로 진화

### ■ 빅데이터 플랫폼의 개념

- 대량의 데이터를 저장 및 분석, 처리할 수 있는 대용량의 고속 저장 공간 보유
- 고성능 계산 능력과 실시간으로 발생하는 빅데이터를 처리 및 분석하여 일관성을 유지
- 빅 데이터에서 발생하는 개인 정보를 위한 정보 보안 관리체계 지원도 필요
- 빅데이터 플랫폼은 오픈 소스인 하둡을 근간으로 많이 사용

# \*. 중간평가 리뷰

3

```
>>> a = 0o17
>>> b = 0xA
>>> c = 2.0
>>> if ( a > b ): c = a * c
..... else: c = b / c
.....

>>> print( c )
```

$a = 1 \times 8 + 7 \times 1 = 15$

$b = A(10) \times 1 = 10$

$c = 2.0$

$\text{if } ( 15 > 10 ): c = 15 \times 2.0$

30.0

# \*. 중간평가 리뷰

4

행렬곱

$$\begin{bmatrix} 1 & 2 & 3 \\ 4 & 5 & 6 \end{bmatrix} \times \begin{bmatrix} 10 & 20 \\ 30 & 40 \\ 50 & 60 \end{bmatrix} = \begin{bmatrix} 220 & 280 \\ 490 & 640 \end{bmatrix}$$

```
import numpy as np
```

```
ar1 = np.array([[1,2,3], [4,5,6]])
```

```
ar2 = np.array([10,20,30,40,50,60]).reshape((3,2))
```

```
ar9 = np.dot(ar1, ar2)
```

```
print(ar9)
```

$$1 \times 10 + 2 \times 30 + 3 \times 50 = 220$$

$$1 \times 20 + 2 \times 40 + 3 \times 60 = 280$$

$$4 \times 10 + 5 \times 30 + 6 \times 50 = 490$$

$$4 \times 20 + 5 \times 40 + 6 \times 60 = 640$$

# \*. 중간평가 리뷰

5

p = 'python is good'

x = p.count('o') x = 3

y = p.find('o') y = 4

z = p.index('o') z = 4

print(x + y - z)

3 + 4 - 4 = 3

## \*. 전수업리뷰

 크롤링 : 웹페이지를 그대로 가져와서 그곳으로 부터 데이터를 추출

### ■ 의의

- 일반적으로 검색엔진에서 사용하는 기술 요소
- 사람이 일일이 해당 사이트의 정보를 검색하는 것이 아니라 컴퓨터 프로그램의 미리 입력된 방식에 따라 끊임없이 새로운 웹페이지를 찾아 종합하고 색인하는 작업을 반복 수행
- 크롤링하는 행위를 하는 소프트웨어를 크롤러(crawler) 라고 부름

### ■ 종류

- 스크래핑(scraping) : 하나의 페이지를 수집함
- 크롤링(crawling) : 동적으로 웹페이지를 돌아다니면서 데이터 수집



# \*. 전수업리뷰

※ API(Application Programming Inteface) : 컴퓨터나 컴퓨터 사이의 연결

## ⌚ 웹 API

### ■ 의의

- 웹 API는 일반적으로 HTTP 통신을 사용하는데 사용
- 지도, 검색, 주가, 환율 등 다양한 정보를 가지고 있는 웹 사이트의 기능을 외부에서 쉽게 사용할 수 있도록 사용 절차와 규약을 정의한 것

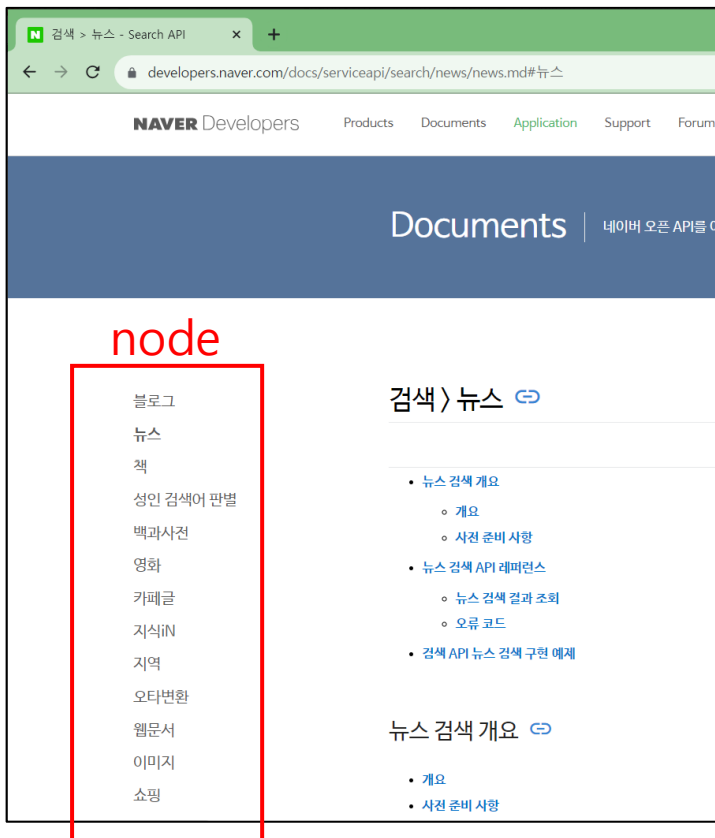
종류	주소
네이버 개발자 센터	<a href="https://developers.naver.com">https://developers.naver.com</a>
카카오 앱 개발 플랫폼 서비스	<a href="https://developers.kakao.com">https://developers.kakao.com</a>
페이스북 개발자 센터	<a href="https://developers.facebook.com">https://developers.facebook.com</a>
트위터 개발자 센터	<a href="https://developer.twitter.com">https://developer.twitter.com</a>

종류	주소
공공데이터포털	<a href="https://www.data.go.kr">https://www.data.go.kr</a>
세계 날씨	<a href="http://openweathermap.org">http://openweathermap.org</a>
유료/무료 API 스토어	<a href="http://mashup.or.kr">http://mashup.or.kr</a> <a href="http://www.apistore.co.kr/api/apiList.do">http://www.apistore.co.kr/api/apiList.do</a>

# \*. 전수업리뷰

## ⌚ 네이버 API - 요청과 응답

<https://developers.naver.com/docs/serviceapi/search/news/news.md#뉴스>



구분	내용 및 설명	
URL	뉴스	<a href="https://openapi.naver.com/v1/search/news.json">https://openapi.naver.com/v1/search/news.json</a>
	블로그	<a href="https://openapi.naver.com/v1/search/blog.json">https://openapi.naver.com/v1/search/blog.json</a>
	카페	<a href="https://openapi.naver.com/v1/search/cafearticle.json">https://openapi.naver.com/v1/search/cafearticle.json</a>
	영화	<a href="https://openapi.naver.com/v1/search/movie.json">https://openapi.naver.com/v1/search/movie.json</a>
	쇼핑	<a href="https://openapi.naver.com/v1/search/shop.json">https://openapi.naver.com/v1/search/shop.json</a>

데이터 요청 주소

# \*. 전수업리뷰

## ⌚ 네이버 API - 요청과 응답

### news 의 요청과 응답

요청 변수	query	검색을 원하는 문자열이며 UTF-8로 인코딩한다.
	start	검색 시작 위치로 최대 1000까지 가능하다. 1(기본값)~1000(최대값)
	display	검색 결과 출력 건수를 지정한다. 10(기본값)~100(최대값)
응답 변수	items	검색 결과로 title, originallink, link, description, pubDate를 포함한다.
	title	검색 결과 문서의 제목이다.
	link	검색 결과 문서를 제공하는 네이버의 하이퍼텍스트 link다.
	originallink	검색 결과 문서를 제공하는 언론사의 하이퍼텍스트 link다.
	description	검색 결과 문서의 내용을 요약한 정보다.
	pubDate	검색 결과 문서가 네이버에 제공된 시간이다.

# \*. 전수업리뷰

## ⌚ 네이버 API - 뉴스 크롤링

### ■ 전체 작업 설계

작업 설계	사용할 코드
1. 검색어 지정하기	srcText = '월드컵'
2. 네이버 뉴스 검색하기	<b>getNaverSearch()</b>
2.1 url 구성하기	url = base + node + srcText
2.2 url 접속과 검색 요청하기	urllib.request.urlopen()
2.3 요청 결과를 응답 JSON으로 받기	json.load()
3. 응답 데이터를 정리하여 리스트에 저장하기	<b>getPostData()</b>
4. 리스트를 JSON 파일로 저장하기	json.dumps()

cmain.py

CODE2.py

CODE1.py

CODE3.py

# \*. 전수업리뷰

## ⌚ 네이버 API - 뉴스 크롤링

### ■ 프로그램 구성 설계

[CODE 0] **cmain.py**

```
def main()
1. 검색어 지정
2. 네이버 뉴스 검색
3. 응답 데이터 정리 후
   리스트에 저장
4. 리스트를 JSON 파일로 저장
```

**CODE2.py**  
[CODE 2]

getNaverSearch()

json.load(responseDecode)

**CODE1.py**  
[CODE 1]

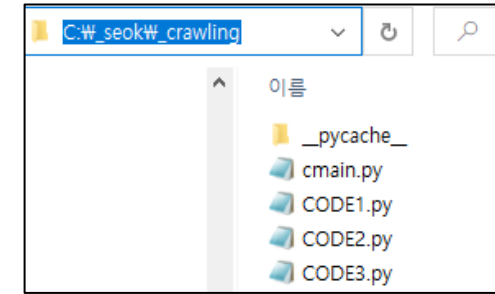
getRequestUrl()

Response.read()

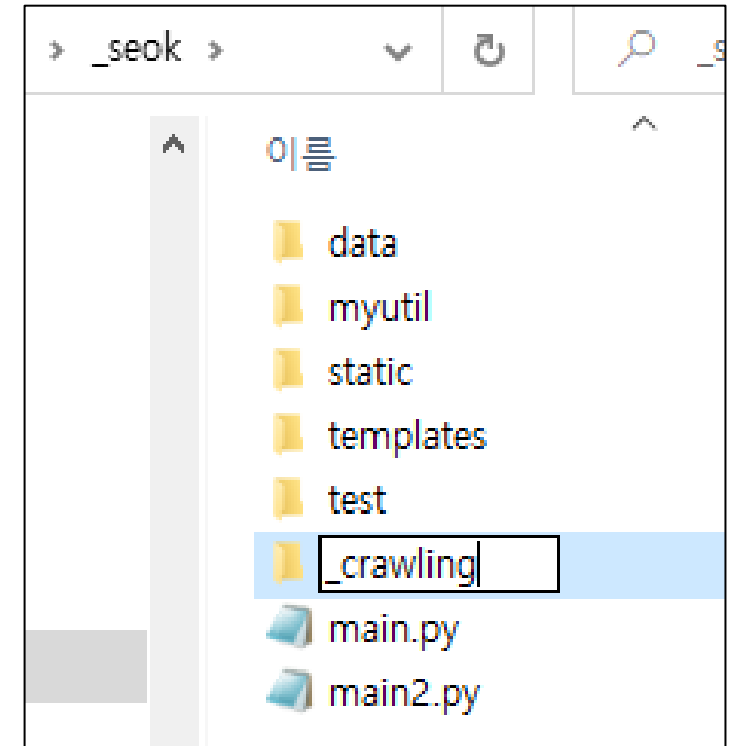
[CODE 3] **CODE3.py**

getPostData()

jsonResult



자신의 홈 디렉터리에



**\_crawling 디렉터리 만들기**



# \*. 전수업리뷰

## ⌚ 네이버 API - 뉴스 크롤링

### ■ 실행

```

C:\> 명령 프롬프트
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\> cd W_seok

C:\W_seok> cd _crawling

C:\W_seok\W_crawling> python cmain.py
검색어를 입력하세요: BTS
----call url = https://openapi.naver.com/v1/search/news.json?query=BTS&
----[2023-10-15 17:27:54.046335] Url Request Success
----call url = https://openapi.naver.com/v1/search/news.json?query=BTS&
----[2023-10-15 17:27:54.140085] Url Request Success
----call url = https://openapi.naver.com/v1/search/news.json?query=BTS&
----[2023-10-15 17:27:54.249461] Url Request Success
----call url = https://openapi.naver.com/v1/search/news.json?query=BTS&
----[2023-10-15 17:27:54.343211] Url Request Success
----call url = https://openapi.naver.com/v1/search/news.json?query=BTS&
----[2023-10-15 17:27:54.436963] Url Request Success
전체 검색 : 436273 건
가져온 데이터 : 50 건
naver_news_BTS(00050) SAVED
C:\W_seok\W_crawling>

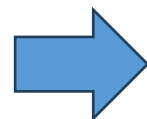
```

cd W\_seok

cd \_crawling

python cmain.py

파일이 생성 됨



naver\_news\_BTS(00050).json



naver\_news\_BTS(00050).xlsx

# \*. 전수업리뷰


## ⌚ 네이버 API - 뉴스 크롤링


### ■ 확인

메모장이나  
vsc 로 확인

```
naver_news_BTS(00050).json - Windows 메모장
파일(F) 편집(E) 서식(O) 보기(V) 도움말(H)

[
  {
    "cnt": 1,
    "title": "\"누님들 감사합니다\" 한덕수 총리에 도착한 '깜짝선물' 정체",
    "description": "그러면서 \"어르신들은 이번 엑스포 응원가를 만드느라 노랏말도 직접 쓰시고, 뮤직비디오도 직접 쓰시고, 뮤직비디오 찍기 전에 일주일간 안무 연습도 하셨다고 한다\"며 <b>BTS</b>와 블랙핑크만 엑스포를 홍보할 수 있는 게 아니라는 수니와...
    "org_link": "https://www.seoul.co.kr/news/newsView.php?id=20231015500088&wlog_tag3=naver",
    "link": "https://www.seoul.co.kr/news/newsView.php?id=20231015500088&wlog_tag3=naver",
    "pDate": "2023-10-15 17:23:00"
  }
]
```


naver\_news\_BTS(00050).json


naver\_news\_BTS(00050).xlsx

excel 로 확인

	A	B	C	D	E	F
	cnt	title	description	org_link	link	pDate
1			그러면서 "어르신들은 이번 엑스포 응원가를 만드느라 노랏말도 직접 쓰시고, 뮤직비디오 찍기 전에 일주일간 안무 연습도 하셨다고 한다"며 <b>BTS</b>와 블랙핑크만 엑스포를 홍보할 수 있는 게 아니라는 수니와...	https://www.seoul.co.kr/news/newsView.php?id=20231015500088&wlog_tag3=naver	https://www.seoul.co.kr/news/newsView.php?id=20231015500088&wlog_tag3=naver	2023-10-15 17:23:00
2	1	"누님들 감사합니다" 한덕수 총리에 도착한 '깜짝선물' 정체	사진=수원시청 '방탄소년단(<b>BTS</b>) 뷔'와 '생태교통 수원 뉴페스타 홍보 그림'이 행궁동 '화성사업소'에 벽화... 이 작가는 도산 안창호, 안중근 의사, <b>BTS</b> 등의 그래피티를 국내와 해외에 선보여 대중에게 알려졌다. 이재준...	http://www.joongboo.com/news/articleView.html?idxno=363615554	http://www.joongboo.com/news/articleView.html?idxno=363615554	2023-10-15 17:14:00
3	2	행궁동 화성사업소에 '<b>BTS</b>' '생태교통 수원 뉴페스타' 벽화 제작	이날 촬영은 <b>BTS</b> RM, 이효리, 조인성, 배두나, 공효진 등 톱스타들과 작업했던 유명 포토그래퍼와 함께해 눈길을 끌었다. 촬영이 시작되자 흥현희와 준범이는 금세 눈빛이 돌변, 웃음기를 속 빼고 치명적인 매력을 발산했다....	https://www.newsen.com/news_view.php?uid=202310151649596710	https://www.newsen.com/news_view.php?uid=202310151649596710	2023-10-15 16:53:00
4	4	'전참시' 이국주 송국영고 이효리 증가 영상 공개..최고 시청률 4.8%				

# \*. 전수업리뷰

## ⌚ 공공데이터 API - 요청과 응답

요청주소 : <http://openapi.tour.go.kr/openapi/service/EdrcntTourismStatsService/getEdrcntTourismStatsList>

요청변수(Request Parameter)					
항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
연월	YM	12	필수	201201	연월
국가코드	NAT_CD	6	옵션	112	국가코드
출입국구분코드	ED_CD	2	옵션	E	출입국구분코드

# \*. 전수업리뷰

## ⌚ 공공데이터 API - 요청과 응답

출력결과(Response Element)					
항목명(국문)	항목명(영문)	항목크기	항목구분	샘플데이터	항목설명
결과코드	resultCode	4	필수	0000	결과코드
결과메시지	resultMsg	50	필수	OK	결과메시지
한 페이지 결과 수	numOfRows	2	옵션	10	한 페이지 결과 수
페이지 번호	pageNo	5	옵션	1	페이지 번호
전체 결과 수	totalCount	7	옵션	12334	전체 결과 수
출입국 구분	ed	14	필수	방한외래관광객	출입국구분
출입국 구분코드	edCd	2	필수	E	출입국 구분코드
국가코드	natCd	6	필수	112	국가코드
국가	natKorNm	80	필수	중국	국가
출입국자수	num	10	옵션	179508	출입국자수
연월	ym	12	필수	201206	연월
결과값 연번	rnum	2	필수	1	결과값 나열순서

# \*. 전수업리뷰

## ⌚ 공공데이터 API - 전체 작업 설계

작업 설계	사용할 코드
1. 데이터를 수집할 국가코드와 연도 입력하기	national_code, nStartYear, nEndYear
2. 데이터 수집 요청하기	<b>getTourismStatsService()</b>
2.1 url 구성하여 데이터 요청하기	<b>getTourismStatsItem()</b>
2.2 url 접속하고 요청하기	<b>getRequestUrl()</b>
2.3 응답 데이터를 리스트로 구성하기	jsonResult, result
3. 데이터를 JSON 파일과 CSV 파일로 저장하기	json.dumps(), to_csv()



# \*. 전수업리뷰

## ⌚ 공공데이터 API - 프로그램 구성 설계

gmain.py

[CODE 0]

def main()

1. 수집할 국가 코드와  
연도 입력

2. 데이터 수집 요청

3. 파일 저장

[CODE 3]

getTourismStatsService()

데이터 리스트

월 데이터 요청

json.loads()

[CODE 1]

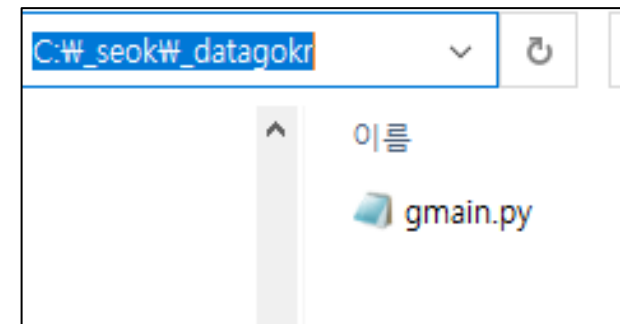
getRequestUrl()

접속할 url

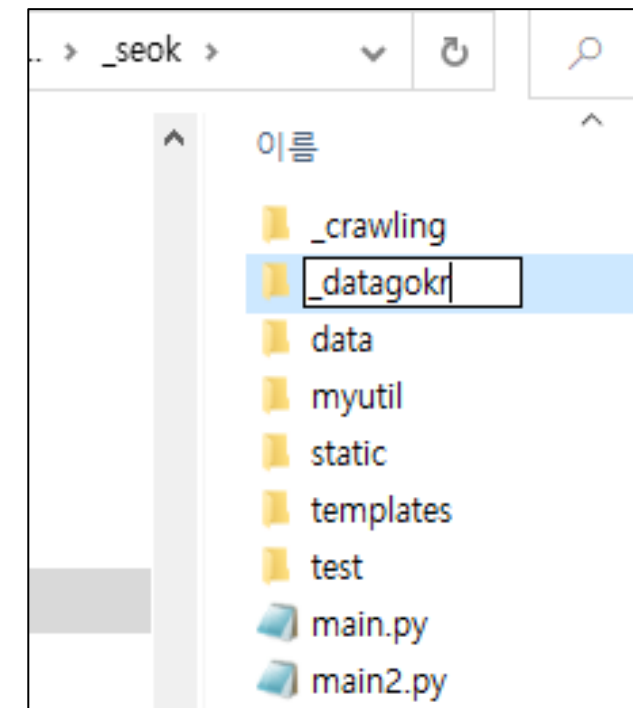
response.read()

[CODE 2]

getTourismStatItem()



자신의 홈 디렉터리에



\_datagokr 디렉터리 만들기

# \*. 전수업리뷰

## ⌚ 공공데이터 API - 실행

명령 프롬프트 - python gmain.py

```
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Wdossa>cd W_seok

C:\W_seok>cd _datagokr

C:\W_seok\W_seok_datagokr>python gmain.py
<< 국내 입국한 외국인의 통계 데이터를 수집합니다. >>
국가 코드를 입력하세요(중국: 112 / 일본: 130 / 미국: 275) : 112
데이터를 몇 년부터 수집할까요? : 2019
데이터를 몇 년까지 수집할까요? : 2021
```

cd W\_seok

cd \_datagokr

python gmain.py

112

2019

2021

# \*. 전수업리뷰

## ⌚ nodejs

### ■ nodejs 란

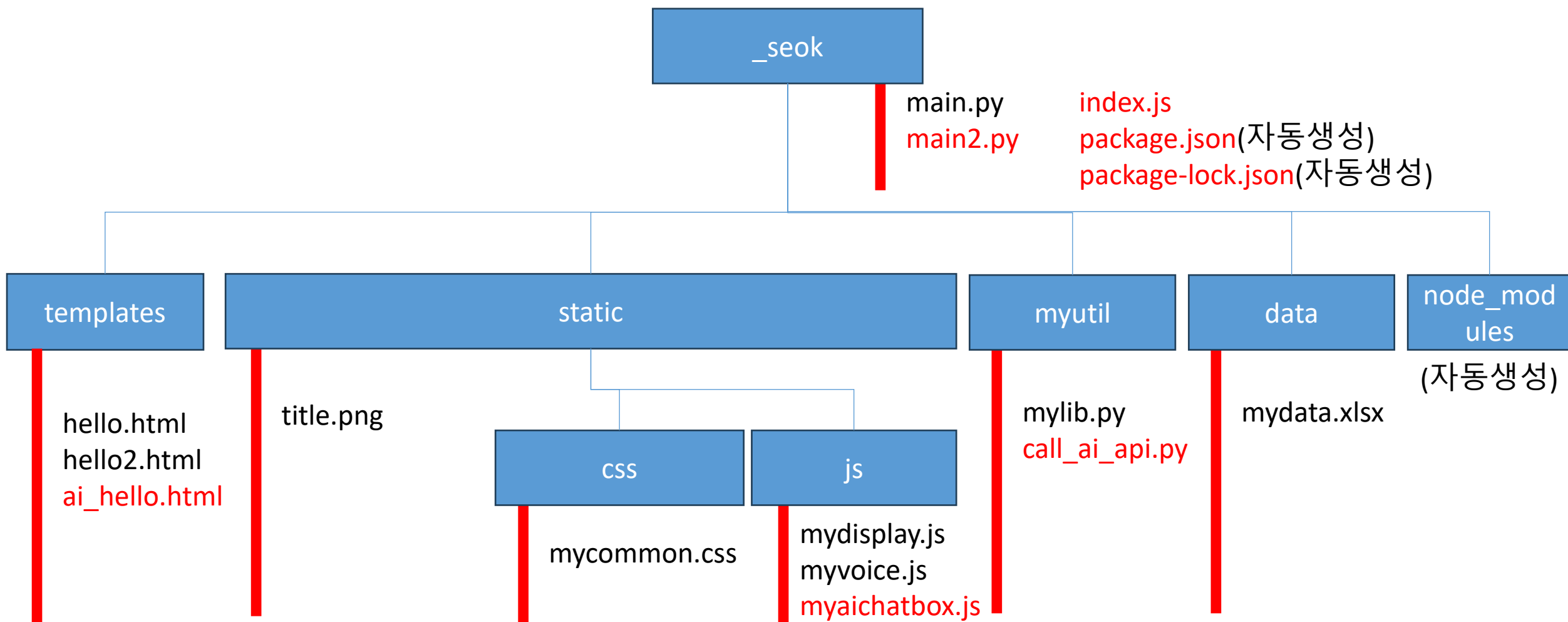
- 오픈 소스 JavaScript 엔진인 크롬 V8에 비동기 이벤트 처리 라이브러리인 libuv를 결합한 플랫폼
- JavaScript로 브라우저 밖에서 서버를 구축하는 등의 코드를 실행할 수 있게 해주는 런타임 환경
- 빈번한 I/O처리에 있어서의 우수한 성능, 서버 확장의 용이성
- JavaScript라는 프론트엔드 필수 언어로 백엔드까지 작성할 수 있다는 엄청난 장점
- 특히 넷플릭스처럼 엄청나게 많은 양의 인풋 아웃풋 데이터를 처리해야 하는 서비스에 있어서 강점

### ■ npm(Node Package Manager)

- node.js 를 위한 패키지 매니저이면서 오픈소스 생태계 관리자
- package.json 문서를 활용하여 패키지 생태계의 명세서(패키지의 종류들과 버전) 를 관리
- package-lock.json 은 패키지의 의존성에 대한 정확하고 구체적인 정보를 가지고 있음

# \*. 전수업리뷰

⌚ nodejs + Flask 디렉토리 구성



# \*. 전수업리뷰

## ⌚ nodejs + Flask - 실행

### ■ nodejs 실행

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dossa>cd \_seok

C:\_seok>npm start

> myai@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
Server listening on port 5555
```

cd \\_seok

npm start

### ■ Flask 실행

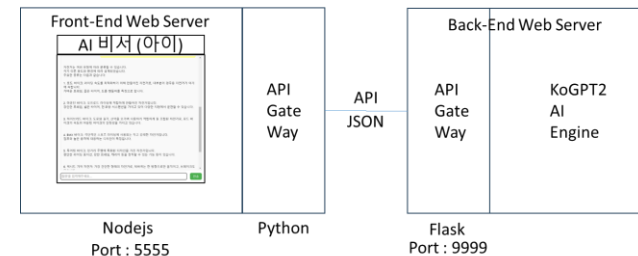
```
C:\_seok>명령 프롬프트 - python main2.py

Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\_seok>python main2.py
* Serving Flask app 'main2'
* Debug mode: off
WARNING: This is a development server. Do not use without proper
security measures.
* Running on http://172.16.11.220:9999
Press CTRL+C to quit
```

cd \\_seok

python main2.py

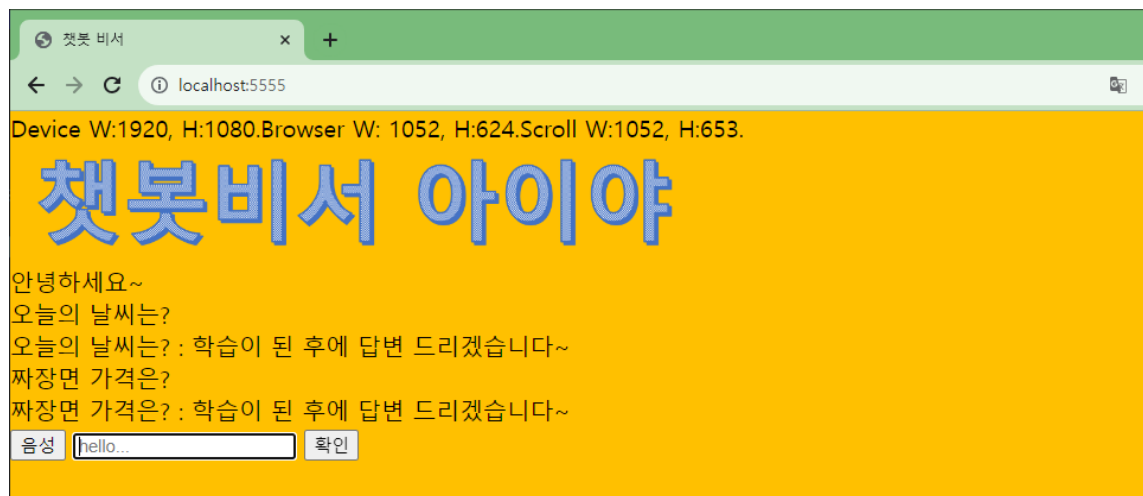
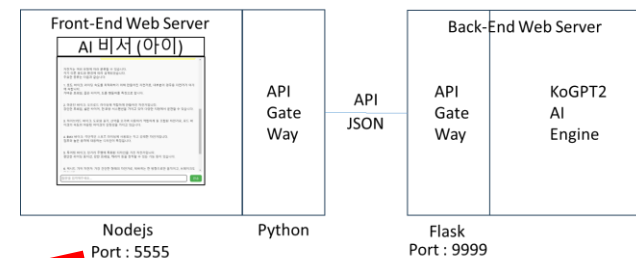




# \*. 전수업리뷰

⌚ nodejs + Flask - 접속

http://localhost:5555/



```

(c) 명령 프롬프트 - python main2.py
(c) Microsoft Corporation. All rights reserved.
C:\Users\wdossa>cd \_seok
C:\_seok>python main2.py
* Serving Flask app 'main2'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a prod
* Running on http://172.16.11.220:9999
Press CTRL+C to quit
172.16.11.220 - - [16/Oct/2023 18:44:03] "GET / HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:03] "GET /static/css/mycommon.css HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:03] "GET /static/title.png HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:03] "GET /static/js/myvoice.js HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:03] "GET /static/js/mydisplay.js HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:03] "GET /favicon.ico HTTP/1.1" 404 -
***input_data : aaa
172.16.11.220 - - [16/Oct/2023 18:44:18] "POST /get_data HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:20] "GET / HTTP/1.1" 200 -
172.16.11.220 - - [16/Oct/2023 18:44:20] "GET /static/js/mydisplay.js HTTP/1.1" 304 -
172.16.11.220 - - [16/Oct/2023 18:44:20] "GET /static/css/mycommon.css HTTP/1.1" 304 -
172.16.11.220 - - [16/Oct/2023 18:44:20] "GET /static/title.png HTTP/1.1" 304 -
172.16.11.220 - - [16/Oct/2023 18:44:20] "GET /static/js/myvoice.js HTTP/1.1" 304 -
***question : 오늘의 날씨는?
***answer : 오늘의 날씨는? : 학습이 된 후에 답변 드리겠습니다~
172.16.11.220 - - [16/Oct/2023 18:44:37] "POST /api/get_data HTTP/1.1" 200 -
***question : 짜장면 가격은?
***answer : 짜장면 가격은? : 학습이 된 후에 답변 드리겠습니다~
172.16.11.220 - - [16/Oct/2023 18:44:55] "POST /api/get_data HTTP/1.1" 200 -
    
```

# \*. 전달 사항



교재

주교재

- PowerPoint 로 만든 pdf 자료
- 데이터 과학 기반의 파이썬 빅데이터 분석 (이지영 지음, 한빛아카데미)

부교재

- 필요 시, 영상 공유



# \*. 전달 사항

## RoadMap

### Hadoop설치

- ✓ VM 셋업
- ✓ JDK
- ✓ Python
- ✓ Hadoop Engine
- ✓ Spark Engine
- ✓ Zeppelin

### 빅데이터분석

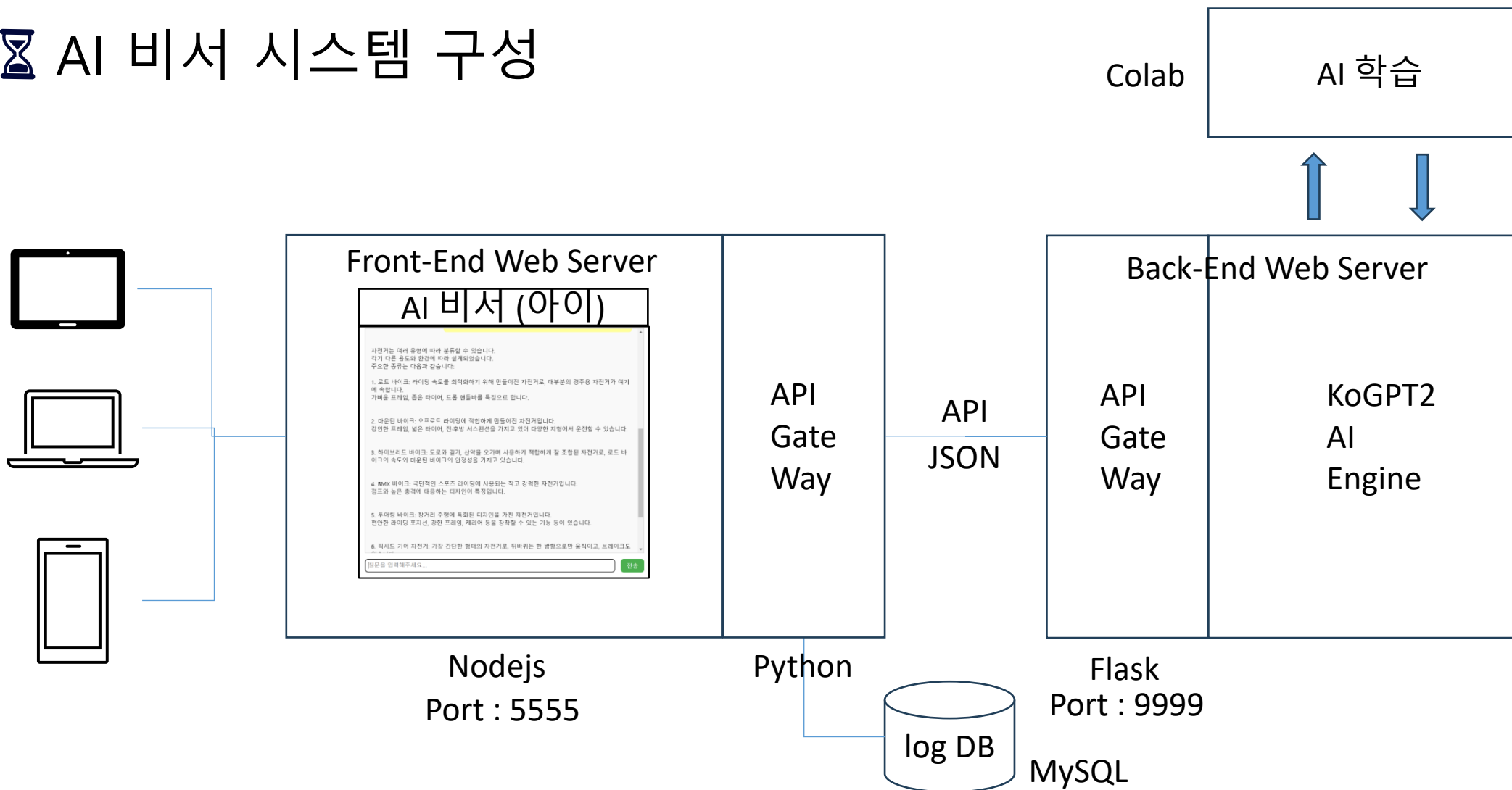
- ✓ 빅데이터 산업의 이해
- ✓ 파이썬 프로그래밍
- ✓ 크롤링
- ✓ 통계분석
- ✓ 텍스트빈도분석
- ✓ 지리정보분석
- ✓ 회귀분석/분류분석
- ✓ 텍스트마이닝

### AI 비서학습

- ✓ 챗봇 데이터 수집
- ✓ Flask 웹서버
- ✓ Nodejs API 연동
- ✓ KoGPT2 환경구성
- ✓ Colab을 이용한 학습
- ✓ 말풍선생성기 활용
- ✓ MySQL
- ✓ 챗봇 비서 만들기

# \*. 전달 사항

## ⌚ AI 비서 시스템 구성



# 1. 라이브러리 크롤링

## Selenium

### ■ 의의

- Selenium(셀레니움) 은 사용자가 아닌 프로그램이 웹 브라우저를 제어할 수 있게 하는 해 주는 라이브러리
- Selenium 은 서버와 클라이언트로 나누는데, 웹 브라우저 종류 마다 클라이언트 프로그램이 별도로 필요(ChromeDriver, Microsoft WebDriver)
- Browser Driver 는 웹 브라우저와 Selenium 서버간 통신을 위한 인터페이스 도구



# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 설치

CMD 창을 띄워서

```

명령 프롬프트
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dossa>python -m pip install selenium
Collecting selenium
  Downloading selenium-4.14.0-py3-none-any.whl (9.9 MB)
----- 9.9/9.9 MB 9.6 MB/s
Requirement already satisfied: certifi>=2021.10.8 in c:\applicat
7)
Collecting trio-websocket~0.9
  Downloading trio_websocket-0.11.1-py3-none-any.whl (17 kB)

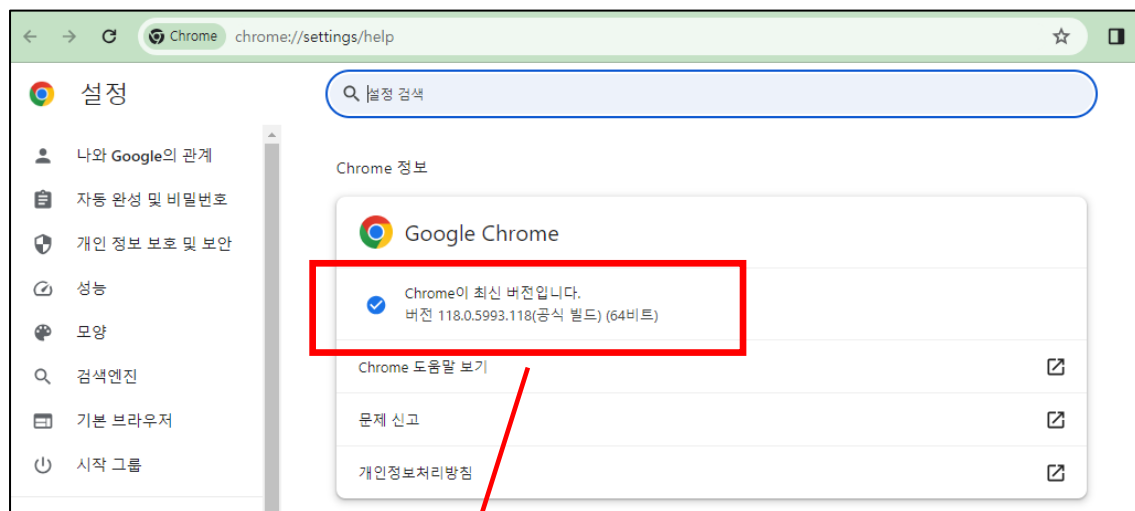
```

python -m pip install selenium

# 1. 라이브러리 크롤링

## ⌚ Selenium

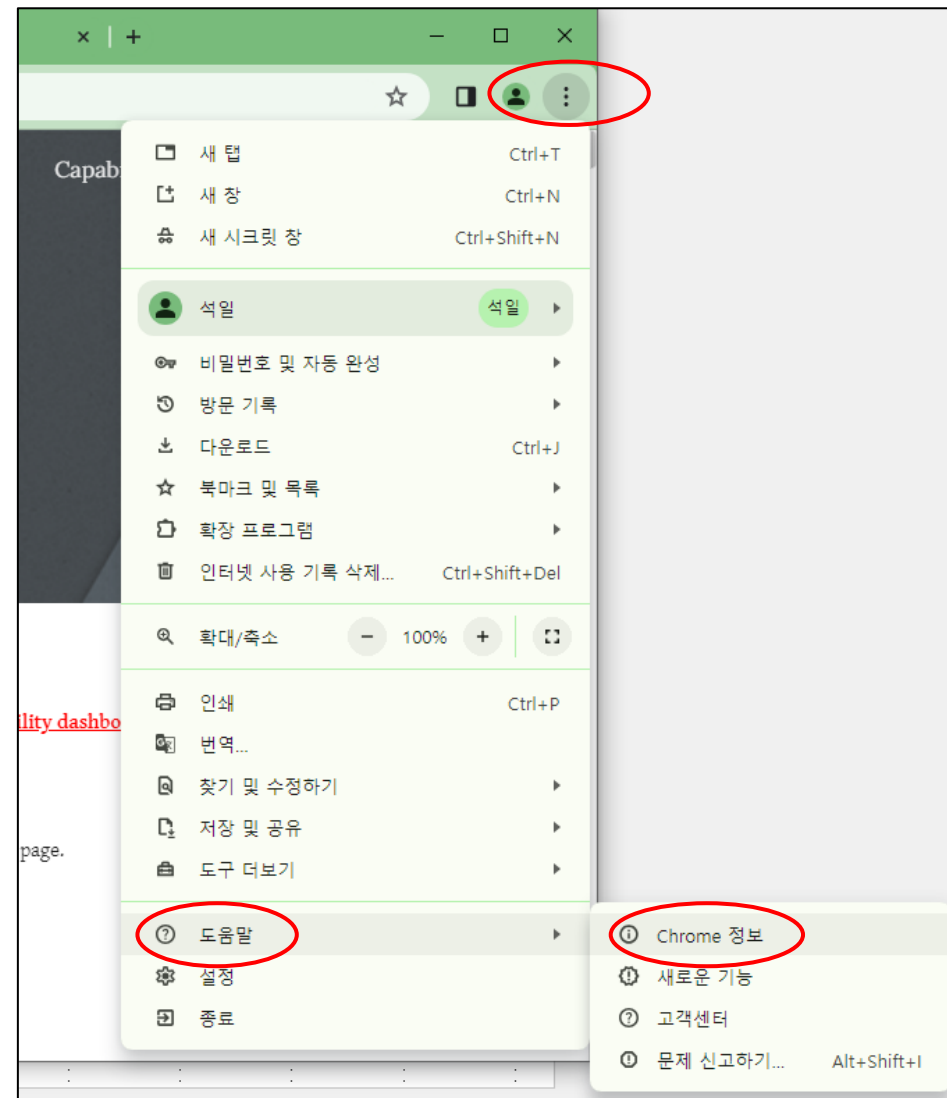
### ■ 크롬 버전 확인



버전을 기록해 둬

118.0.5993

크롬 브라우저 실행 후,

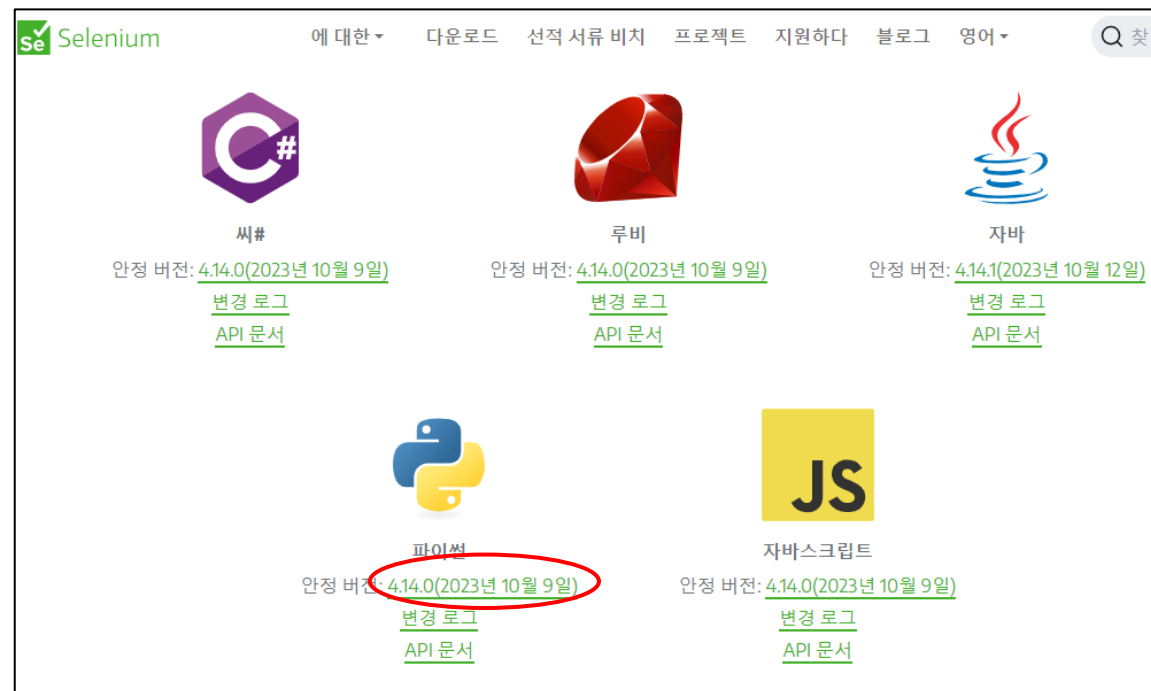
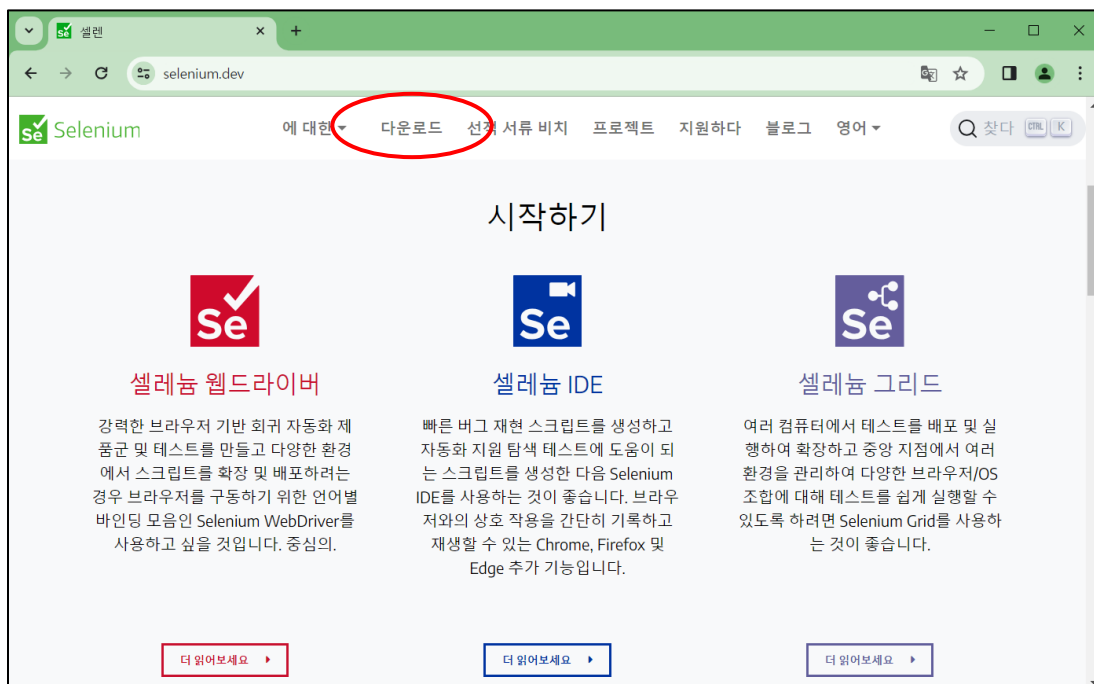


# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 웹드라이버 설치

<https://www.selenium.dev/>



# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 웹드라이버 설치

**포옹**

**임티어니**

**이세메라우**

**마이크로더**

**테베카**

**티투스포르너**

---

**분류자**

**개발현황**

- 5 - 생산/안정

**대상 청중**

- 개발자

**특히**

- OSI 승인 :: Apache 소프트웨어 라이선스

Selenium은 선택한 브라우저와 인터페이스하려면 드라이버가 필요합니다. 예를 들어 Firefox에는 아래 예제를 실행하기 전에 설치해야 하는 [geckodriver](#)가 필요합니다. PATH에 있는지 확인하십시오. 예를 들어 /usr/bin 또는 /usr/local/bin에 배치하십시오.

이 단계를 준수하지 않으면 selenium.common.Exceptions.WebDriverException 오류가 발생합니다. 메시지: 'geckodriver' 실행 파일이 PATH에 있어야 합니다.

기타 지원되는 브라우저에는 자체 드라이버를 사용할 수 있습니다. 더 널리 사용되는 일부 브라우저 드라이버에 대한 링크는 다음과 같습니다.

크롬 :	<a href="https://chromedriver.chromium.org/downloads">https://chromedriver.chromium.org/downloads</a>
가장자리 :	<a href="https://developer.microsoft.com/en-us/microsoft-edge/tools/webdriver/">https://developer.microsoft.com/en-us/microsoft-edge/tools/webdriver/</a>
파이어폭스 :	<a href="https://github.com/mozilla/geckodriver/releases">https://github.com/mozilla/geckodriver/releases</a>
사파리 :	<a href="https://webkit.org/blog/6900/webdriver-support-in-safari-10/">https://webkit.org/blog/6900/webdriver-support-in-safari-10/</a>

**예시 0:**

- 새 Firefox 브라우저를 열어주세요
- 주어진 URL에서 페이지를 로드합니다.

ChromeDriver - WebDriver for Chro... ChromeDriver Capabilities & ChromeC

### Current Releases

- If you are using Chrome version 115 or newer, please [consult the Chrome for Testing availability dashboard](#). This page provides endpoints for specific ChromeDriver version downloading.
- For older versions of Chrome, please see below for the version of ChromeDriver that supports it.

For more information on selecting the right version of ChromeDriver, please see the [Version Selection](#) page.

**ChromeDriver 114.0.5735.90**  
Supports Chrome version 114  
For more details, please see the [release notes](#).

**ChromeDriver 114.0.5735.16**  
Supports Chrome version 114

없으면, 여기 클릭

자신의 크롬브라우저 버전과  
동일한 버전을 찾음

# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 웹드라이버 설치

chromedriver / win64 확인 후,  
마우스로 긁어서 링크 복사

googlechromelabs.github.io/chrome-for-testing/

**테스트용 Chrome 가용성**

이 페이지에는 Chrome 출시 채널별로 사용 가능한 최신 크로스 플랫폼 테스트용 Chrome 버전 및 자산이 나열되어 있습니다.  
테스트용 Chrome을 기반으로 출시 데이터를 기반으로 자동화된 스크립트를 구축하려는 경우 [JSON API 엔드포인트](#)를 참조하세요.  
마지막 업데이트 @2023-10-31T12:09:33.184Z

채널	버전	개정	상태
<u>안정적인</u>	118.0.5993.70	r1192594	✓
<u>안정적(예정)</u>	118.0.5993.117	r1192594	✗
<u>베타</u>	119.0.6045.59	r1204232	✓

chromedriver	mac-x64	x64/chromedriver-mac-x64.zip	200
chromedriver	win32	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/win32/chromedriver-win32.zip	200
chromedriver	win64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/win64/chromedriver-win64.zip	200

Upcoming version: 118.0.5993.117 (r1192594)

Binary	Platform	URL	HTTP status
chrome	linux64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/linux64/chrome-linux64.zip	404
chrome	mac-arm64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/mac-arm64/chrome-mac-arm64.zip	404
chrome	mac-x64	https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.117/mac-x64/chrome-mac-x64.zip	404

# 1. 라이브러리 크롤링

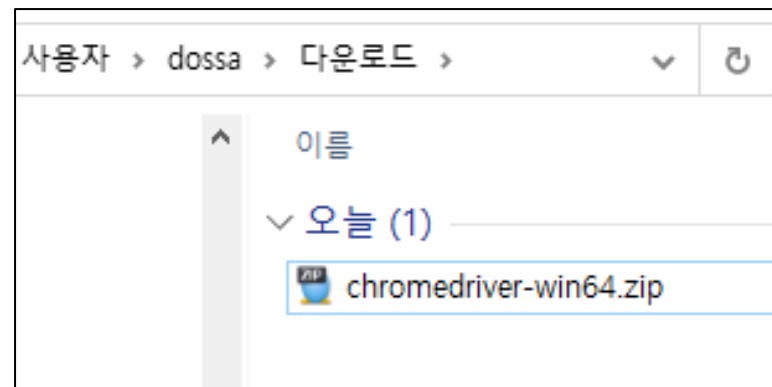
## ⌚ Selenium

### ■ 웹드라이버 설치



Binary	Platform	URL
chromedriver	win32	<a href="https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/win32/chromedriver-win32.zip">https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/win32/chromedriver-win32.zip</a>
chromedriver	win64	<a href="https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/win64/chromedriver-win64.zip">https://edgedl.me.gvt1.com/edgedl/chrome/chrome-for-testing/118.0.5993.70/win64/chromedriver-win64.zip</a>

Upcoming version: 118.0.5993.117 (r1192594)



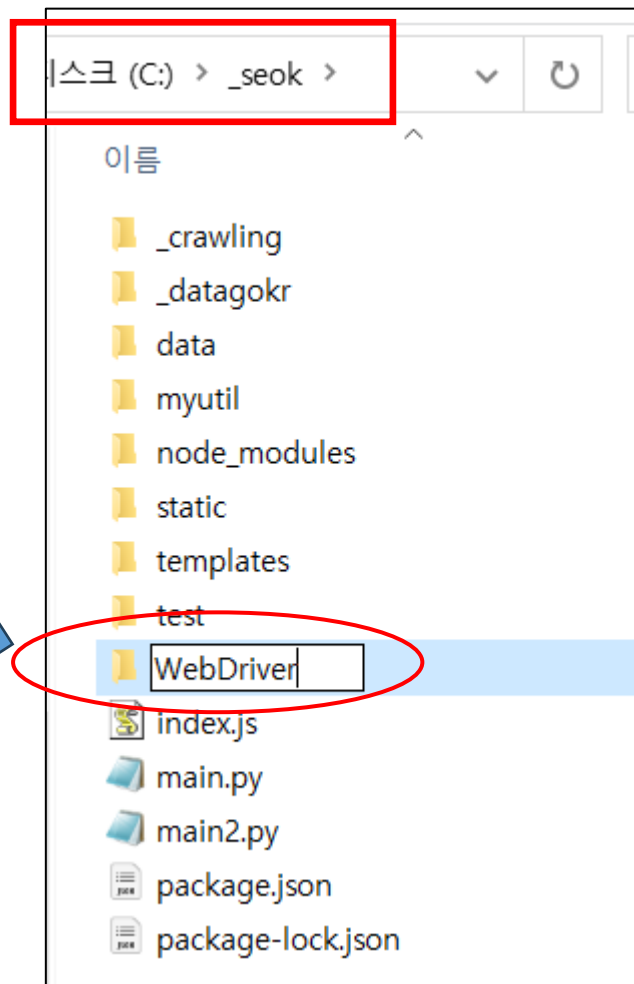
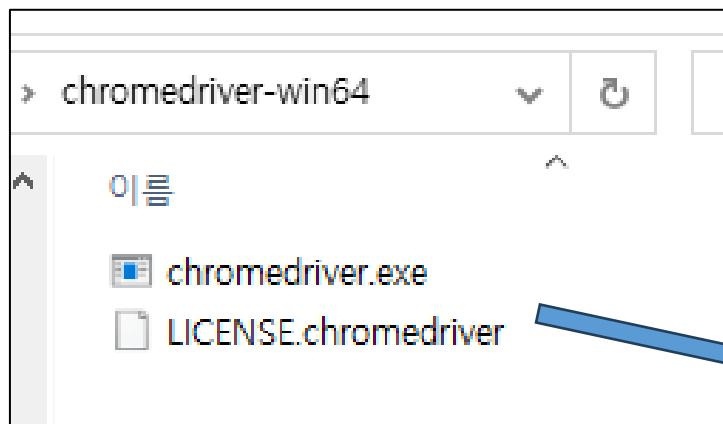
다운로드가 되면 압축을 푼다

브라우저 주소창에 붙여 넣기

# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 웹드라이버 설치



자신의 디렉토리 밑에  
WebDriver 폴더 생성 후,

chromedriver.exe 와 라이선스  
파일을 이동



# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 크롤링 실습

4칸 8칸 12칸

```
#-----
# 크롬드라이버를 활용한 크롤링
#-----
from bs4 import BeautifulSoup
import urllib.request
import pandas as pd
import datetime
import time

from selenium import webdriver
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys
from selenium.webdriver.chrome.options import Options

chrome_options = Options()
chrome_options.add_experimental_option("detach", True)
```

주식 - 설명문

Selenium 패키지의 WebDreiver를 импорт

# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 크롤링 실습

4칸 8칸 12칸

```
# 불필요한 에러 메시지 없애기
chrome_options.add_experimental_option("excludeSwitches", ["enable-logging"])

# 브라우저 생성
CoffeeBean_URL = "https://www.coffeebeankorea.com/store/store.asp"
#wd = webdriver.Chrome("C:\\_seok\\WebDriver\\chromedriver.exe")
wd = webdriver.Chrome(options=chrome_options)
wd.get(CoffeeBean_URL)

#[CODE 1]
def CoffeeBean_store(result):
    for i in range(1, 10): #매장 수만큼 반복
        wd.get(CoffeeBean_URL)
        time.sleep(2) #웹페이지 연결할 동안 1초 대기
        try:
            wd.execute_script("storePop2(%d)" %i)
            time.sleep(2) #스크립트 실행할 동안 2초 대기
            html = wd.page_source
```

자바스크립트 함수 호출해 매장 정보 페이지 열기

자바스크립트 함수가 수행된 페이지의 소스 코드를 저장

# 1. 라이브러리 크롤링

## ⌚ Selenium

### ■ 크롤링 실습

4칸 8칸 12칸

```
soupCB = BeautifulSoup(html, 'html.parser')
store_name_h2 = soupCB.select("div.store_txt > h2")
store_name = store_name_h2[0].string
print(store_name) #매장 이름 출력하기
store_info = soupCB.select("div.store_txt > table.store_table > tbody > tr > td")
store_address_list = list(store_info[2])
store_address = store_address_list[0]
store_phone = store_info[3].string
result.append([store_name]+[store_address]+[store_phone])
except:
    continue
return
```

<div class="store\_txt"> 태그 내부의 <h2> 태그  
➔ <h2>학동역</h2>

<table class="store\_table"> 태그 내부의 <td> 태그

➔ <td>평일 : 07:00~23:00 | 주말 : 08:00~22:00</td>,  
<td>DT(드라이브 스루) 매장입니  
다.</td>,  
<td>서울시 강남구 학동로 211 1층 <!--span  
class="lot">(서울시 강남구 학동로  
211 1층)</span--></td>,  
<td>02-3444-0000</td>]

# Selenium

c:\w\_seok\w\_crawling 밑에

## clibmain.py 로 저장

## ■ 크롤링 실습

4칸      8칸      12칸

```
#[CODE 0]
def main():
    result = []
    print('CoffeeBean store crawling >>>>>>>>>>>>>>>>>>>')
    CoffeeBean_store(result)                                #[CODE 1]

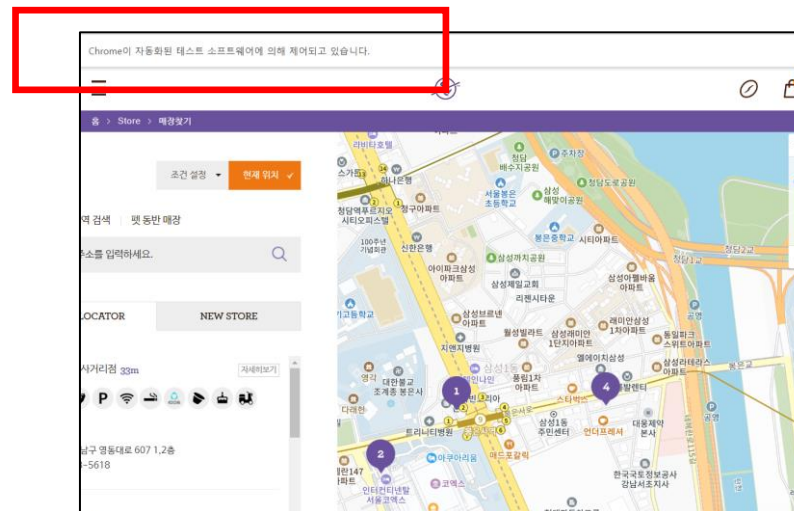
    CB_tbl = pd.DataFrame(result, columns = ('store', 'address','phone'))
    CB_tbl.to_csv('./CoffeeBean.csv', encoding = 'cp949', mode = 'w', index = True)

if __name__ == '__main__':
    main()
```

# Selenium

## ■ 실행

C:\\_명령어\명령어

[illegible]


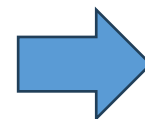

```
cd W_seok
```

cd\_crawling

```
python clibmain.py
```

A	B	C	D
	store	address	phone
0	차병원점	서울시 강남구 논현로 566 강남차병원1층	02-538-7615
1	강남대로점	서울시 서초구 강남대로 369 1층	02-588-5778

파일이 생성 됨

 CODE3.py CoffeeBean.csv

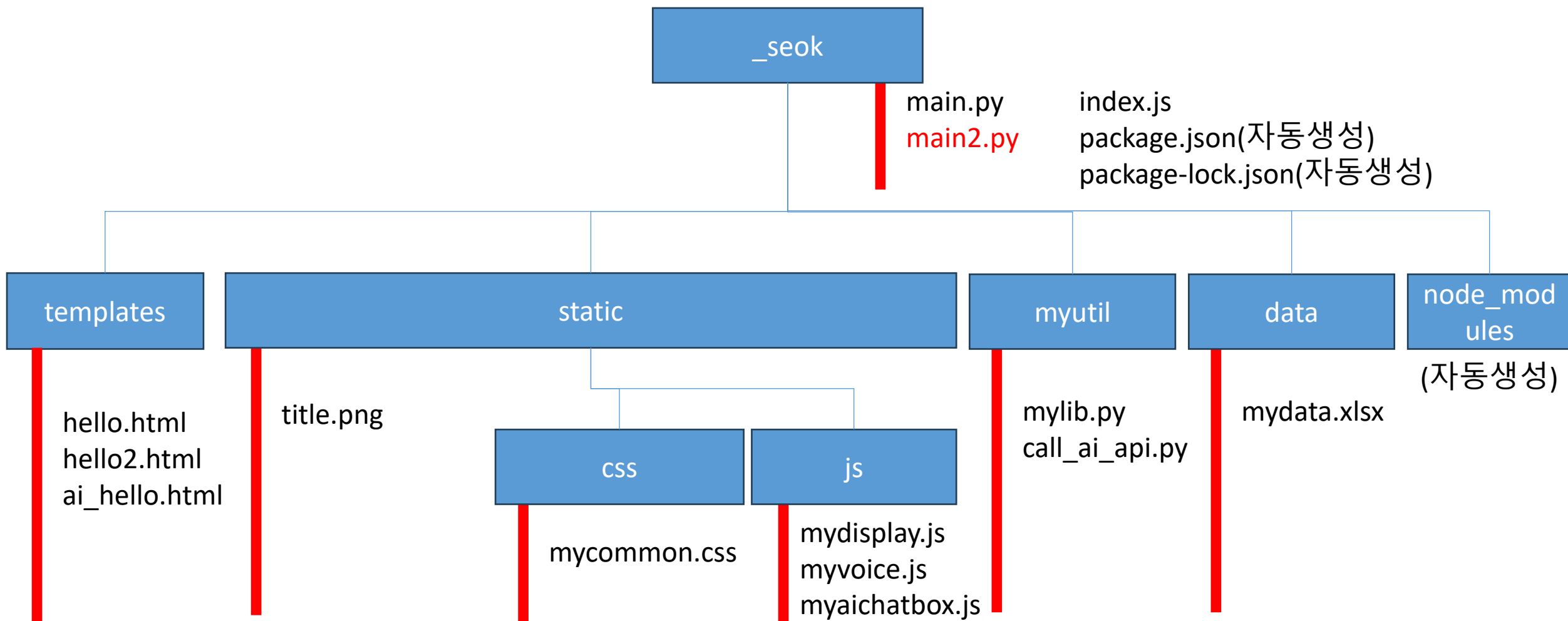
## 2. KoGPT 를 활용한 챗봇 환경 구축

### KoGPT

- GPT(Generative Pre-trained Transformer)
  - 글로벌 AI 연구기관인 OpenAI 사에서 개발한 AI모델
  - 생성형 AI 로 딥러닝을 사용하여 인간과 유사한 텍스트를 생성하는 대규모 자연어 기술
  - 이전 단어나 문자를 기반으로 다음 단어나 문자를 예측하고 텍스트요약, 질문에 대한 답변, 챗봇생성 등 다양한 작업을 수행
  - KoGPT는 카카오브레인에서 만든 GPT-3 모델의 한국어 특화 버전
  - 60억개의 매개변수와 2000억개 토큰의 한국어 데이터를 바탕으로 구축
  - KoGPT는 오픈소스 커뮤니티 깃허브(github) 에 2021년 공개

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 디렉토리 구성





## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 패키지 설치

<https://pytorch.org/get-started/locally/>

PyTorch

nightly. Please ensure that you have **met the prerequisites below (e.g., numpy)**, depending on your package manager. Anaconda is our recommended package manager since it installs all dependencies. You can also **install previous versions of PyTorch**. Note that LibTorch is only available for C++.

PyTorch Build	Stable (2.1.0)	Preview (Nightly)		
Your OS	Linux	Mac	Windows	
Package	Conda	Pip	LibTorch	Source
Language	Python	C++ / Java		
Compute Platform	CUDA 11.8	CUDA 12.1	ROCm 5.6	CPU
Run this Command:	<pre>pip3 install torch torchvision torchaudio</pre>			

To analyze traffic and optimize your experience, we serve cookies on this site. By clicking or navigating, you agree to allow our usage of cookies. As the current maintainers of this site, Facebook's Cookies Policy applies. Learn more, including about available controls: [Cookies Policy](#).

CMD 창에서 설치

`python -m pip install torch torchvision torchaudio`

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 프로그램코딩

- main2.py 파일 수정 – 아래 **붉은색** 내용 추가

```
#=====
# Flask 웹서버 메인 프로그램
#=====
import socket
import pandas as pd
import requests
from flask import Flask, render_template, request, jsonify
from myutil.mylib import mylib_Read_xlsx_Data, mylib_ViewPage
#-----
# KoGPT
#-----
import torch
from transformers import GPT2LMHeadModel
from transformers import PreTrainedTokenizerFast

device = torch.device('cuda' if torch.cuda.is_available() else 'cpu')
print('***** my device = ', device)
```

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 프로그램코딩

- main2.py 파일 수정 - 아래 **붉은색** 내용 추가

```

4칸 8칸 12칸 16칸
PRETRAINED_MODEL = "skt/kogpt2-base-v2"
tokenizer = PreTrainedTokenizerFast.from_pretrained(PRETRAINED_MODEL,
                                                    bos_token='</s>', eos_token='</s>', unk_token='<unk>',
                                                    pad_token='<pad>', mask_token='<mask>')

model = GPT2LMHeadModel.from_pretrained(PRETRAINED_MODEL)
def request_AI(_req):
    _res = model.generate(_req,
                          max_length=128,
                          repetition_penalty=2.0,
                          pad_token_id=tokenizer.pad_token_id,
                          eos_token_id=tokenizer.eos_token_id,
                          bos_token_id=tokenizer.bos_token_id,
                          use_cache=True)

    return _res
#-----

app = Flask(__name__)

```

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 프로그램코딩

- main2.py 파일 수정 – 아래 **붉은색** 내용 추가

```

4칸 8칸
#=====
@app.route('/', methods=['POST', 'GET'])
def home():
    return render_template('hello2.html')
#=====
@app.route('/get_data', methods=['POST'])
def get_data():
    try:
        input_data = request.form["input_data"]
        print('***input_data : ', input_data)

        _file = './data/mydata.xlsx'
        _list = mylib_Read_xlsx_Data(_file)
        df = pd.DataFrame(_list[1:], columns=_list[0])
        result = mylib_ViewPage(df, input_data)

    return result

```

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 프로그래밍코딩

- main2.py 파일 수정 – 아래 **붉은색** 내용 추가

```

4칸 8칸
except Exception as ee:
    print('***error : ', ee)
#=====
@app.route('/api/get_data', methods=['POST'])
def api_page():

    question = request.json['question']
    print('***question : ', question)
    apikey = request.json['key']
    if apikey != 'AAAAAAAAAAAAABBBCCC111':
        return jsonify({'answer': 'not supported'})

    answer = ""

```

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 프로그램코딩

- main2.py 파일 수정 – 아래 **붉은색** 내용 추가 및 수정

```

4칸 8칸
try:
    input_ids = tokenizer.encode(question, return_tensors='pt')
    generated = request_AI(input_ids)
    answer = tokenizer.decode(generated[0])
    #answer = question + " : 학습이 된 후에 답변 드리겠습니다~"
except Exception as ee:
    answer = "오류가 발생했습니다~" + ee

print('***answer : ', answer)
return jsonify({'answer': answer})
#=====
if __name__ == '__main__':
    #_myip = socket.gethostbyname(socket.gethostname())
    app.run(host='172.16.11.220', port=9999, debug=False)

```

파일 – 저장

\_seok 밑에 main2.py 로 저장

## 2. KoGPT 를 활용한 챗봇 환경 구축

### ⌚ 실행

#### ■ nodejs 실행

```
C:\Windows\system32\cmd.exe
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dossa>cd \_seok

C:\_seok>npm start

> myai@1.0.0 start
> nodemon index.js

[nodemon] 2.0.20
[nodemon] to restart at any time, enter `rs`
[nodemon] watching path(s): *.*
[nodemon] watching extensions: js,mjs,json
[nodemon] starting `node index.js`
Server listening on port 5555
```

cd \\_seok

npm start

#### ■ Flask 실행

```
C:\_seok>명령 프롬프트 - python main2.py

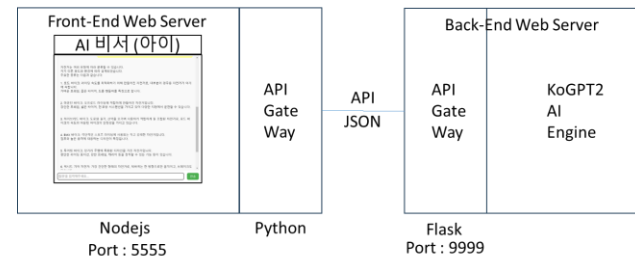
Microsoft Windows [Version 10.0.19045.3570]
(c) Microsoft Corporation. All rights reserved.

C:\Users\dossa>cd \_seok

C:\_seok>python main2.py
* Serving Flask app 'main2'
* Debug mode: off
WARNING: This is a development server. Do not use without proper
security measures.
* Running on http://172.16.11.220:9999
Press CTRL+C to quit
```

cd \\_seok

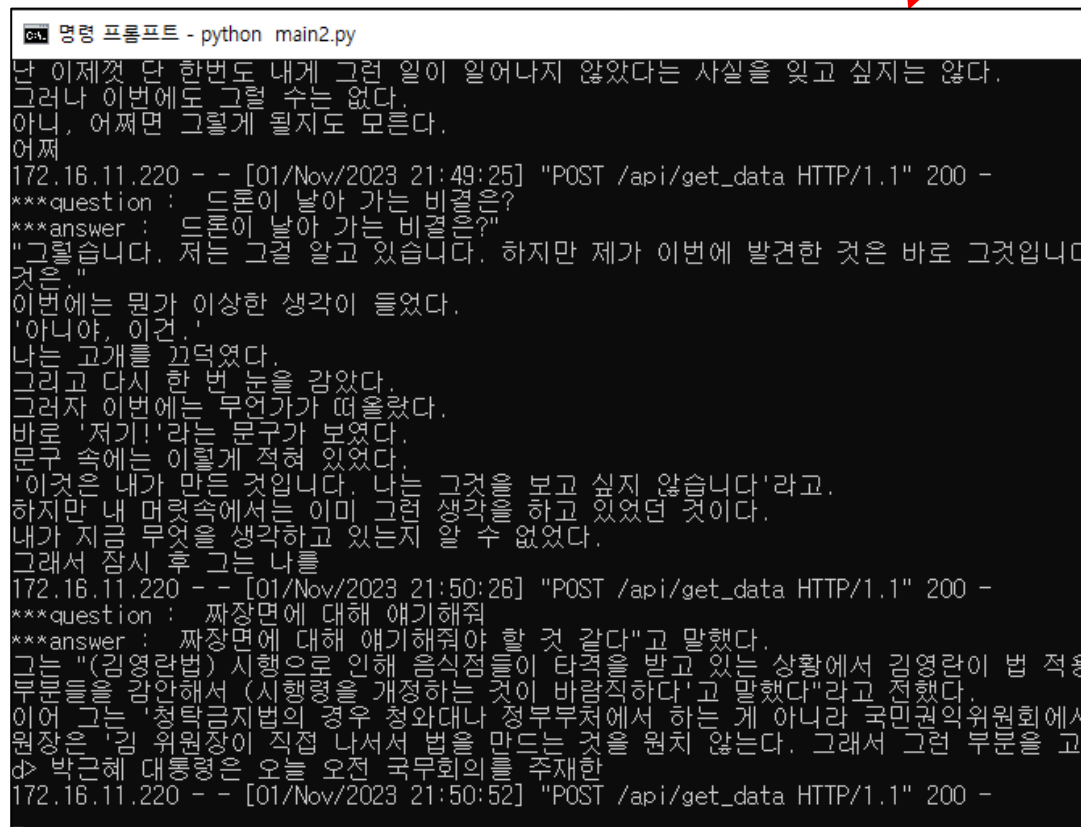
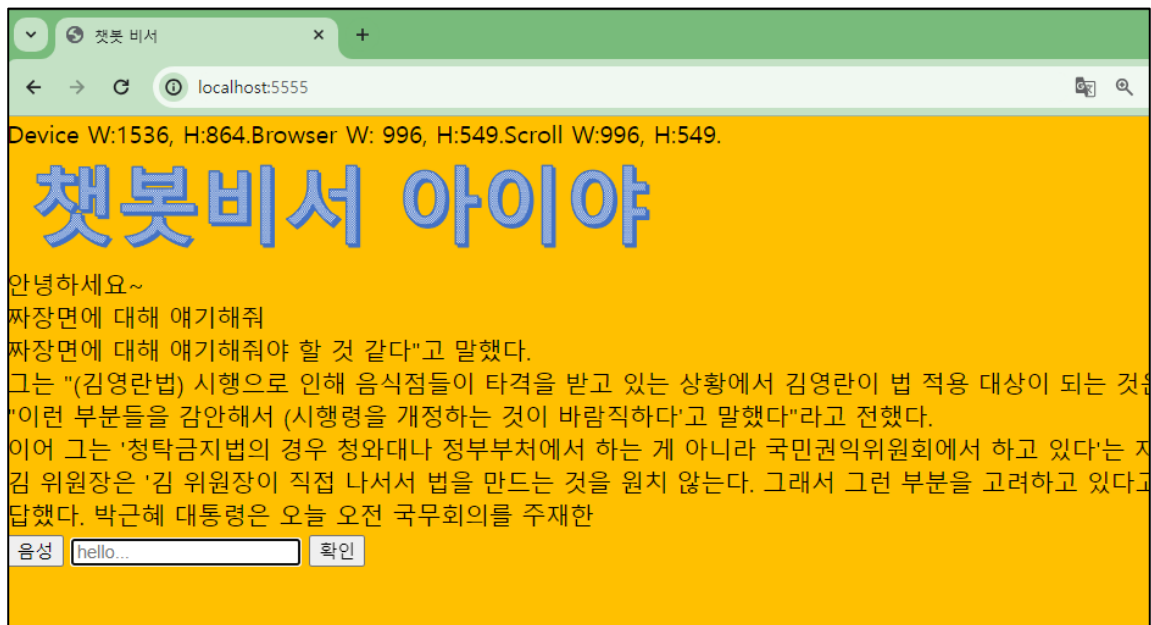
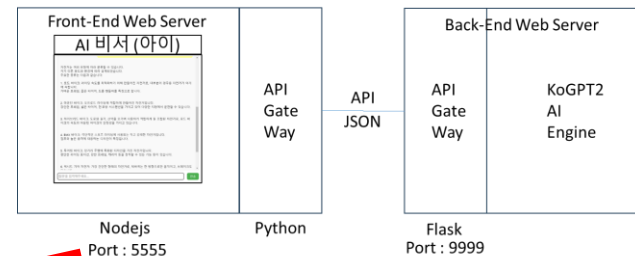
python main2.py



## 2. KoGPT 를 활용한 챗봇 환경 구축

⌚ 접속

<http://localhost:5555/>





# 참고 자료

- 자바와 파이썬으로 만드는 빅데이터시스템(제이펍, 황세규)
- 위키독스(<https://wikidocs.net/22654>)
- 네이버블로그(<https://blog.naver.com/classmethodkr/222822485338>)
- 데이터분석과 인공지능 활용 (NOSVOS, 데이터분석과인공지능활용편찬위원회 편)

## 참고 사이트

유튜버 : 빅공잼 : <https://www.youtube.com/watch?v=bnYxO2XRCQ0>

네이버 블로그 : 빅공잼

<https://biggongjam.notion.site/3-Hadoop-cd6944182da74edf8d2339b654e0bfb9>

<https://biggongjam.notion.site/4-Spark-2c341ddc8715411484cb2f0254b60126>

Q n A