WHOIS X PYTHON STUDY
BASIC CRAWLING

2018.11.21 | 최소혜 (purelledhand@gmail.com)

NAVER D2

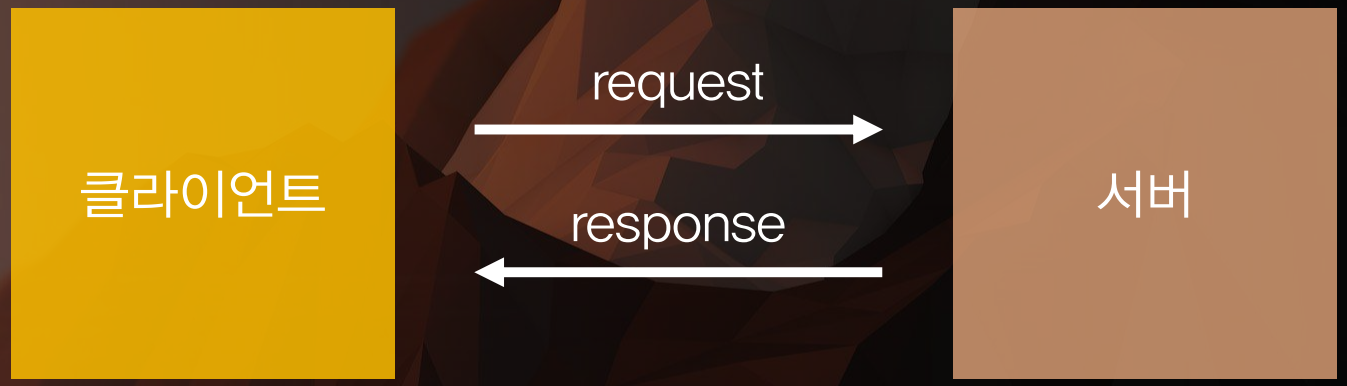
WHOIS X PYTHON STUDY
BASIC CRAWLING

2018.11.21 | 최소혜 (purelledhand@gmail.com)

- 웹 클라이언트
  교재 9.1
- 웹에서 다양한 정보 크롤링해오기
  교재 8.1, 8.2, 9.1, 9.3

NAVER D2

INTRO : Web Client

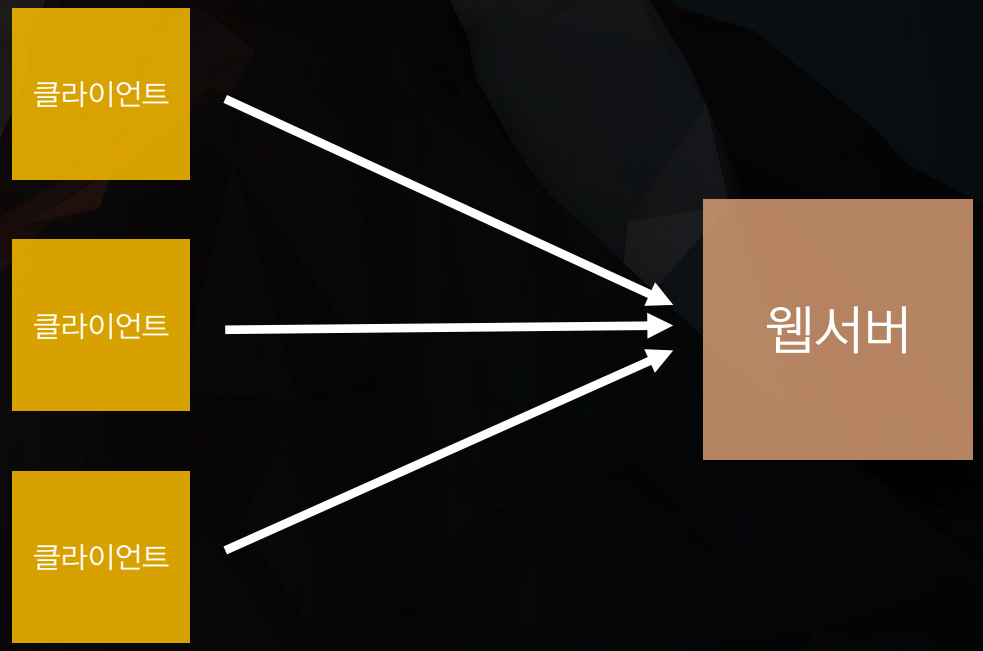


INTRO : BASEBAND 통신(시분할)을 하는 HTTP

Broadband 통신 : 주파수 분할 방식을 사용 (ex. TV)

Baseband 통신 : 시분할 방식을 사용 (ex. Ethernet)

- 시간을 아주 잘게 쪼개 클라이언트들과 번갈아 연결 하면서 각각의 클라이언트들이 마치 계속 연결 되어있는 것처럼 느끼도록 함



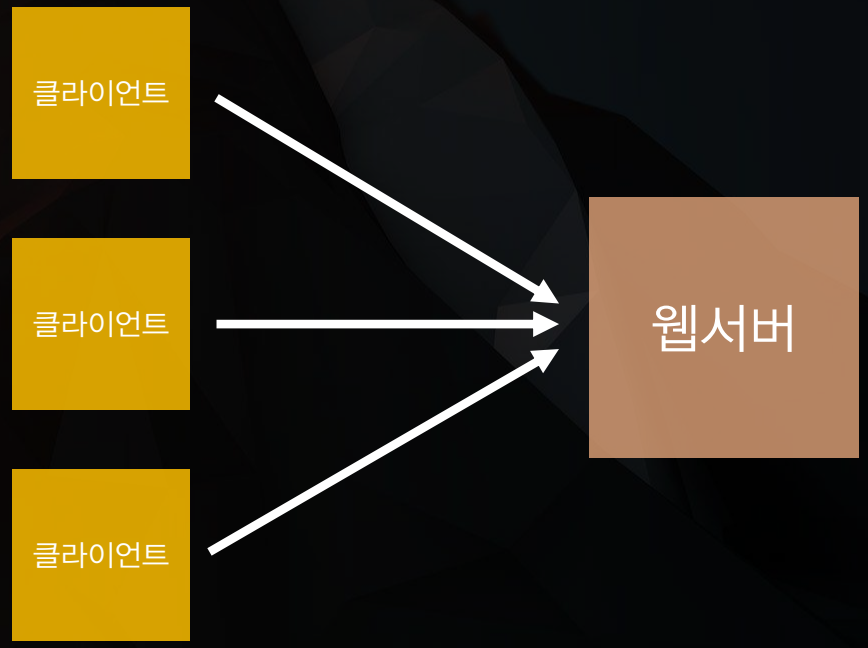
INTRO : BASEBAND 통신(시분할)을 하는 HTTP

Baseband 통신 : 시분할 방식을 사용

- 시간을 아주 잘게 쪼개 클라이언트들과
  번갈아 연결 하면서 각각의 클라이언트들이
  마치 계속 연결 되어있는 것처럼 느끼도록 함

- **서버는 지금 연결을 맺고있는게 누구인지는 알아야 한다.**
  로그인정보, 장바구니, 좋아요 표시 등등

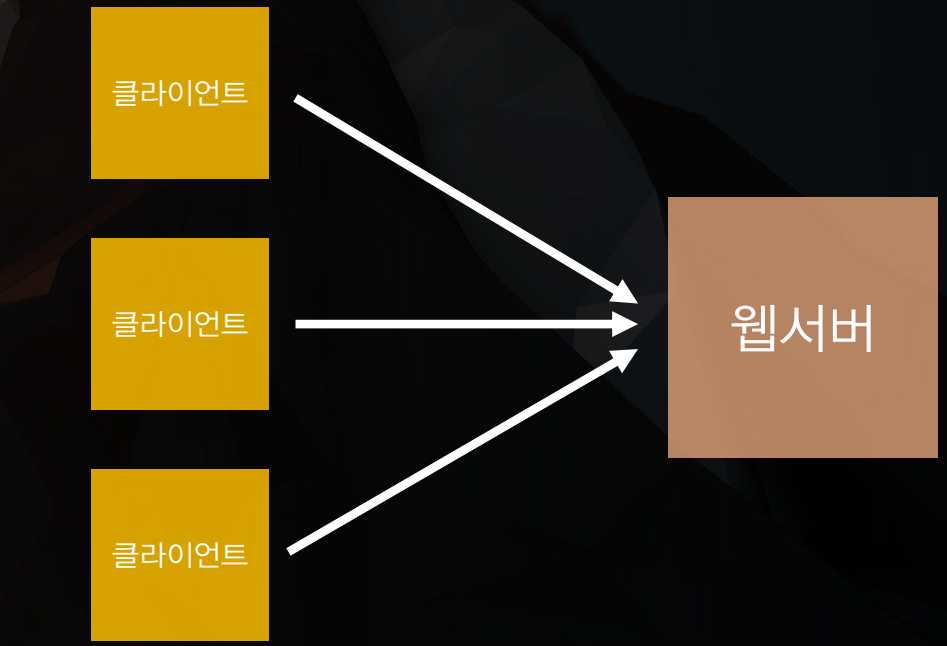
- 클라n과 연결을 끊고 클라k와 연결하고…
  수많은 연결을 맺고 끊으면서
  누가 누군지 어떻게 식별해야 할까에 대한 고민



INTRO : 쿠키와 세션 (Baseband 통신 방식의 관점으로 바라본.)

- **서버는 지금 연결을 맺고있는게 누구인지는 알아야 한다.**
  로그인정보, 장바구니, 좋아요 표시 등등

- 클라n과 연결을 끊고 클라k와 연결하고…
  수많은 연결을 맺고 끊으면서
  누가 누군지 어떻게 식별해야 할까에 대한 고민



쿠키와 세션이 필요한 이유

INTRO : 쿠키와 세션 (Baseband 통신 방식의 관점으로 바라본.)

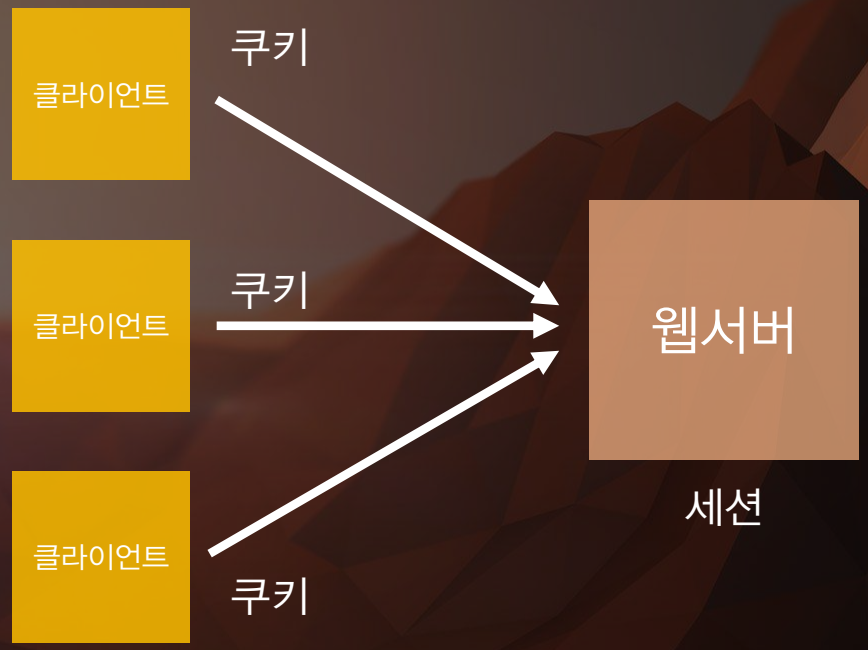
쿠키와 세션이 하는 일

- **서버는 지금 연결을 맺고있는게 누구인지는 알도록 한다.**
  로그인정보, 장바구니, 좋아요 표시 등등

- 클라n과 연결을 끊고 클라k와 연결하고…
  수많은 연결을 맺고 끊으면서
  서버가 누가 누군지 식별할 수 있도록 한다.

INTRO : 쿠키와 세션 (Baseband 통신 방식의 관점으로 바라본.)

쿠키와 세션이 하는 일



클라이언트가 http프로토콜을 이용해
웹서버에 웹페이지를 요청

⬇

각 클라이언트를 구분하기 위해
서버는 **클라이언트에게 쿠키를 발급**
동시에 같은 정보를 서버의 세션에 저장

⬇

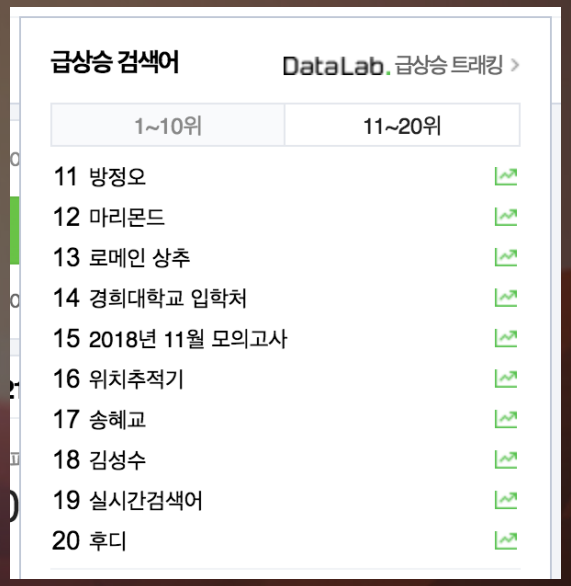
웹서버는 쿠키와 세션을 비교해가면서
클라이언트를 식별

웹에서 다양한 정보 크롤링해오기 (Super Duper Basic)

라이브러리를 설치해주세요 (requests, bs4)

```
pip install requests
pip install bs4
```

네이버 실시간 검색어 크롤링해오기

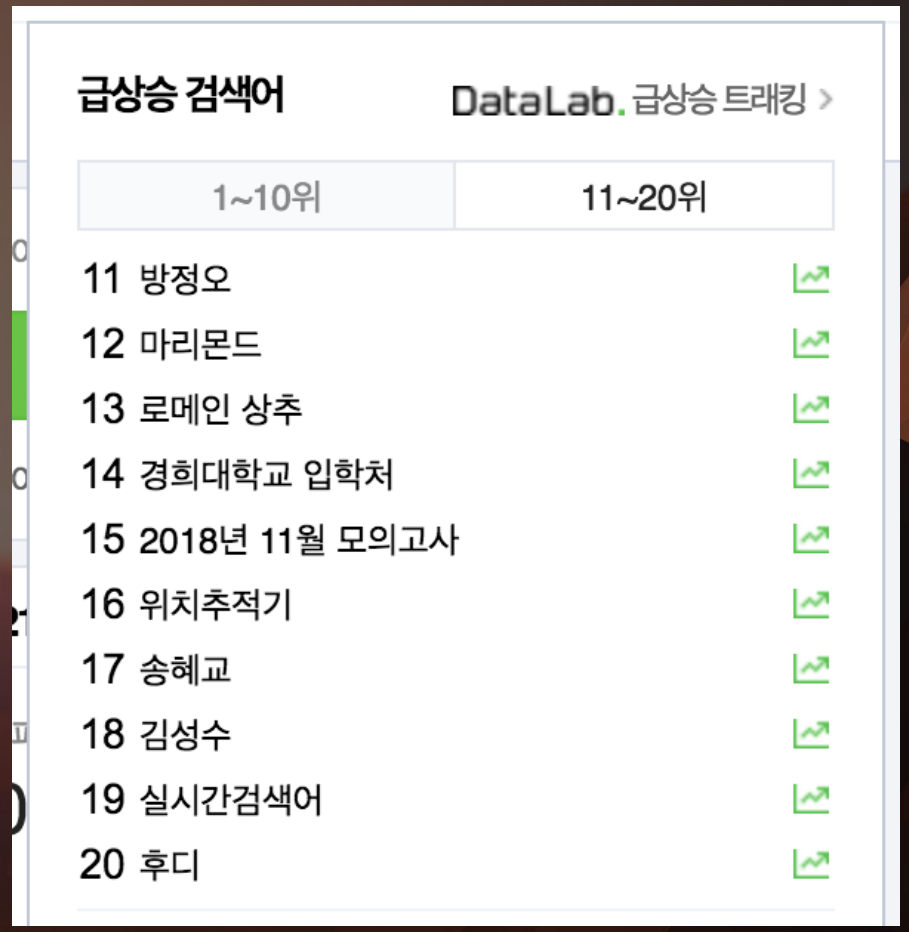


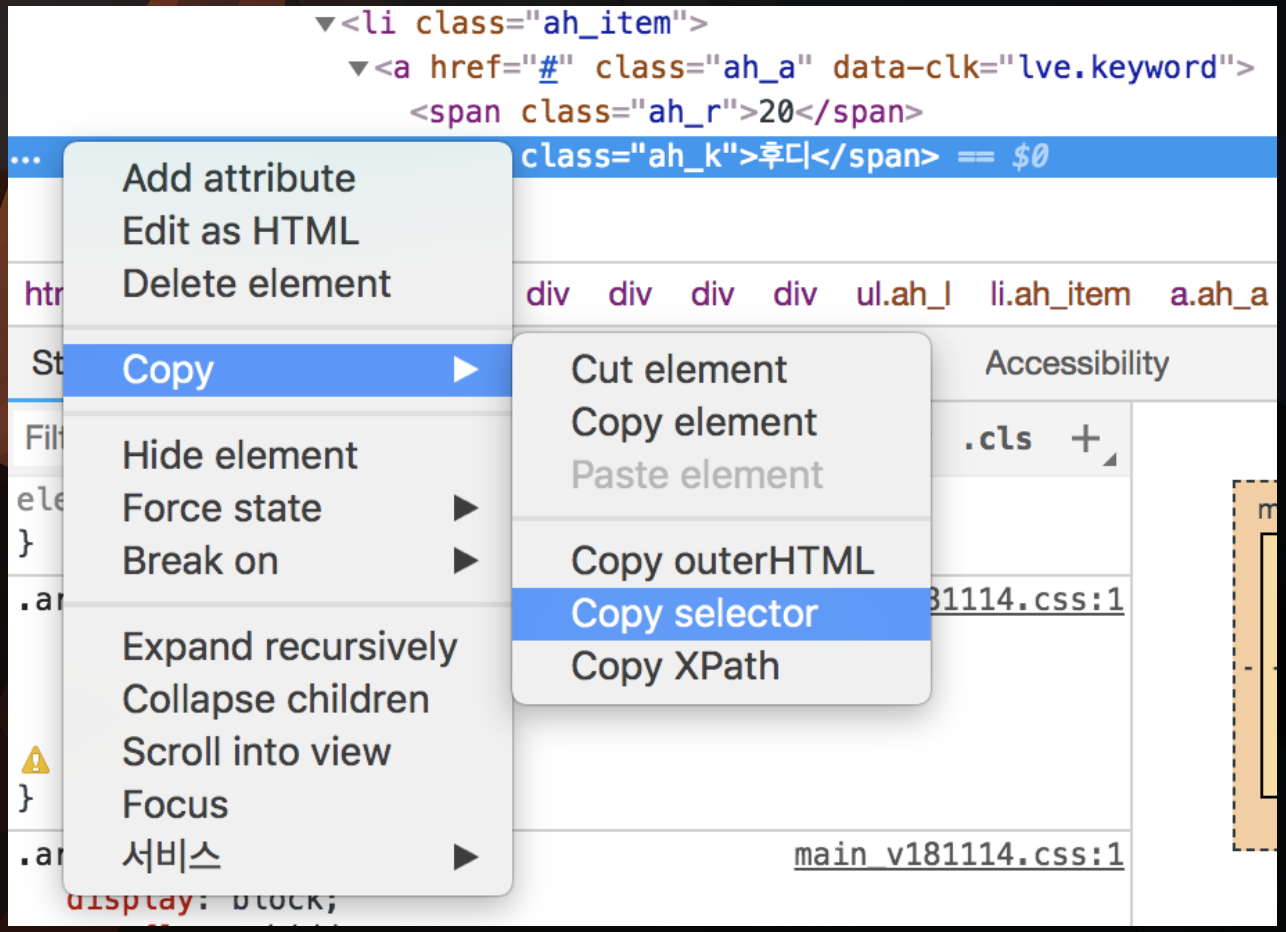

```
import requests
from bs4 import BeautifulSoup

req = requests.get('https://www.naver.com')
html = req.text
#print(html)

soup = BeautifulSoup(html, 'html.parser')
```

네이버 실시간 검색어 크롤링해오기





네이버 실시간 검색어 크롤링해오기

```
import requests
from bs4 import BeautifulSoup

req = requests.get('https://www.naver.com')
html = req.text
#print(html)

soup = BeautifulSoup(html, 'html.parser')

issues = soup.select(
    '#PM_ID_ct > div.header > div.section_navbar
    )
#print(issues)
```

네이버 실시간 검색어 크롤링해오기

**#PM_ID_ct > div.header > div.section_navbar >
div.area_hotkeyword.PM_CL_realtimeKeyword_base >
div.ah_roll.PM_CL_realtimeKeyword_rolling_base > div > ul > li:nth-child(20) > a > span.ah_k**

li로 변경해서 ul내의 모든 li태그를 가져오자

```
<ul>
    <li></li> # li:nth-child (1)
    <li></li> # li:nth-child (2)
    <li></li> # li:nth-child (3)
    <li></li> # li:nth-child (4)
    <li></li> # li:nth-child (5)
</ul>
```

li

네이버 실시간 검색어 크롤링해오기

```
<a href="#" class="ah_a" data-clk="lve.keyword">
    <span class="ah_r">20</span>
    <span class="ah_k">후디</span>
</a>
```

20 후디

#PM_ID_ct > div.header > div.section_navbar > div.area_hotkeyword.PM_CL_realtimeKeyword_base > div.ah_roll.PM_CL_realtimeKeyword_rolling_base > div > ul > li > a

네이버 실시간 검색어 크롤링해오기

```
import requests
from bs4 import BeautifulSoup

req = requests.get('https://www.naver.com')
html = req.text
#print(html)

soup = BeautifulSoup(html, 'html.parser')

issues = soup.select(
    '#PM_ID_ct > div.header > div.section_navbar > div.area_hotkeyword.PM_CL_realtimeKeyword_base > div.ah_
    )
#print(issues)
```

주석 풀고 확인해보기

네이버 실시간 검색어 크롤링해오기

```
<a href="#" class="ah_a" data-clk="lve.keyword">
    <span class="ah_r">20</span>
    <span class="ah_k">후디</span>
</a>
```

Select_one() method
Select_one('span[class="ah_r"]').text

```
purelledhand% python3 crawl.py
[1] 골프장 동영상
[2] 레스모아
[3] 애슐리 1+1
[4] 경희대학교 입학처
[5] 로메인 상추
[6] 조선일보 손녀
[7] 윤진영
[8] 2018년 11월 모의고사
[9] 첫눈
[10] 연성대학교
[11] 유승준
```

```
for issue in issues:
  print("["+issue.select_one(?).text +"]  "+issue.select_one(?).text)
```

네이버 수요 웹툰 목록 크롤링해오기

```
purelledhand% python3 webtoon_crawl.py
WEBTOON :유미의 세포들
LINK :/webtoon/list.nhn?titleId=651673&weekday=wed

WEBTOON :복학왕
LINK :/webtoon/list.nhn?titleId=626907&weekday=wed

WEBTOON :고수
LINK :/webtoon/list.nhn?titleId=662774&weekday=wed

WEBTOON :연놈
LINK :/webtoon/list.nhn?titleId=667573&weekday=wed
```

HINT
Webtoon title : a.text
Webtoon link : a.get('href')

네이버 수요 웹툰 목록 크롤링해오기

```
import requests
from bs4 import BeautifulSoup
import json
import csv
import os
req = requests.get('https://comic.naver.com/webtoon/weekdayList.nhn?week=wed')
html = req.text
soup = BeautifulSoup(html, 'html.parser')

webtoons = soup.select(
    '#content > div.list_area.daily_img > ul > li > dl > dt > a'
    )
```

네이버 수요 웹툰 목록 json 파일로 추출하기

```
data = {}

CURRENT_DIR = os.path.dirname(os.path.abspath(__file__))

for title in webtoons:
  data[title.text] = title.get('href')

with open(os.path.join(CURRENT_DIR, 'result.json'), 'w+') as json_file:
  json.dump(data, json_file)

json_file.close()
```

WHOIS X PYTHON STUDY
BASIC CRAWLING

네이버 수요 웹툰 목록 CSV로 추출하기

	A	B
1	TITLE	LINK
2	유미의 세포들	/webtoon/list.nhn?titleId=651673&weekday=wed
3	복학왕	/webtoon/list.nhn?titleId=626907&weekday=wed
4	고수	/webtoon/list.nhn?titleId=662774&weekday=wed
5	연놈	/webtoon/list.nhn?titleId=667573&weekday=wed
6	헬퍼 2 : ...	/webtoon/list.nhn?titleId=670143&weekday=wed
7	세상은 돈과 ...	/webtoon/list.nhn?titleId=710747&weekday=wed
8	귀곡의 문	/webtoon/list.nhn?titleId=718020&weekday=wed
9	이츠마인	/webtoon/list.nhn?titleId=710760&weekday=wed
10	여심강타(fe...	/webtoon/list.nhn?titleId=717480&weekday=wed
11	신석기녀	/webtoon/list.nhn?titleId=703308&weekday=wed
12	신암행어사	/webtoon/list.nhn?titleId=703307&weekday=wed
13	조선왕조실톡	/webtoon/list.nhn?titleId=642598&weekday=wed
14	레사 시즌2~3	/webtoon/list.nhn?titleId=603159&weekday=wed
15	격기3반	/webtoon/list.nhn?titleId=701535&weekday=wed
16	가우스전자 시...	/webtoon/list.nhn?titleId=675554&weekday=wed
17	일렉시드	/webtoon/list.nhn?titleId=717481&weekday=wed
18	언덕 위의 제...	/webtoon/list.nhn?titleId=671421&weekday=wed
19	2018 루키...	/webtoon/list.nhn?titleId=717031&weekday=wed
20	12차원 소년들	/webtoon/list.nhn?titleId=717059&weekday=wed
21	요리GO	/webtoon/list.nhn?titleId=703849&weekday=wed
22	미시령	/webtoon/list.nhn?titleId=697533&weekday=wed
23	펀브로커	/webtoon/list.nhn?titleId=710765&weekday=wed
24	성공한 덕후	/webtoon/list.nhn?titleId=703628&weekday=wed
25	그 판타지 세...	/webtoon/list.nhn?titleId=316909&weekday=wed
26	고교생을 환불...	/webtoon/list.nhn?titleId=708453&weekday=wed

NAVER D2

네이버 수요 웹툰 목록 CSV로 추출하기

```
jsonfile = open('result.json','r')
json_data = json.load(jsonfile)
#print(json_data)

csvfile = open(file='wed_webtoon.csv',mode='w',newline='', encoding='euc_kr')
csvwriter = csv.writer(csvfile)

count = 0;

for key in json_data:
        if count == 0:
                csvwriter.writerow(["TITLE", "LINK"])
                count += 1
        #print(key+json_data[key])
        csvwriter.writerow([key, json_data[key]])

jsonfile.close()
```