투빅스 11기 정규세션

ToBig's 10기 임진혁

크롤링!(web scraping)

Contents

Unit 01 | 인트로!

- **웹 스크레이핑이란?**

➡ Web scraping**이란, 웹사이트로부터 데이터를 수집하는 모든 과정을 말한다.**

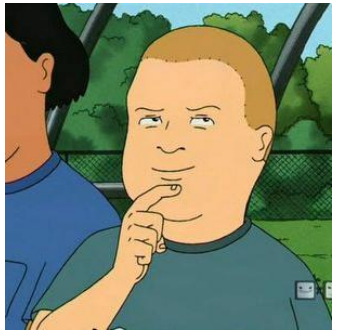
- **크롤링이란?**

➡ Web crawling**은 보통** 조직적, 자동화된 방법으로 웹을 탐색해서 복사본을 생성하는 행위를 말한다.
그 어원에서 알 수 있듯이 웹페이지의 연결된 링크들을 중복없이 효율적으로 탐색하는 역할을 가진 수집봇-"크롤러"(오토봇)에서 유래한 말이다.

웹 스크레이핑이 크롤링이란 개념을 포함한다고 할 수 있지만,
혼용되어 쓰이면서 뉘앙스에 차이가 있을 뿐
거의 같은 의미로 쓰이게 됨.

Unit 01 | 인트로!

"우린 이미 스크레이핑을 무의식적으로 하고 있었다??!"



-Datamarket 홈페이지에서 강의자료를 다운받는 행위도
웹사이트로부터 데이터를 수집한 것이므로 웹스크레이핑!!

하지만 웹페이지가 데이터를 제공하는 방식은 "파일다운로드" 방식 뿐만 아니라
매우 다양함. 모든 방식이 "파일" 방식이면 그냥 다운로드하면 크롤링 수업 끝!

그게 아니니까 문제!

Unit 01 | 인트로!

웹페이지들은 매우 다양한 방식으로 데이터를 저장!

보통 웹사이트들의 데이터들은 HTML,XML,JSON,API 등의 문서 형태로 저장되며

각각의 데이터들은 HTML 파서 , CSS 선택자, 정규표현식등으로 추출 가능~

즉,데이터들을 얻을려면 각 저장방식에 맞는 추출 기술을 적용해야함!

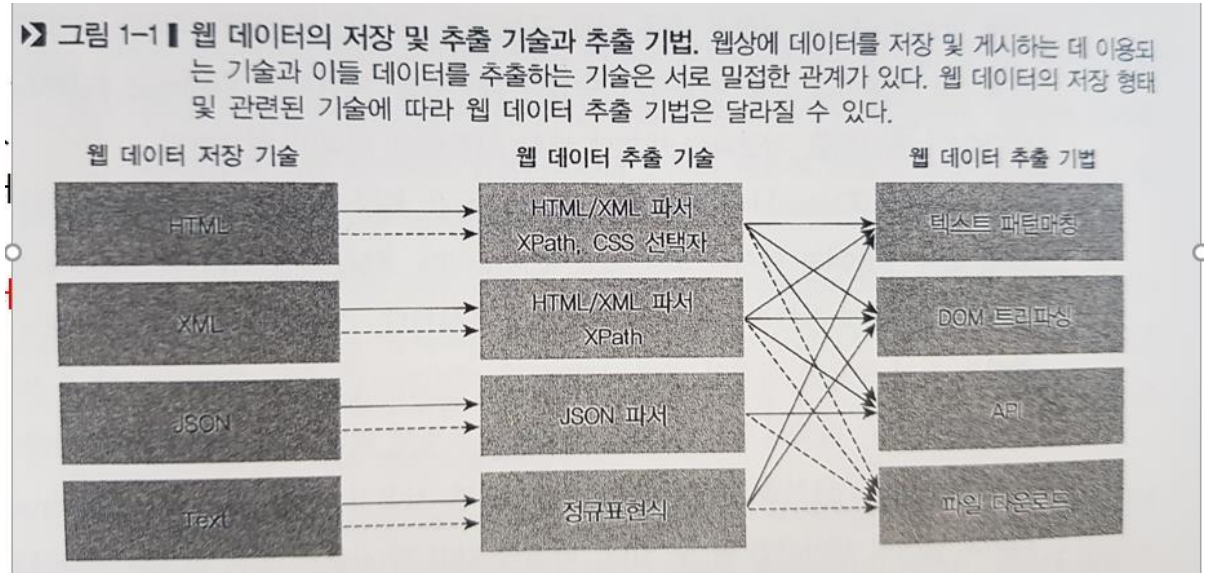


그림 1-1 | 웹 데이터의 저장 및 추출 기술과 추출 기법. 웹상에 데이터를 저장 및 게시하는 데 이용되는 기술과 이들 데이터를 추출하는 기술은 서로 밀접한 관계가 있다. 웹 데이터의 저장 형태 및 관련된 기술에 따라 웹 데이터 추출 기법은 달라질 수 있다.

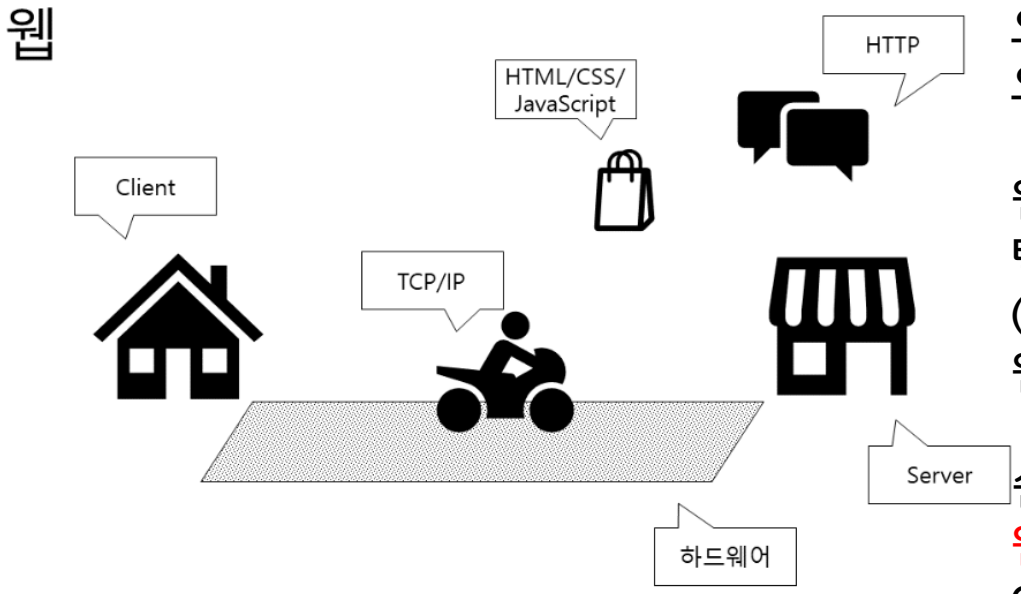
Unit 01 | 인트로!

각 저장 기술에 따른 추출 기술은 이미 정리되어 있고 , 구글링으로도 쉽게 익힐 수 있음!

결국 중요한 것 → 내가 추출하고자 하는 데이터가 어떤 방식으로 저장되어 있는지 파악하여
적합한 추출 기술을 선택해 추출하는 것!
따라서, 각 저장 기술과 추출 기술에 대한 이해가 필요! == 활용도 증가!!

오늘 수업은 다양한 저장 기술과 추출 기술에 대한 기본적인 이해(핥기)가 되겠습니다!!
-정적 웹페이지 크롤링 (HTML,XPATH,CSS 등)
-동적 웹페이지 크롤링 (셀레늄)
-SNS 크롤링 (트위터 등)
-공공데이터 API
→ 이 정도 기본으로도 공모전이나 프로젝트에 바로 활용 충분!!

[웹]에 대한 이해.



인터넷은
요청을 보내는 클라이언트 (CLIENT),
요청에 응답을 하는 서버(SERVER)로 구성!!

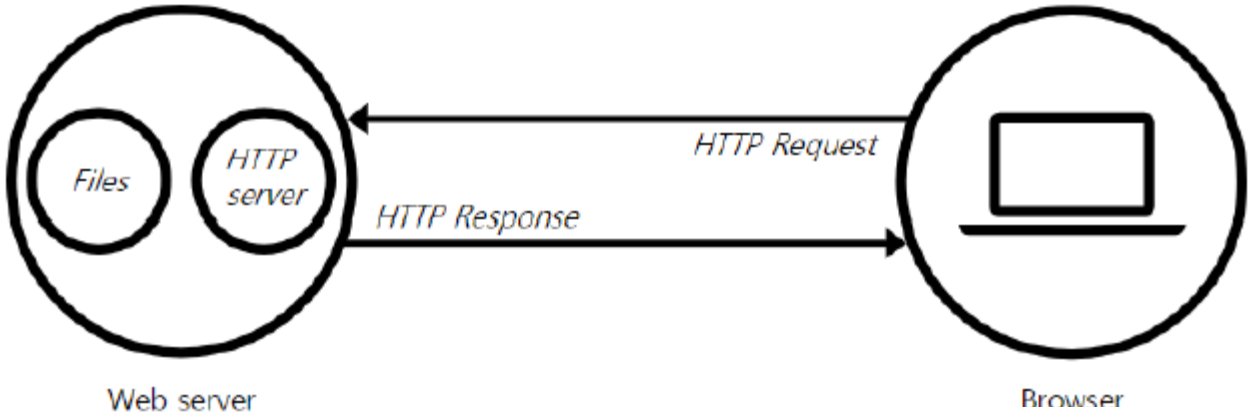
웹(월드 와이드 웹)은 인터넷의 한 종류이며
텍스트뿐만 아니라 음악,링크,사진까지 담고있는 하이퍼텍스트
(Hypertext)라고 하는
웹문서를 중심으로 합니다!!

쉽게 말해서, (HyperText Markup Language)
웹문서를 표현하는 양식 = HTML = 파일의 형식
CSS: HTML의 스타일 담당 ,JavaScript: HTML의 동적기능담당

파일을 전송하는 방법/방식:
HTTP(HyperText Transfer Protocol)

[웹]에 대한 이해.

서버: 외부에서 요청하면 규칙대로 정보를 제공하는 것
브라우저: 서버가 주는 것들을 사용자에게 보여주는 것
웹서버: text(html, css, js, etc), image. 브라우저가 약속된 모양으로 우리에게 보여줌.



HTTP는 클라이언트가 서버에 요청을 보내면,
서버가 응답을 하는 방식으로 통신이 이뤄집니다.

요청의 종류에 따라 Get,Post 방식이 있으며
대부분의 웹페이지들은 Get 방식이지만
정부기관들은 Post 방식을 쓰는 경향이 있다.

서버는 응답을 할 때 상태 코드를 전송한다. 상태코드는 세 자리 숫자로 되어 있다. 첫 자리 숫자에 따른 의미는 다음과 같다.

- 2XX: 성공. 보통 200을 사용한다.
- 3XX: 리다이렉션. 요청한 URL이 다른 URL로 이동했다. 301 (영구이동)과 302 (임시이동)이 자주 사용된다.
- 4XX: 요청 오류. 잘못된 요청을 보낸 경우이다. 대표적으로 잘못된 URL을 입력한 경우 발생하는 404 (찾을 수 없음)가 있다
- 5XX: 서버 오류. 버그나 장애 등 서버 측의 오류이다. 500(내부 서버 오류), 502 (불량 게이트웨이) 등이 있다.

[HTML]에 대한 이해 +CSS +XPATH

기본문법

-HTML은 웹 페이지를 만드는 데 널리 쓰이는 마크업 언어.
-이 언어로 사람과 컴퓨터 둘다 웹 페이지를 어떻게 보여줄 지 이해할 수 있다.
-HTML은 트리 형태의 구조를 가진다.

-가장 크게 모든 문서는 <html>로 시작하여 </html>로 끝난다.
-그 아래에는 <head> 와 <body> 가 있다.

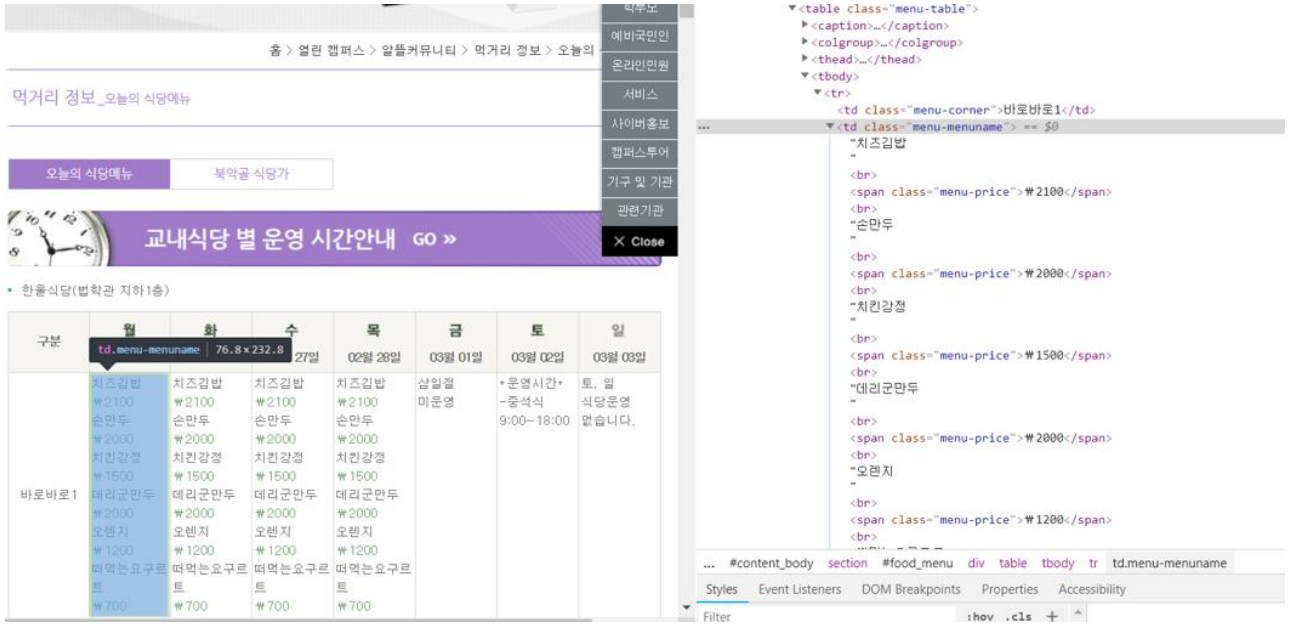
- HTML에서 엘리먼트(element)는 페이지의 구성 요소이다.
- 엘리먼트에는 제목, 문단, 표, 목록 등이 있다

(HTML 페이지 예시)

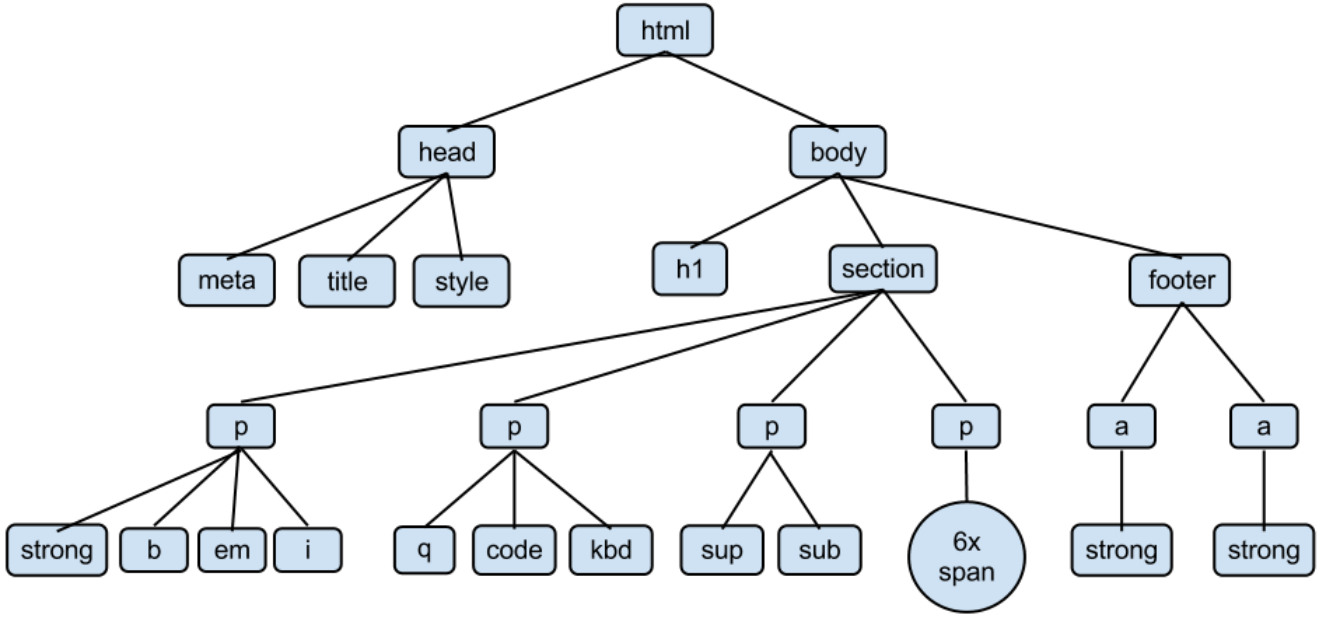
```
<!DOCTYPE html>
<html>
<head>
<title>페이지 제목</title>
</head>
<body>

<h1>제목1</h1>
<p>문단</p>

</body>
</html>
```

제목1

문단



[HTML]에 대한 이해 +CSS +XPATH　　트리 구조? +태그(노드)



모든 노드(Node)는 태그(tag)로 감싸여 있음.
태그는 엘리먼트의 종류를 나타냄
태그는 < > 안에 태그 이름을 쓴다.
내용의 시작과 끝에 태그를 쓰고 종료태그일 경우
/(슬래시)를 붙인다

```
<tagname>내용</tagname>
```

태그에는 엘리먼트 태그
이외에도 서식 태그가 있다.

모든 엘리먼트는 속성(attribute)를 가질 수 있다.
속성은 시작 태그와 같이 명시되고
속성 ="값" 형태로 표현한다.

[HTML]에 대한 이해 +CSS +XPATH

트리 구조? +태그(노드)

HTML – Hypertext Markup Language
자료를 문서형태로 만들기 위한 명령어

CSS – Cascading Style Sheet
HTML 요소를 시각적으로 꾸며주는 역할

```
<!DOCTYPE html>
<html> tag
<head>
  <link rel="stylesheet" href="styles.css">
</head>
<body>

<h1>This is a heading</h1>
<p>This is a paragraph.</p>

</body>
</html>
```

```
body {
    background-color: powderblue;
}
h1 {
    color: blue;
}
p {
    color: red;
}
```

```
p { color : green; }
```

이것을 풀이해 보면,

P의 색상은 초록색입니다.

선택자 ↓ 속성명 속성값 속성명 속성값

```
a { background-color: yellow; font-size: 16px; }
```

↑ 선언 시작 ↑ 선언 구분자 ↑ 선언 끝

CSS 선택자를 이용하여 특정 속성(어트리뷰트)를 가진 노드를 특정지어 추출할 수 있다!

[HTML]에 대한 이해 +CSS +XPATH

자주 사용되는 태그

제목 (heading) – `<h1>` , `<h2>` , `<h3>` , `<h4>` , `<h5>` , `<h6>`

문단 – `<p>`

링크 – `<a>`

링크들은 `<a>` 태그로 나타낸다.

링크 주소는 `href` 라는 어트리뷰트(attribute)에 지정한다.

구역 – `<div>`

```
<!DOCTYPE html>
<html>
<body>

<h1>This is heading 1</h1>
<h2>This is heading 2</h2>
<h3>This is heading 3</h3>
<h4>This is heading 4</h4>
<h5>This is heading 5</h5>
<h6>This is heading 6</h6>

</body>
</html>
```

This is heading 1

This is heading 2

This is heading 3

This is heading 4

This is heading 5

This is heading 6

```
<!DOCTYPE html>
<html>
<body>

<a href="http://www.google.com">구글 링크</a>

</body>
</html>
```

구글 링크

[HTML]에 대한 이해 +CSS +XPATH

특정 태그를 사용하여 텍스트 포맷을 변경할 수 있다.

`<b>` 는 글씨를 굵게 표기해준다.

`<i>` 는 글씨에 기울임을 적용한다.

`<strong>` 은 중요한 텍스트로 인식시켜서 표기한다.

```
<!DOCTYPE html>
<html>
<body>

<p><b>This text is bold</b></p>
<p><i>This text is italic</i></p>
<p><strong>This text is important</strong></p>

</body>
</html>
```

This text is bold

This text is italic

This text is important

자주 사용되는 태그

자주 사용되는 어트리뷰트

`alt` - 이미지가 출력되지 않을 때 alt 에 있는 텍스트 정보가 출력된다.

`src` - 이미지의 url 주소나 파일명으로 지정한다.

`href` - URL 링크 주소로 지정한다.

`id` - 엘리먼트에 대한 특별한 아이디를 부여한다. 원칙적으로 하나의 아이디는 같은 문서에 중복해서 나올 수 없다. 그렇지만 어기는 경우도 있다.

`class` - 같은 유형의 엘리먼트를 나타낸다. 보통 서식을 지정하기 위해 사용한다. 엘리먼트는 한 개 이상의 클래스를 가질 수 있다.

```
<a class='class_1'>1번 값</a>
<li>
    <a class='class_1'>2번 값</a>
```

[HTML]에 대한 이해 +CSS +XPATH

HTML에 대한 기본적인 파악은 완료!
이제 충분히 크롤링을 할 수 있음!!

HTML을 파싱(parsing) 해줄 파이썬 모듈에 BeautifulSoup과 lxml이 있다.
(파싱- 사람이 이해할 수 있게 HTTP의 응답을 해석하는 것)

BeautifulSoup
단점: 기능이 적고 느림
장점: 약간 쉽고 깨진 HTML을 약간 더 잘 인식
lxml
단점: 약간 어렵고 깨진 HTML을 조금 더 못 인식
장점: 기능이 많고 빠름

HTML에서 특정 부분을 추출 하는 방법에 CSS 선택자와 XPath 방식이 있다.

CSS 선택자
HTML 문서의 서식을 지정하기 위한 용도
짧고 간단
대부분의 경우에는 충분함
XPath
복잡한 조건으로 노드를 선택할 때 사용
lxml의 기본 사용법

[HTML]에 대한 이해 +CSS +XPATH

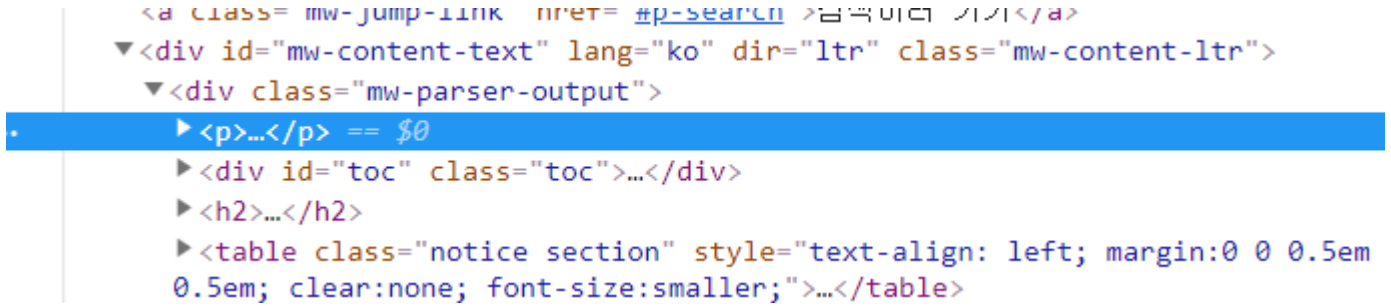
위키에서 웹검색 : 인공지능
https://ko.wikipedia.org/wiki/%EC%9D%B8%EA%B3%B5%EC%A7%80%EB%8A%A5

F12를 누르면 이 웹페이지의 데이터를 HTML 형식으로 확인가능!!

Ctrl+Shift+c 를 누르면 내가 추출하고 싶은 부분을 특정가능!

이 웹페이지 주소를 복사해주세요!! -> 실습1

CSS선택자에서 class는 .으로 id는 #으로 표시한다.

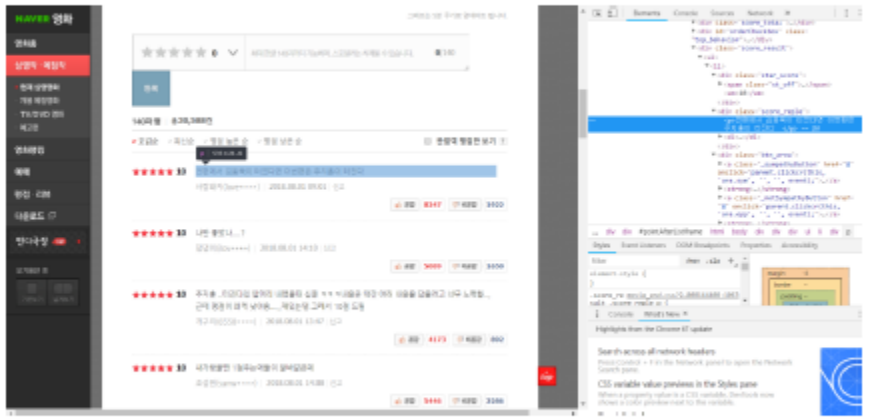


[보다 간편한 조작을 위한 beautiful soup를 이용한 조작법]

크롤링의 3단계 요청(Requests) 추출(Parsing) 저장(Save)

```
import requests
```

requests: Python에서 HTTP 요청을 보내는 모듈



Html 소스는 우리가 F12버튼을 눌렀을 때 나오는
개발자 도구 속 html 소스

[보다 간편한 조작을 위한 beautiful soup를 이용한 조작법]

크롤링의 3단계 요청(Requests) 추출(Parsing) 저장(Save)

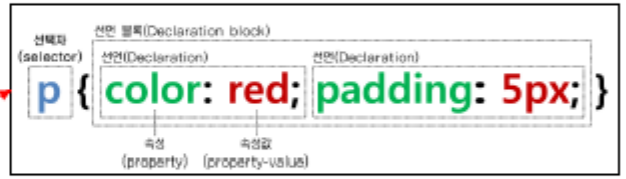
```
from bs4 import BeautifulSoup
```

BeautifulSoup: Python에서 HTML과 XML 파일로부터 데이터를 뽑아내기 위한 라이브러리

파싱(Parsing): 데이터를 조립해 원하는 데이터를 뽑는 것.
주로 HTML 소스 구조를 보고 CSS 선택자를 통해 값을 뽑아준다.

```
[<tr>
 <th>번호</th>
 <th colspan="2">평점</th>
 <th>140자평</th>
 <th class="al"><span class="th_m1">글쓴이 · 날짜</span></th>
 </tr>, <tr>
 <td class="ac num">14478590</td>
 <td><div class="fr point_type_n">
 <td class="point">8</td>
 <td class="title">
 <a class="movie" href="?st=mcode&amp;sword=163687&amp;target=after">몽식</a>
 <br/>나는 정말 괜찮고 좋았다 하지만 재미로 보기엔 너무 지루하다
```

```
# 2. 제목확인
for tit in reviews[1:]:
    print(tit.find('a',{"class":"movie"}).find(text=True))
```



선택자 <a>는 tag
속성은 class
속성값은 movie

⇒ 여러 개의 <a> 중에서
⇒ 속성 class가 movie인 아이를 선언

Unit 02 | 웹데이터 저장 기술에 대한 이해+정규표현식

크롤링의 3단계 요청(Requests) 추출(Parsing) 저장(Save)

```
from bs4 import BeautifulSoup
```

BeautifulSoup: Python에서 HTML과 XML 파일로부터 데이터를 뽑아내기 위한 라이브러리

BeautifulSoup 내의 Parsing Method

태그를 검색하려면?

코드	의미
soup.find_all("a")	모든 a 태그 검색
soup.title.find_all(string=True)	string 이 있는 title 태그 모두 검색
soup.find_all("a", limit=2)	a 태그를 두개만 가져옴
soup.find_all("p", "title")	p 태그와 속성 값이 title 이 있는거
soup.find_all(["a", "b"])	a태그와 b태그 찾기
soup.body.b soup.a	간단한 검색 # body 태그 아래의 첫번째 b 태그 # 첫번째 a 태그

find( )

코드	의미
soup.find("div", attrs={"data-value": True})	data-로 시작하는 속성 find
soup.find("div").name	태그명 얻기
soup.find("div")['class'] soup.find("div").get('class')	속성 얻기 # 만약 속성 값이 없다면 에러 # 속성 값이 없다면 None 반환

[이걸 다 어떻게 외우나요??]

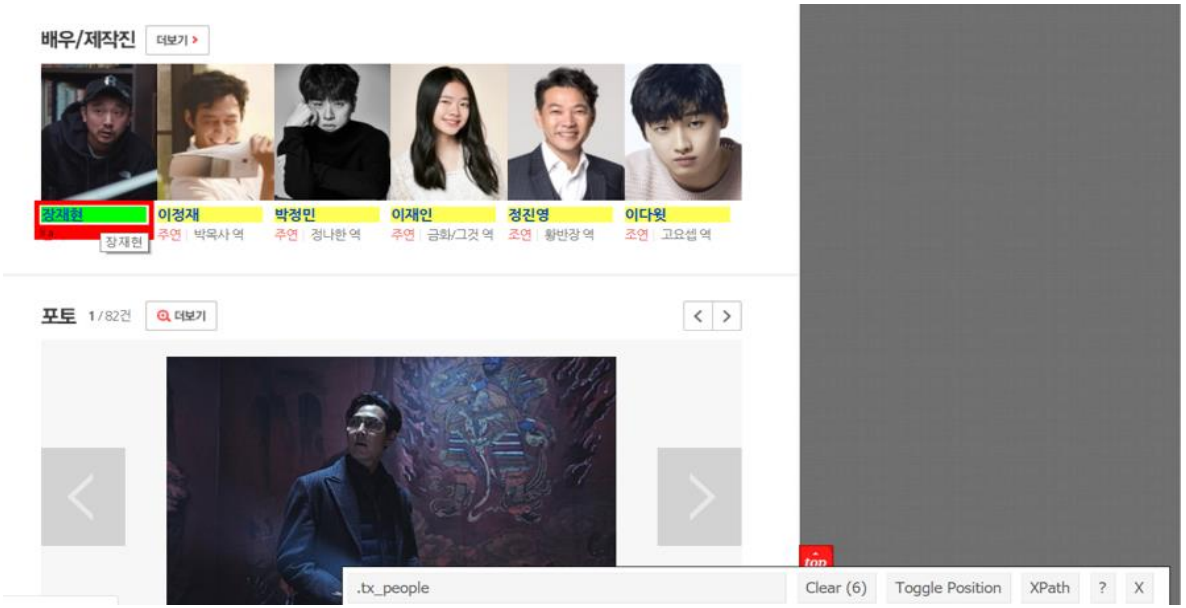
다 못 외웁니다.
많이 쓰이는 문법을 기억하거나 필요할 때 마다 찾아봅니다.

하지만..!!

[사기템1]

https://chrome.google.com/webstore/detail/selectorgadget/mhjhnkcfbdhnjickkkdbjoemdmbfginb?hl=ko

구글의 확장 프로그램!! Xpath, css선택자 모두 지원!! 그냥 자동으로다가 알려줍니다!!



[그러면 정규표현식은 무엇인가요?]

파이썬의 정규표현식 모듈 're'


```
In [5]: import re
        p = re.compile('010-?[0-9]{4}-?[0-9]{4}')
        p.findall('내 연락처는 010-1234-5678입니다')
Out[5]: ['010-1234-1234']

In [4]: p.findall('내 연락처는 01012345678입니다')
Out[4]: ['01012341234']
```

파이썬의 정규표현식 모듈 're'의 Method

Method	목적
match()	문자열의 처음부터 정규식과 매치되는지 조사한다.
search()	문자열 전체를 검색하여 정규식과 매치되는지 조사한다.
findall()	정규식과 매치되는 모든 문자열(substring)을 리스트로 리턴한다
finditer()	정규식과 매치되는 모든 문자열(substring)을 iterator 객체로 리턴한다

Unit 02 | 웹데이터 저장 기술에 대한 이해+정규표현식

[정규표현식은 추출한 난잡한 데이터들 중에서 필요한 내용만을 추출하기 위해 사용!!]

분류	문법	설명
특수 문자	₩s, ₩S	공백, ₩s를 제외한 문자
	₩d, ₩D	숫자, ₩d를 제외한 문자
	₩w, ₩W	영숫자와 밑줄, ₩w를 제외한 문자
	₩t, ₩v, ₩V	탭 문자, 수직 탭 문자, ₩v 제외한 문자
	₩r, ₩n	CarrageReturn, 줄 바꾸기
	₩a, [₩b]	Alert, Backspace
	₩c~, ₩f	제어 문자, 페이지 바꾸기
	₩h, ₩H	수직 whitespace, ₩H 제외한 문자
	₩0~, ₩d~	8진수 문자, 10진수 문자
	₩x~, ₩u~	16진수 문자, Unicode 문자
메타 문자	.	임의의 문자
	|	왼쪽 문자 또는 오른쪽 문자 매핑
	[문자열]	문자열의 문자중 임의의 문자에 매핑
	[^문자열]	문자열의 문자를 제외한 문자에 매핑
	[a-z]	a 문자부터 z 문자까지 매핑
수량자	?	0개 또는 1개 일치
	*, *?	0개 이상 일치, 최소 크기의 문자열
	+, +?	1개 이상 일치, 최소 크기의 문자열
	{m}, {m}?	m개 일치, 최소 크기의 문자열
	{m, }	m개 이상 일치
	{m, n}	m개 이상이고 n이 이하 일치

분류	문법	설명
위치	^, ₩A	문자열의 시작
	$, ₩Z	문자열의 끝
	₩<	단어의 시작
	₩>	단어의 끝
	₩b, ₩B	단어의 경계, ₩b를 제외한 문자
추출/탐색	(pattern)	pattern에 해당하는 문자열 그룹 추출
	(?:pattern)	pattern을 체크만 하고 저장하지 않음
	(?i:pattern)	(?:pattern)에서 대소문자 구분 않음
	(?=~), (?!~)	Positive/Negative Lookaheads
	₩n, $n	그룹 추출로 추출한 문자열 그룹 n은 1, 2, 3, … (₩n이 POSIX 방식)
	?, ?<=	전방 탐색, 후방 탐색
	?!, ?<!	부정형 전방탐색, 부정형 후방탐색
	?(조건)true	조건 지정
	?(조건)true|false	else 표현식 조건 지정
변환	₩l, ₩u	다음 문자를 소문자로, 대문자로
	₩L~₩E	~에 해당하는 문자를 소문자로
	₩U~₩E	~에 해당하는 문자를 대문자로
기타	(?i), (?m)	대소문자 무시, 다중행 모드
	(?x)	공백 무시 모드, #으로 주석 표시
	(?P<name>pattern)	역참조시 이름을 지정 in Python ₩1 = ₩g<1> = ₩g<name> m.group('name')

https://regexr.com/에서 확인가능합니다!!

예를 들어,
추출한 데이터가
<영화이름 평점 감독>의 형태인데
평점 정보만 갖고 싶다면

정규표현식으로 숫자만 추출하게
활용합니다!!

Unit 03 | 크롤링이 안되는 경우

1. 정적인 웹페이지가 아니거나
자바스크립트로 작성되었을 경우→ 셀레늄

2.웹사이트가 접근을 차단할 경우 → 유저 에이전트 변경

3. 특정 인가자만 접근을 허용할 경우 → 리퍼러

4.POST 방식으로 웹페이지가 작성되었을 경우 -> 리퍼러

Unit 03 | 크롤링이 안되는 경우

[셀레늄]

- 셀레늄 다운로드 페이지: https://www.seleniumhq.org/download
- 크롬 웹드라이버: https://sites.google.com/a/chromium.org/chromedriver/downloads

드라이버를 설치하여 마치 실제 웹 브라우저로 탐색하는 것과 똑같이 웹사이트들을 돌아다니며 크롤링을 할 수 있다. → 대신 속도가 느리다.

웹에 있는 버튼이 작동을 하는지 확인하거나, 버튼을 100번씩 반복해서 누르는 테스트 등을 일일이 사람이 클릭하여 테스트하면 번거롭기 때문에, 웹 브라우저를 제어하여 테스트하도록 도와준다.

크롤링이 어려운 사이트의 경우, 셀레늄을 사용하면 실제 웹 브라우저를 제어하여 사람처럼 접속할 수 있기 때문에 쉽게 크롤링을 할 수 있다.

실습 2!!

Unit 03 | 크롤링이 안되는 경우

[셀레늄]

Element를 찾는 함수와 Selenium 요소 조작은 필요할 때마다 찾아쓰는 것이 좋습니다!

Element를 찾는 함수

find_element_by_id(id)	Id 속성으로 요소 하나 추출	
find_element_by_name(name)	Name속성으로 요소 하나 추출	
find_element_by_class_name(name)	클래스 이름이 name에 해당하는 요소 하나 추출	find_elements_by_class_name
find_element_by_partial_link(text)	링크의 자식요소에 포함돼 있는 텍스트로 요소 하나 추출	find_elements_by_partial_link
find_element_by_tag_name(name)	태그이름이 name에 해당하는 요소 하나 추출	find_elements_by_tag_name
find_element_by_link_text(text)	링크 텍스트로 요소 하나 추출	
find_element_by_xpath(query)	Xpath를 지정해 요소 하나 추출	find_elements_by_xpath
find_element_by_css_selector(query)	css 선택자로요소 하나 추출	find_elements_by_css_selector

Unit 03 | 크롤링이 안되는 경우

[셀레늄]

Selenium 요소 조작

clear()	글자 입력란에 글자를 지움
click()	요소를 클릭
get_attribute(name)	Name에 해당하는 값을 추출
is_displayed()	요소가 화면에 출력되는지 확인
is_selected()	체크박스 등의 요소가 선택된 상태인지 확인
is_enabled()	요소가 활성화돼 있는지 확인
screenshot(filename)	스크린샷을 찍는다.
send_keys(value)	키를 입력한다.
submit()	입력 양식을 전송한다.
value_of_css_property(name)	Name에 해당하는 CSS속성의 값을 추출
Id	요소의 id속성
location	요소의 위치

parent	부모 요소
rect	크기와 위치정보를 가진 딕셔너리 자료형을 리턴
screenshot_as_base64	BASE64로 스크린샷을 추출
screenshot_as_png	PNG형식으로 스크린샷 추출
size	요소의 크기
tag_name	태그 이름
text	요소의 내부글자

Unit 03 | 크롤링이 안되는 경우

[유저 에이전트]

일부 웹 사이트는 클라이언트가 웹 브라우저 이외의 프로그램으로 접근하는 것을 차단한다. 그렇다면 서버는 클라이언트가 사용하는 프로그램을 어떻게 알 수 있을까? 약간 어이 없지만 클라이언트가 요청을 할 때 `User-Agent` 라는 이름으로 접속 프로그램의 종류를 전달해주기 때문이다. 바꿔말하면 웹 스크랩 프로그램도 웹 브라우저의 `User-Agent` 를 사용하면 차단된 사이트에 접속할 수 있다.

브라우저별 `User-Agent` 값은 useragentstring.com에서 찾아볼 수있다.

실습3→ 유저 에이전트를 통해 접속 프로그램이 정상적인 브라우저로 속여서
정상적인 접근- 크롤링이 가능해짐!!

Unit 03 | 크롤링이 안되는 경우

[리퍼러]

클라이언트는 서버에 요청을 보낼 때 여러 가지 부가 정보를 보낸다. 예를 들면 A라는 사이트에서 링크를 눌러 B라는 사이트로 접속할 경우에, B라는 사이트에 A를 리퍼러(Referer)로 전달하는 것이다. 리퍼러는 영어로 '추천인'이라는 뜻이다. 참고로 문법상으로 referrer이 맞지만, 표준안에 referer로 잘못 쓰였기 때문에 웹과 관련해서는 후자로 쓴다.

일부 웹 사이트의 경우에는 리퍼러를 이용해 접속을 허용하거나 차단하기도 한다. 예를 들어 네이버 카페의 경우 일정 등급 이상의 회원만 볼 수 있는 게시물도 검색을 하면 볼 수 있게 한다. 이 경우 리퍼러에 네이버 검색이 들어있으면 접근을 허용하는 것이다.

Unit 03 | 크롤링이 안되는 경우

[POST]

POST는 클라이언트에서 서버로 정보를 보내기 위한 요청 방법 중 하나!!!.
일반적으로는 로그인이나 파일 업로드, 글 쓰기 등을 위해 사용되지만
한국의 정부기관들은 검색이나 게시물 읽기 등에도 POST를 남용하는 경향이 있다.
이 경우 사용자가 전송한 정보에 따라 다른 결과를 보여주는 식이기 때문에
검색 결과나 게시물의 고유한 URL이 없어서 링크를 걸 수가 없다.

-→ 예를 들면 국토거래시스템에서 특정 지역의 토지가격을 알고싶다면
그 지역과 아파트 이름, 주거형태 등의 정보를 입력해야지
원하는 정보가 나오기 문에 일반적인 크롤링으로는 원하는 정보에 접근할수 없다!!

→ 클라이언트가 요구할 정보를 JSON형태로 만들어서 요청한다!!

Unit 04 | 그 외 유용한 SNS 크롤링

• 많은 공모전이나 실무에서 머신러닝을 할 , 필요한 데이터가 없는 경우가 대부분!!!
그러한 데이터들을 보충해주기 위해서 크롤링을 활용

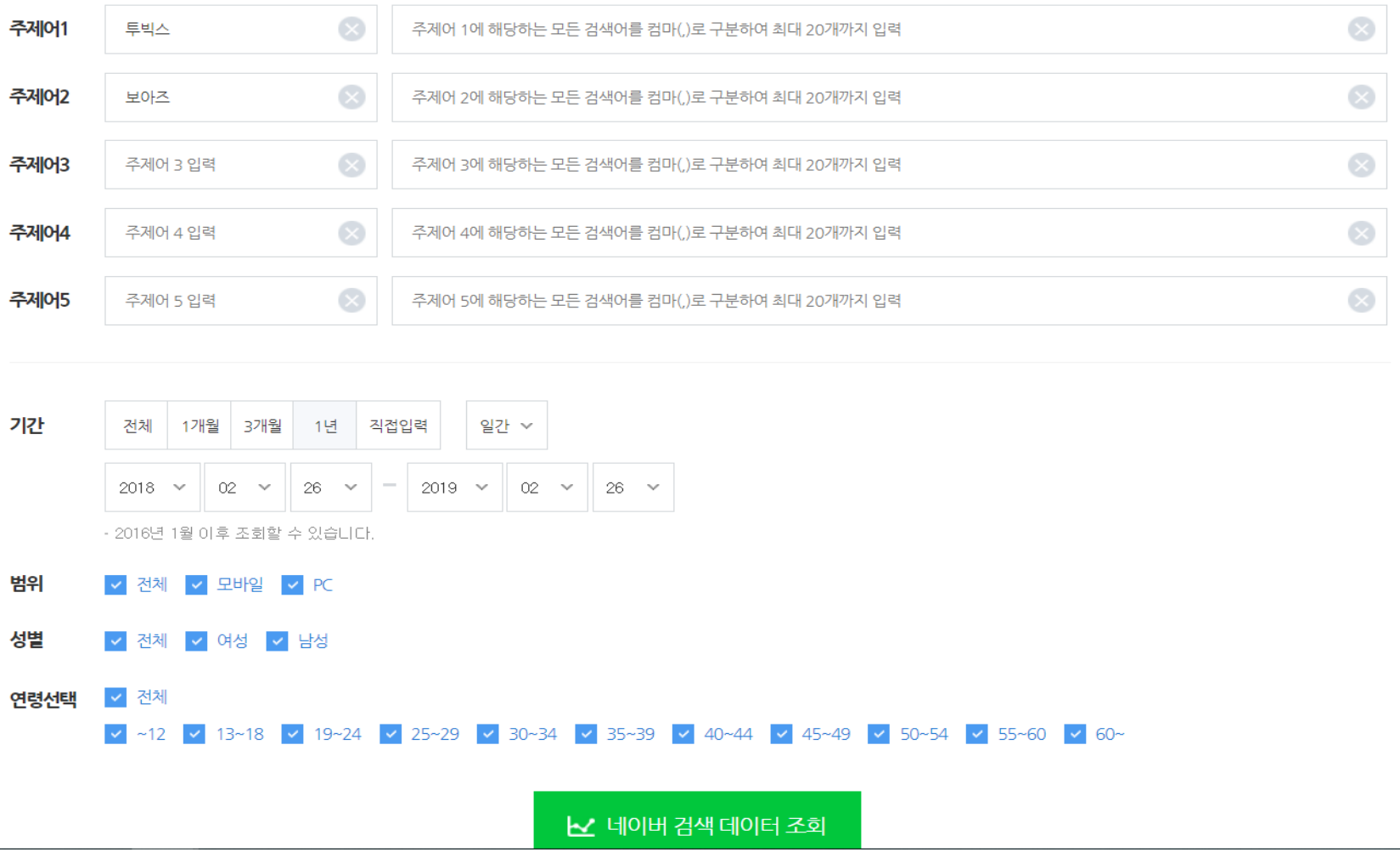
특히 SNS데이터들을 1)리얼 타임 2) 시계열 3)본질적으로 “반응 ” 의 특징들을 가지고 있어 매우 많이 쓰임.

하지만, 작년 6월 페이스북 개인정보 유출사건 이후 공식적으로 크롤링이 차단됨.
-> 셀레늄 적용 (너무 느림 , 멀티스레드 적용)
-> 검색데이터를 대신 활용

Unit 04 | 그 외 유용한 SNS 크롤링

- https://trends.google.co.kr/trends/?geo=KR
- https://datalab.naver.com/keyword/trendSearch.naver

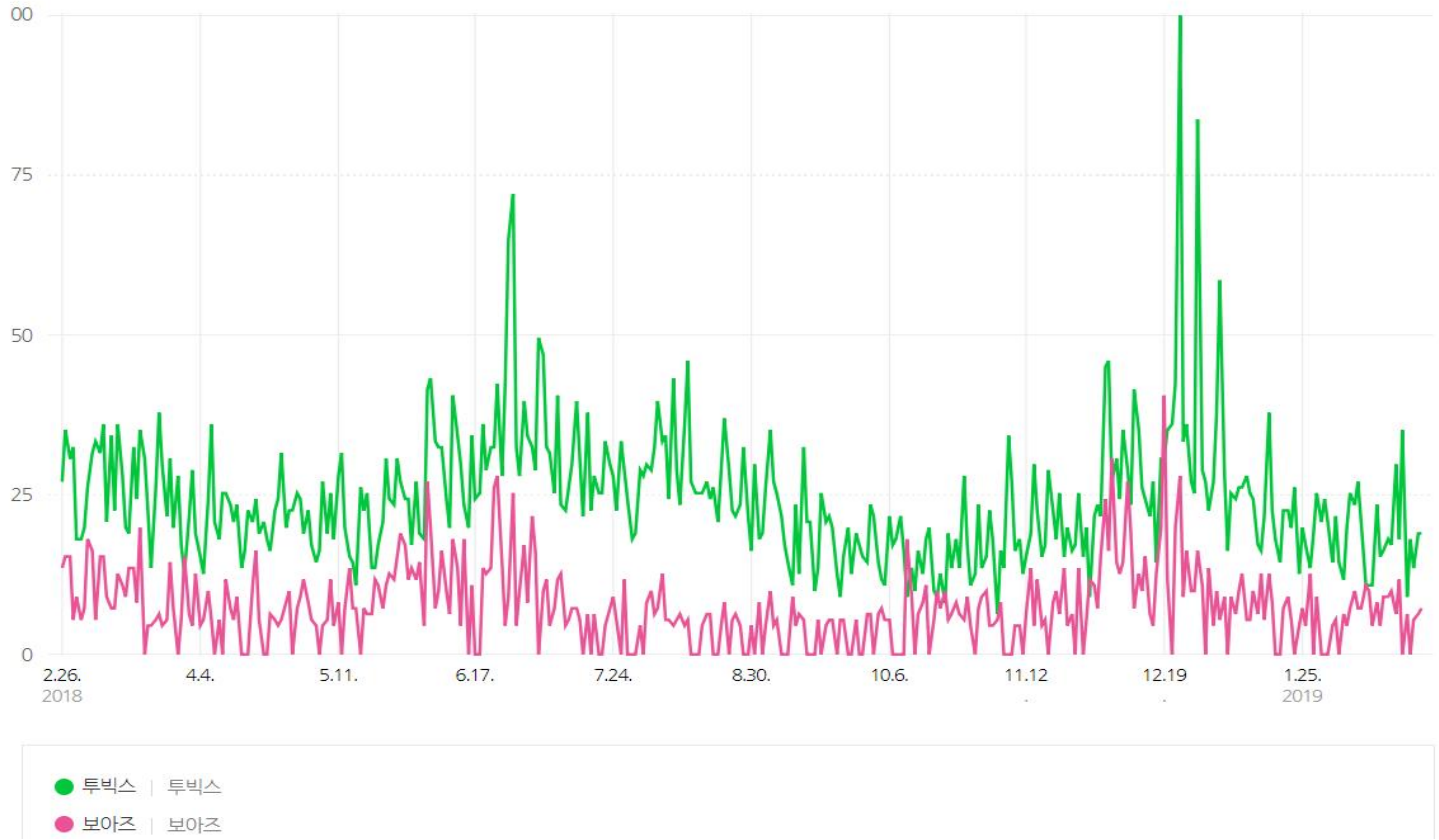
Unit 04 | 그 외 유용한 SNS 크롤링



투빅스 vs 보아즈

과연…?

Unit 04 | 그 외 유용한 SNS 크롤링



오호호
호호호
호호호

Unit 04 | 그 외 유용한 SNS 크롤링

data dependent on the endpoint selected.

These premium Search endpoints provide functionality beyond what's available in our standard search/Tweets endpoint, including:

- more Tweets per request
- higher rate limits
- a counts endpoint that returns time-series counts of Tweets
- more complex queries
- metadata enrichments, such as expanded URLs and improved profile geo information

To make it easy for you to adjust your use as your needs evolve, our premium APIs include flexible month-to-month contracts and scaled tiers of access based on the number of requests. To ensure it's easy to get started, our premium APIs include limited access within a free sandbox.

Pricing for the elevated tiers of the Search Tweets: 30-day endpoint start at $149/month for 500 requests, while pricing for the Search Tweets: Full-archive endpoint starts at $99/month for 100 requests.

트위터 역시 정책변경 이후로
일정이상의 데이터와 일주일 이전 데이터를 확인할려면 과금이 필요 10만원~

Unit 04 | 그 외 유용한 SNS 크롤링

- [사기템2]

- 아직 막히지 않은 트위터 크롤링 패키지를 발견!!

- 1만건 이상 ,2006년부터 전부 확인 가능!!

들어주셔서 감사합니다.