

[Python] python 에서 Seleium을 통한 동적 크롤링 - 2 — 나무늘보의 개발 블로그

노트북: 첫 번째 노트북

만든 날짜: 2020-10-27 오전 8:31

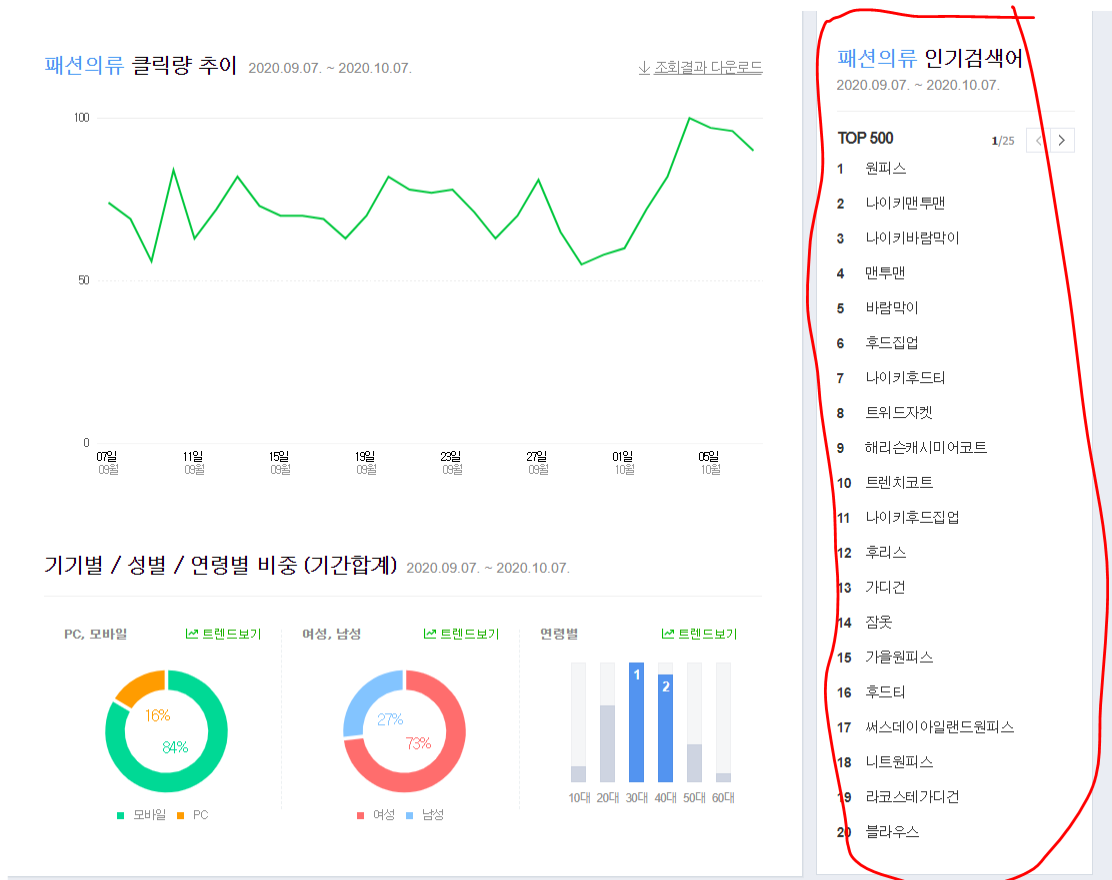
URL: <https://continuous-development.tistory.com/113?category=736681>

Python

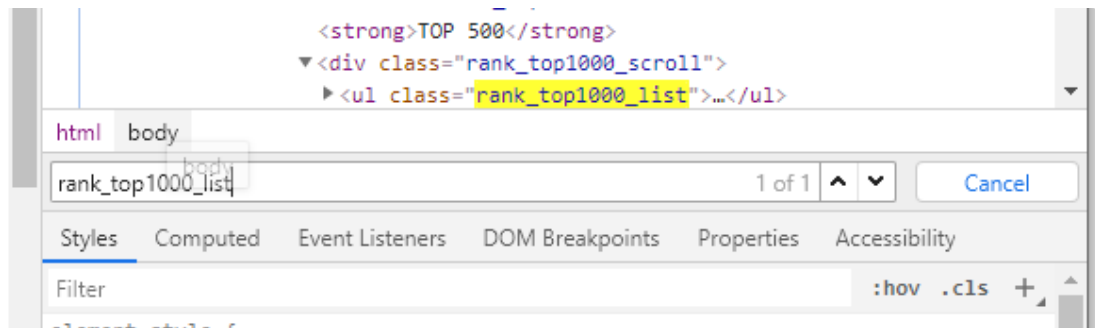
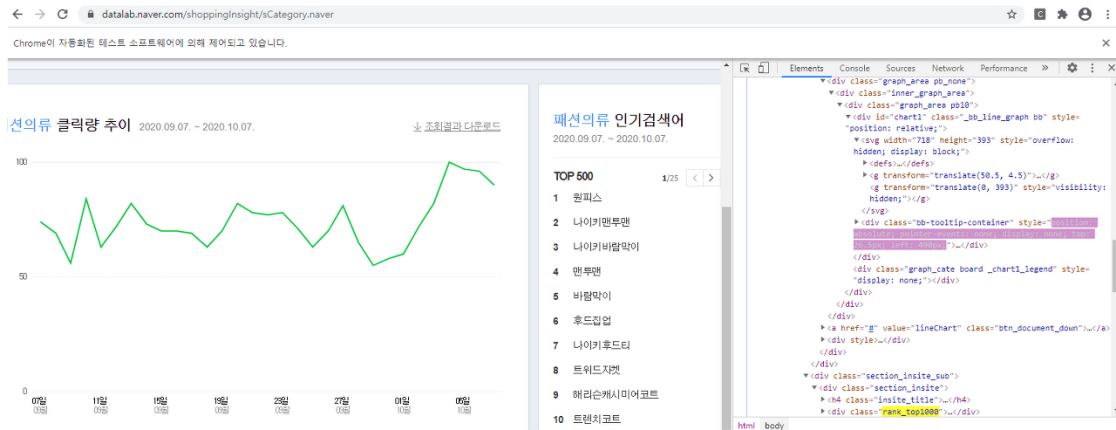
[Python] python 에서 Seleium을 통한 동적 크롤링 - 2

2020. 10. 8. 16:33 수정 삭제 공개

이제는 네이버 랩을 크롤링해보자



가져올 부분은 해당 부분이다.



해당 부분을 가져오려 할 때
기본적으로 이 함수들을 넣어주고

```

In [48]: from selenium import webdriver
path = './chromedriver.exe'
driver = webdriver.Chrome(path)
driver

Out [48]: <selenium.webdriver.chrome.webdriver.WebDriver (session="94f2f1beb314c7cd475b38ac6b13ee76")>

```

- Selenium 특정요소를 찾는 함수
- find_element_*
- find_elements_*
- find_element_by_id() # id
- find_element_by_name() # name
- find_element_by_css_selector() # class
- 리턴 시켜주는 타입 WebElement 객체를 리턴 text

해당 class를 가져오려 하기 때문에 css_selector를 가져오고 해당 클래스
명인. rank_top 1000_list를 가져온다.

```

In [73]: ul_tag = driver.find_element_by_css_selector('.rank_top1000_list')
print(ul_tag)

<selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="992d53b6-6b28-4ddc-8e9e-c935ad35160f")>

```

그 다음 ul태그였던 rank_top1000_list에서 li로 되어 있는걸 모두 가져온다.

```
In [74]: li_tag_list = ul_tag.find_elements_by_tag_name('li')
         print(li_tag_list)

[<selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="a0178a07-4541-4417-8fb2-b3b2657e1ed5")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="b3d56cc2-9298-4cc9-ba4c-bbdc9f6f43e9")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="f3e97b46-b5dd-4861-961e-ec82100f5a8")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="e98f54b9-d108-457b-b344-a99cb9f2ad1d")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="485ed1dc-7793-4b6c-8f71-ed37d2ccf3c")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="27af34d9-38af-4293-87e7-9ad256a96751")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="b609bb1d-6793-4b57-a97f-ba33685d7313")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="ee39b183-40d7-40f9-87ea-4db5c023a415")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="cb2f724-eaa0-4622-b7cc-e2ba11f8f0ec")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="9f126a6f-75f5-4070-622e-480c23a12787")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="cfc6205d-ddcc-4349-af28-b91c1938cdd3")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="8e6f6801-115b-4ff5-9eca-09a4663cd389")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="631d2093-b4dd-4caf-b764-1329baed2bf9")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="9d645ce9-b8a2-47c0-8623-aa7449b3fff4")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="5bfcf33e-10f6-438a-a3e0-d3cc6f843763")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="d92f4e42-337d-4867-9023-fb5a91a17a8a")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="ec99f0c0-d797-4c1b-bfe2-2b6663333db1")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="5d1f1f92-280d-4bd0-b39e-3172ef80da6")>, <selenium.webdriver.remote.webelement.WebElement (session="486dfcb851e51393c23419166b470841", element="64baf31d-9b31-4ea3-8db6-734adf63cd2c")>]
```

여기서 저장한 ul_tag에서 원하는 태그(li)를 가져오기위해 find_elements_by_tag_name('li')로 가져오게한다.

```
In [76]: for li in li_tag_list:
         print(li.text.split('❖'))

['1', '원피스']
['2', '나이키맨투맨']
['3', '나이키바람막이']
['4', '맨투맨']
['5', '바람막이']
['6', '후드집업']
['7', '나이키후드티']
['8', '트위드자켓']
['9', '헤리슨캐시미어코트']
['10', '트렌치코트']
['11', '나이키후드집업']
['12', '후리스']
['13', '가디건']
['14', '잠옷']
['15', '가을원피스']
['16', '후드티']
['17', '써스데이아일랜드원피스']
['18', '니트원피스']
['19', '라코스테가디건']
['20', '블라우스']
```

여기서가져온 li_tag_list를 하나씩 빼서 출력을 하는데 개행문자를 기준으로 나눠서 나오게 출력 한다.

[Python] python 에서 Seleium을 통한 동적 크롤링 - 4

[Python] python 에서 Seleium을 통한 동적 크롤링 - 3

[Python] python 에서 Seleium을 통한 동적 크롤링 - 2

[Python] python 에서 Seleium을 통한 동적 크롤링 - 1

[Python] python 에서 Seleium을 설치 방법

[Python] BeautifulSoup을 통한 이미지 블로그 스크래핑하기

python click 이벤트

python Seleium

python 동적 크롤링



나무늘보스

혼자 끄적끄적하는 블로그 입니다.