

크롤링 응용

파이썬과 BeautifulSoup을 이용하면
이 웹 크롤러를 간단하게 만들 수 있다.

네이버 뉴스의 '많이본 뉴스' 를 가져오기

```
import requests
from bs4 import BeautifulSoup

# 주소 :
# https://news.naver.com/main/ranking/popularDay.nhn?rankingType=popular_da

# 위의 주소에서 알 수 있듯이 맨 뒤에 날짜를 바꿔주면
# 해당하는 날짜의 많이 본 뉴스를 볼 수 있다.

url = "https://news.naver.com/main/ranking/popularDay.nhn?
rankingType=popular_day&date=20200421"

r = requests.get(url)
html = r.content
soup = BeautifulSoup(html, 'html.parser')

# 원하는 데이터 추출하기
# 네이버 많이 본 뉴스 페이지에서 헤드라인만 추출해서 출력
titles_html = soup.select('.ranking_section > ol > li > dl > dt > a')

# 30개의 헤드라인이 순서대로 출력
for i in range(len(titles_html)):
    print(i+1, titles_html[i].text)
```

삼성전자 주식 일별 시세 가져오기

네이버 증권에서 제공하는 삼성전자 종목(005930)의 일별시세를 가져오기

주소 : http://finance.naver.com/item/sise_day.nhn?code=005930

위의 주소와 같이 뒷부분에 code=005930와 같이 종목코드를 입력해주면 해당 종목의 일별시세를 볼 수 있다.

종목의 일별시세 페이지에서 날짜, 종가, 거래량만 추출해서 출력해 보겠습니다.

개발자 도구(ctrl+shift+I 또는 F12)를 통해 소스를 보면 날짜, 종가, 거래량이 나온 부분을 찾을 수 있다.

'table', 'tr', 'td' 태그 안의 텍스트임을 알 수 있다.

```
import requests
from bs4 import BeautifulSoup as bs

url="http://finance.naver.com/item/sise_day.nhn?code=005930"
r= requests.get(url)
html = r.content
soup = bs(html, 'html.parser')
print(soup.prettify())

# 종목의 코드와 페이지수를 입력하는 함수
def print_stock_price(code, page_num):

    # result에는 날짜, 종가, 거래량이 추가된다.
    result = [[], [], []]

    # 주소 뒷부분에 &page=2 와 같은 형식으로 연결해주면
    # 해당 페이지의 일별 시세를 볼 수 있다.
    for n in range(page_num):
        url = "http://finance.naver.com/item/sise_day.nhn?code=" + code + '

        r = requests.get(url)
        html = r.content
        soup = BeautifulSoup(html, 'html.parser')

        # table 안의 tr 태그를 리스트형태로 가져온다.
        tr = soup.select('table > tr')

        # 첫번째 tr 태그는 th 태그가,
        # 마지막 tr 태그는 페이지 넘버가 있어서 제외
        for i in range(1, len(tr)-1):

            # text가 없는 row가 존재.
            if tr[i].select('td')[0].text.strip():
                # text가 있는 row에 대해서
                # 첫번째(날짜), 두번째(종가), 일곱번째(거래량)
                # td 태그의 text를 가져온다.
                result[0].append(tr[i].select('td')[0].text.strip())
                result[1].append(tr[i].select('td')[1].text.strip())
                result[2].append(tr[i].select('td')[6].text.strip())

    for i in range(len(result[0])):
```

```
        print(result[0][i], result[1][i], result[2][i],)
```

```
#----- print_stock_price() END -----
```

```
# 해당 종목의 코드와 50페이지를 입력
```

```
stock_code = '005930'
```

```
pages = 50
```

```
# 날짜, 종가, 거래량이 최근순으로 출력
```

```
print_stock_price(stock_code, pages)
```