학습 정리

팀	빅나물	구성원	이지현, 조영현

일정	발제자	주제
8일차(6/5)	이지현	BeautifulSoup 사용법 및 간단 웹 파싱 기초(실습)

주요 내용 요약

BeautifulSoup을 활용한 웹 파싱 실습

- 변하는 데이터들을 자동화 시켜두어, 일정한 시간에 가져와 데이터베이스에 저장하기.

- 다음 금융 시가총액 상위 종목 가져오기

  다음 금융 사이트

```
url="http://finance.daum.net/"
  res=req.urlopen(url).read()
  soup=BeautifulSoup(res,"html.parser")

  # print('soup',soup.prettify())
  # .prettify() 이쁘게 보여주기 위해

  top=soup.select("ul#topMyListNo1 > li")

  for i,e in enumerate(top,1):
      (top,1) 시작 인덱스 1로 지정.
      print(i,",",e.find("a").string," : ", e.find("span").string)
```

- 네이버 금융 Top4 종목 가져오기

```
  url="https://finance.naver.com/"
  res=req.urlopen(url).read().decode('cp949')
  # utf-8 : 한글 깨짐, unicode_escape : 한글 깨짐
  soup=BeautifulSoup(res,"html.parser")

  # print(soup)

  top4=soup.select("tbody#_topItems1 > tr")

  i=1
  for e in top4:
      if e.find("a") is not None:
          print(i,e.select_one("a").string)
          i+=1
```

- 인프런 추천 강좌 10개 가져오기

현재 홈페이지 변경되어 이 방식으로 가져와지지 않음.

```
base="https://www.inflearn.com/"
quote=rep.quote_plus("추천-강좌")
print(quote)

url=base+quote
res=req.urlopen(url).read()
soup=BeautifulSoup(res,"html.parser")

recommend=soup.select("ul.slides")[0]

for i,e in enumerate(recommend,1):
    print(i,e.select_one("h4.block_title>a"))
```

실습: 다음 실시간 인기 검색어 + link 스크랩핑 해보기

다음 실시간 이슈 검색어 Top10가져오기

```
from bs4 import BeautifulSoup
import urllib.request as req
import urllib.parse as rep
import sys
import io

sys.stdout=io.TextIOWrapper(sys.stdout.detach(),encoding='utf-8')
sys.stderr=io.TextIOWrapper(sys.stderr.detach(),encoding='utf-8')

url="https://www.daum.net/"
res=req.urlopen(url).read()
soup=BeautifulSoup(res,"html.parser")
# print(soup.prettify())

top10=soup.select("div.realtime_part>ol.list_hotissue>li>div>div:nth-of-type(1)>span.txt_
issue>a")
for i,e in enumerate(top10,1):
    print(i,e.string,e["href"])
```

- **3Vs**

  Volume: 데이터의 양

  Variety: 데이터의 다양성

  Velocity: 데이터의 속도

- 네이버에서 원하는 사진 한 번에 다운로드 받기

```
from bs4 import BeautifulSoup
import urllib.request as req
import urllib.parse as rep
import sys
import io
import os

sys.stdout=io.TextIOWrapper(sys.stdout.detach(),encoding='utf-8')
sys.stderr=io.TextIOWrapper(sys.stderr.detach(),encoding='utf-8')

base="https://search.naver.com/search.naver?where=image&sm=tab_jum&query="
quote=rep.quote_plus("박보영")
url=base+quote
print(url)

res=req.urlopen(url)
savePath="C:\\imagedown\\" # c:\imagedown\

try:
    if not (os.path.isdir(savePath)):
        os.makedirs(os.path.join(savePath))
# 권한이 없어서 폴더 만드는 것 실패한 경우
except OSError as e:
    if e.errno != errno.EEXIST:
        print("폴더 만들기 실패!")
        raise #에러 강제로 실행 시킴

soup=BeautifulSoup(res,"html.parser")

img_list=soup.select("div.img_area._item >a.thumb._thumb>img")

for i,img_list in enumerate(img_list,1):
    # print(img_list['data-source'])
    fullFileName=os.path.join(savePath,savePath+str(i)+'.jpg')
    # 폴더에 저장될 이름
    req.urlretrieve(img_list['data-source'],fullFileName)

print("다운로드 완료")
```

- 브라우저를 이용해서 접속한 경우, user-agent 값이 브라우저로 request로 날아감.

네이버에서 이 값을 보고 response를 해줄 때(브라우저에서 최종적으로 rendering될 때), jquery라던지 그 프레임워크에 따라 화면이 만들어짐.

but, 실습 시, 브라우저를 사용한 것이 아니라 http 통신으로 값을 날렸기 때문에 태그 요소와 브라우저에서 보는 것이 다를 수 있다.

src값이 아니라 data-source값이 이미지 소스 값이 들어 있음!

참고 사이트 인프런 파이썬 입문 및 웹 크롤링을 활용한 다양한 자동화 어플리케이션 제작하기 https://www.inflearn.com/course/python-%ED%8C%8C%EC%9D%B4%EC%8D%AC-%EC%9B%B9-%EB%8D%B0%EC%9D%B4%ED%84%B0-%ED%81%AC%EB%A1%A4%EB%A7%81/dashboard