

## 학습 정리

팀	이진조	구성원	이규진, 조현진
---	-----	-----	----------

일정	발제자	주제
5/31	조현진	requests 모듈 기초, 웹 브라우저 없는 스크래핑 및 파싱 실습(인프런, 네이버 카페 자동화)

### 주요 내용 요약

## http 통신기초 - cookies, session, request, response

- request, response 간단 개념 알아보기  
request: 요청  
response: 응답
- cookie, session 개념 알아보기
- http 통신
  1. 비연결지향
  2. 상태정보 유지 안함(쿠키와 세션)  
쿠키 : 자동 로그인, 오늘하루는 이 창을~~, 쇼핑몰 장바구니  
세션 : 서버

## requests 모듈 기초(1)

- requests 설치  
: pip install requests
- requests 모듈사용법 (1) 및 장점 - urllib
  - naver로 요청 보내보기

```
import sys
import io
import requests

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

s = requests.Session()

r = s.get('https://www.naver.com') # PUT(FETCH), DELETE, GET ,POST
print('1',r.text)

s.close() # requests를 닫아줘야한다
```
  - 다른방법의 요청

```

import sys
import io
import requests

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

s = requests.Session()

# 1.
r = s.get('http://httpbin.org/cookies', cookies={'from': 'myName'})
print(r.text)

# 2.
url = 'http://httpbin.org/get'
headers = {'user-agent': 'myPythonApp_1.0.0'}

r = s.get(url, headers=headers)
print(r.text)

s.close()

```

- with 문 사용(close 와는 다른방법)

```

with requests.Session() as s:
    r = s.get('https://www.naver.com')
    print(r.text)

```

- Json 데이터 핸들링

- <https://jsonplaceholder.typicode.com>
- album 불러오기

```

r = s.get('https://jsonplaceholder.typicode.com/albums')

```

```

print(r.text)

```

- 딕셔너리 값으로 가져오기

```

r = s.get('https://jsonplaceholder.typicode.com/posts/1')

```

```

print(r.json())
print(r.json().keys())
print(r.json().values())
print(r.encoding)
print(r.content) # 바이너리값으로 나옴
print(r.raw)

```

- requests 모듈 테스트 실습

```

import sys
import io
import requests, json

```

```

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

```

```

s = requests.Session()

```

```

r = s.get('http://httpbin.org/stream/20', stream=True)

```

```

# print(r.text)
# print(r.encoding)
# print(r.json())
if r.encoding is None:

```

```

r.encoding = 'utf-8'

for line in r.iter_lines(decode_unicode=True):
    # print(line)
    b = json.loads(line) # dict
    # print(b['origin'])
    for e in b.keys():
        print("key:", e, "values:", b[e])

```

## request 모듈 기초 (2)

- requests 모듈 사용법 (2)

```

import sys
import io
import requests, json

```

```

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

```

```

r = requests.get('https://api.github.com/events')
r.raise_for_status()
print(r.text)

```

```

jar = requests.cookies.RequestsCookieJar()
jar.set('name', 'kim', domain='httpbin.org', path='/cookies')

```

```

r = requests.get('http://httpbin.org/cookies', cookies=jar)
r.raise_for_status()
print(r.text)

```

```

r = requests.get('https://github.com', timeout=3)
print(r.text)

```

```

r = requests.post('https://httpbin.org/post', data={'name': 'kim'}, cookies=jar)
print(r.text)

```

```

payload1 = {'key1': 'value1', 'key2': 'value2'} # dict
payload2 = (('key1', 'value1'), ('key2', 'value2')) # tuple
payload3 = {'some': 'nice'}

```

```

# form
r = requests.post('http://httpbin.org/post', data = payload2)
print(r.text)

```

```

# json
r = requests.post('http://httpbin.org/post', data = json.dumps(payload3))
print(r.text)

```

- requests 모듈 Rest API 실습

```

import sys
import io
import requests, json

```

```
# Rest : POST, GET, PUT:UPDATE, REPLACE (FETCH : UPDATE, MODIFY), DELETE
```

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

```
payload1 = {'key1': 'value1', 'key2': 'value2'} # dict  
payload2 = (('key1','value1'), ('key2','value2')) # tuple  
payload3 = {'some':'nice'}
```

```
# put
```

```
r = requests.put('http://httpbin.org/user/delete', data=payload1)  
print(r.text)
```

```
r = requests.put('http://jsonplaceholder.typicode.com/posts/1', data=payload1)  
print(r.text)
```

```
# delete
```

```
r = requests.delete('http://jsonplaceholder.typicode.com/posts/1')  
print(r.text)
```

## 파이썬 requests 통신 실습 고급(1)

- 루리웹(Ruliweb) 사이트 로그인 처리 후 게시판 글 가져오기
  - 로그인과 로그인이 필요한 게시글 확인하기

```
import sys  
import io  
from bs4 import BeautifulSoup  
import requests
```

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

```
# 로그인 유저정보
```

```
LOGIN_INFO = {  
    'user_id': 'outsider7224',  
    'user_pw': '11111112!!!'  
}
```

```
# Session 생성, with 구문안에서 유지
```

```
with requests.Session() as s:  
    login_req =  
    s.post('https://user.ruliweb.com/member/login_proc', data=LOGIN_INFO)  
    # HTML 소스확인  
    print('login_req', login_req.text)  
    # Header 확인  
    print('headers', login_req.headers)
```

```
# 로그인이 되었을때 특정 게시글의 글을 가져올때
```

```
if login_req.status_code == 200 and login_req.ok:  
    post_one = s.get('글의 주소') # get방식  
    post_one.raise_for_status() #예외일때의 방식
```

```

soup = BeautifulSoup(post_one.text, 'html.parser')
# print(soup.prettify())
article = soup.select_one("table:nth-of-type(3)").find_all('p')
# print(article)
for i in article:
    if i.string is not None:
        print(i.string)

```

- 인프런(Inflearn) 사이트 로그인 처리 후 개인정보 가져오기

```

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

```

```

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

```

# 로그인 유저정보

```

LOGIN_INFO = {
    'user_id': 'ID',
    'user_pw': 'PASSWORD',
    'user-submit': rep.quote_plus('로그인'),
    'user-cookie': 1
}

```

# Session 생성, with 구문안에서 유지

```

with requests.Session() as s:
    login_req = s.post('페이지 주소', data=LOGIN_INFO)
    # HTML 소스확인
    # print('login_req', login_req.text)
    # Header 확인
    # print('headers', login_req.headers)
    if login_req.status_code == 200 and login_req.ok: # 로그인 후 특정 글을 가져올때
        post_one = s.get('글의 주소') # get방식
        post_one.raise_for_status() #예외일때의 방식
        soup = BeautifulSoup(post_one.text, 'html.parser')
        # print(soup.prettify())
        badges = soup.select("div.badges > ul > li > a >img")
        for i, z in enumerate(badges, 1):
            fullFileName = os.path.join("c:/", str(i) + '.jpg')
            req.urlretrieve(z['src'], fullFileName)

```

