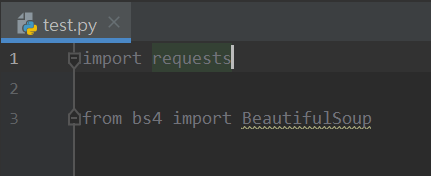
3주차 : 파이썬으로 데이터 수집하기

기존 프로젝트 close project 로 닫고 홈 화면에서 새로운 프로젝트 다시 만들기(프로젝트 이름은 가능하면 영어로)

패키지 설치 완료 확인하기



Stage 2

Requests는 코드를 가지고 와주는 역할

Bs4를 씌우면 하나를 가지고 올 수 있다. 선택자 가져올 수도 있음.

print(row) # 200이 아닌 다른 숫자가 나오면 오류이다.

print(row.text) # 써놓은 주소의 소스코드

지금까지의 전체 코드

import requests  
from bs4 import BeautifulSoup  
  
# requests.get("긁고 싶은 url")  
row = requests.get("https://tv.naver.com/r/")  
print(row.text) # 써놓은 주소의 소스코드

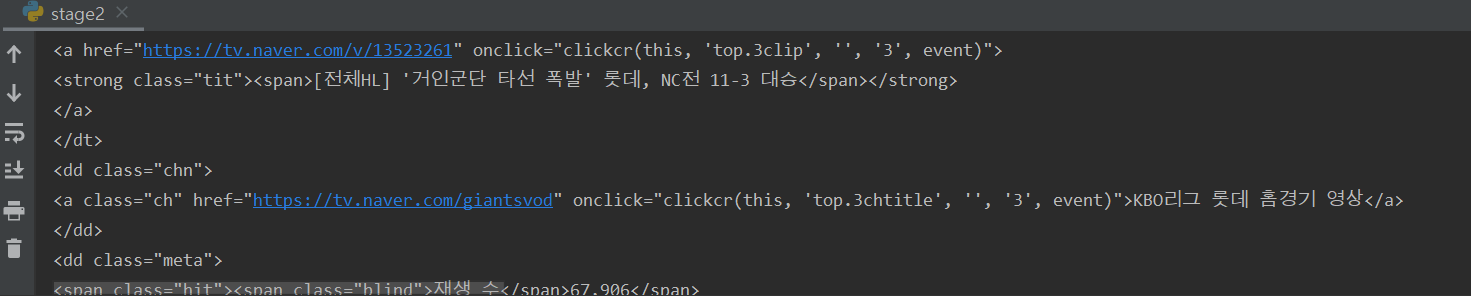
코드 전체 출력한다.

# BeautifulSoup(소스코드, 파싱 종류)  
html = BeautifulSoup(row.text,'html.parser')  
print(html)

선택자로 원하는 코드만 긁어내기

#html.select('선택자')  
# div.inner와 관련된 코드만 나온다.  
clips = html.select('div.inner')  
print(clips[2]) # 0,1,2 순서에 맞게 해당 코드만 나오게 한다.

#clips 는 리스크![항목1,항목2,항목3]

결과물>2즉3위에해당하는코드가나온다

복수개의 대상자이다. 그러면 select 사용. 하나의 상황에서는 select\_one 사용

print(clips[0].select\_one('dt.title').text) #제목 클립 안에 제목은 하나니까 select\_one 사용 text는 글자만 나오게 한다.

# 제목 클립 안에 제목은 하나니까 select\_one 사용 text는 글자만 나오게 한다.  
title = clips[0].select\_one('dt.title').text  
print(title)  
  
# 채널명 재생수 좋아요  
chn = clips[0].select\_one('dd.chn').text  
hit = clips[0].select\_one('span.hit').text  
like = clips[0].select\_one('span.like').text  
print(chn, hit, like)

반복문을 사용하지 않고 클립 1~3위 정보 출력하기 (너무 복잡하다.)

