Python News Crawling TIL 2

2022-03-08

수행 단계

~~STEP 1. 소스 조사 : 제공 사이트 조사, 제공 정보 조사, 확보 가능 정보 확인~~

~~STEP 2. 웹 구성 체계 확인 : HTML 구조 확인, 제공 정보 확인, 크롤링 가능 여부 확인~~

STEP 3. 크롤링 진행 : parsing 방안 및 위치 확인, request 방법 확인, data 저장 형태 설계

STEP 4. 최종 데이터 생성 : 데이터 저장 형태, 데이터 저장

소스 조사 및 웹 구성 체계 확인 결과

- **네이버 뉴스의 기사 URL 형태 (우크라이나를 키워드로 검색 시)**
  - https://search.naver.com/search.naver?where=news&ie=utf8&sm=nws_hty&query=우크라이나
- **네이버 뉴스 HTML 구성 (뉴스 title, URL 찾을 시)**
  - source : https://search.naver.com/search.naver?where=news&ie=utf8&sm=nws_hty&query=”키워드”
  - input : 검색어 (예 : 우크라이나), 필요한 뉴스 기사 수 (예 : 25)
  - output : 뉴스 제목, URL 이 담긴 엑셀
- **크롤링 사용 툴 : Python (Requests, Beautifulsoup 패키지)**
  - Requests : request 를 보내 웹 페이지 소스 받음
  - Beautifulsoup : 웹 페이지의 소스를 parsing 하여 원하는 정보를 찾음

STEP 3. 크롤링 진행

- **필요 환경 및 패키지 설치**
  - python

  - requests, bs4, re, pandas 패키지 사용
    - requests : 웹 페이지 소스 추출 (HTML)

    - bs4 : HTML 파싱, 필요 태그 및 소스 추출
    - re : 조건부 문자열 (정규 표현식), 태그 탐색 시 일반화 조건을 사용하기 위함
    - pandas : 데이터 프레임, 엑셀 변환
- 코드 작성
    - 패키지 importing

```
# 패키지 importing

import requests
from pandas import DataFrame
from bs4 import BeautifulSoup
import re
from datetime import datetime
import os
```

```
# 편집 모드 단축키
shift + enter : run cell, select below
ctrl + enter : run cell
alt + enter : reun cell, insert below

# mode 변경
ctrl + m / esc

# Command 모드 단축키
b : 셀 추가 (아래로 추가)
a : 셀 추가 (위로 추가)
dd : 해당 셀 삭제
```

    - 현재 시간 저장

```
# 현재 시간 저장 : 나중에 output 으로 엑셀 저장 시 크롤링 한 날짜, 시간을 파일명에 넣기 위해 저장하는 변수

date = str(datetime.now())
date = date[:date.rfind(':')].replace(' ', '_')
date = date.replace(':','시') + '분'
```

    - input 생성

```
query = input('검색 키워드를 입력하세요 : ')
query = query.replace(' ', '+')
news_num = int(input('총 필요한 뉴스기사 수를 입력해주세요(숫자만 입력) : '))
```

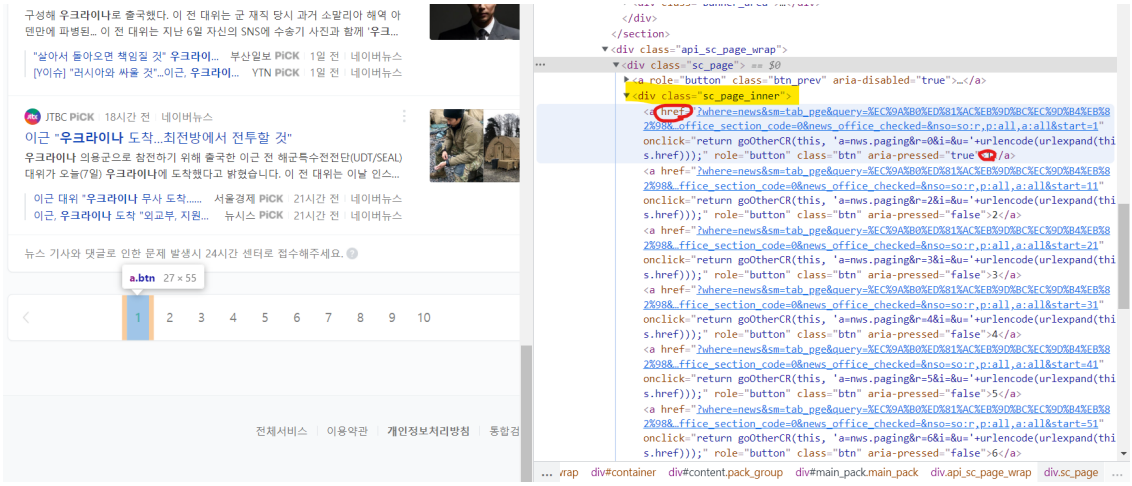
    - 요정할 URL 생성 및 요청

```
# 요청할 URL 생성 및 요청 :
# query 로 받은 키워드를 URL 조건절 중 키워드에 해당하는 변수에 대응시켜 요청 URL 을 만든다
# 그리고 requests 패키지의 get함수를 이용하여 HTML 코드를 받아온다
# 받은 코드를 bs4 BeautifulSoup 함수를 이용하여 파싱한다

news_url = 'https://search.naver.com/search.naver?where=news&sm=nws_hty&query={}'
req = requests.get(news_url.format(query))
soup = BeautifulSoup(req.text, 'html.parser')
```

    - 원하는 정보를 담을 변수 생성 (딕셔너리)

```
# 원하는 정보를 담을 변수 생성 (딕셔너리) :
# 뉴스 기사 정보를 저장할 딕셔너리를 생성 (key : 번호, value : 뉴스 기사 정보)
# idx : 현재 뉴스의 번호
# cur_page : 네이버 뉴스의 웹 페이지, 추출하려는 기사 수가 현재 페이지에 있는 기사보다 많은 경우 다음페이지로 넘어가야 하기 때문에 현 페이지 번호를 기억하도

news_dict = {}
idx = 0
cur_page = 1
```

    - parsing 한 HTML 코드에서 원하는 정보 탐색 (뉴스 기사 title, URL)




```
# parsing 한 HTML 코드에서 원하는 정보 탐색 (뉴스 기사 title, URL)

print()
print('크롤링 중...')

# 현재 뉴스 기사 번호 (idx) 가 원하는 뉴스 기사 수(news_num) 보다 작은 동안 아래 코드 실행
while idx < news_num:

    table = soup.find('ul',{'class' : 'list_news'}) # 뉴스 바운딩 박스 <ul>
    li_list = table.find_all('li', {'id': re.compile('sp_nws.*')}) # 뉴스 바운딩 박스 안의 각 뉴스 기사 <li>
    area_list = [li.find('div', {'class' : 'news_area'}) for li in li_list] # 뉴스 기사 안의 뉴스 제목, 본문이 담긴 태그 <div>
    a_list = [area.find('a', {'class' : 'news_tit'}) for area in area_list] # 각 뉴스 기사 내부 title, URL 정보가 담긴 태그 <a>

    for n in a_list[:min(len(a_list), news_num-idx)]:
        news_dict[idx] = {'title' : n.get('title'),'url' : n.get('href') }
        idx += 1
    cur_page += 1

    # 현재 수집한 뉴스 기사 수가 부족한 경우 다음 페이지로 넘어가야 하므로 다음 페이지에 해당하는 URL 추출 <div class="sc_page_inner">
    # 하위에 존재하는 a 태그 내부에 페이지 번호와 URL 정보 존재
    # cur_page 변수와 일치하는 페이지 번호의 URL 가져옴
    pages = soup.find('div', {'class' : 'sc_page_inner'})
    next_page_url = [p for p in pages.find_all('a') if p.text == str(cur_page)][0].get('href')
    req = requests.get('https://search.naver.com/search.naver' + next_page_url)
    soup = BeautifulSoup(req.text, 'html.parser')
```

- 데이터 프레임 변환 및 저장

```
# 데이터 프레임 변환 및 저장

print('크롤링 완료')

# 크롤링한 뉴스 정보가 담긴 딕셔너리를 데이터 프레임으로 변환
print('데이터프레임 변환')
news_df = DataFrame(news_dict).T

# 크롤링한 키워드와 크롤링 날짜를 엑셀 파일 명으로 하여 저장
folder_path = os.getcwd()
```

```
xlsx_file_name = '네이버뉴스_{}_{}.xlsx'.format(query, date)
news_df.to_excel(xlsx_file_name)

# 저장을 완료한 폴더를 띄움
print('엑셀 저장 완료 | 경로 : {}\\{}'.format(folder_path, xlsx_file_name))
os.startfile(folder_path)
```

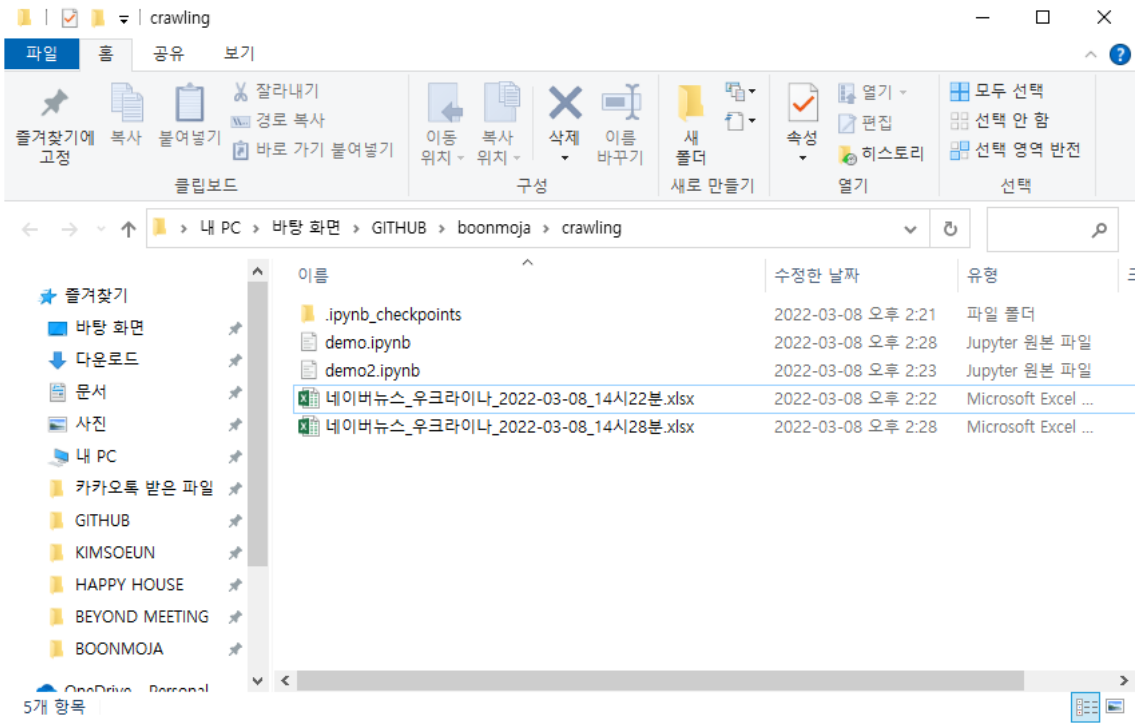
STEP 4. 최종 데이터 생성

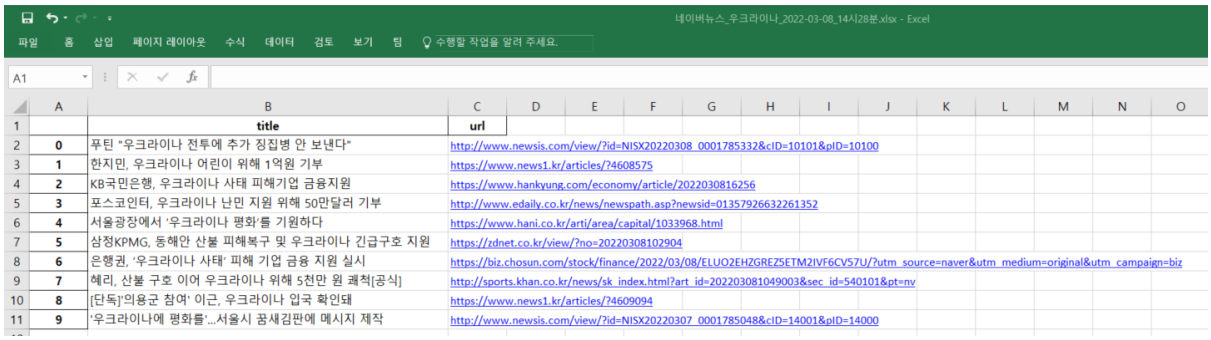
```
    area_list = [area.find('a', {'class' : 'news_tit'}) for area in area_list] # 각 뉴스

    for n in a_list[:min(len(a_list), news_num-idx)]:
        news_dict[idx] = {'title' : n.get('title'),'url' : n.get('href') }
        idx += 1
    cur_page += 1

    # 현재 수집한 뉴스 기사 수가 부족한 경우 다음 페이지로 넘어가야 하므로 다음 페이지
    # 하위에 존재하는 a 태그 내부에 페이지 번호와 URL 정보 존재
    # cur_page 변수와 일치하는 페이지 번호의 URL 가져옴
    pages = soup.find('div', {'class' : 'sc_page_inner'})
    next_page_url = [p for p in pages.find_all('a') if p.text == str(cur_page)][0].ge
    req = requests.get('https://search.naver.com/search.naver' + next_page_url)
    soup = BeautifulSoup(req.text, 'html.parser')
```

크롤링 중...

In [51]:
```
# 데이터 프레임 변환 및 저장

print('크롤링 완료')

# 크롤링한 뉴스 정보가 담긴 딕셔너리를 데이터 프레임으로 변환
print('데이터프레임 변환')
news_df = DataFrame(news_dict).T

# 크롤링한 키워드와 크롤링 날짜를 엑셀 파일 명으로 하여 저장
folder_path = os.getcwd()
xlsx_file_name = '네이버뉴스_{}_{}.xlsx'.format(query, date)
news_df.to_excel(xlsx_file_name)

# 저장을 완료한 폴더를 띄움
print('엑셀 저장 완료 | 경로 : {}₩₩{}'.format(folder_path, xlsx_file_name))
os.startfile(folder_path)
```

```
크롤링 완료
데이터프레임 변환
엑셀 저장 완료 | 경로 : C:₩Users₩multicampus₩Desktop₩GITHUB₩boonmoja₩crawling₩네이버뉴스_우크라이나_2022-03-08_14시28분.xlsx
```





		title	url
2	0	푸틴 "우크라이나 전투에 추가 징집병 안 보낸다"	http://www.newsis.com/view/?id=NISX20220308_0001785332&cID=10101&pID=10100
3	1	한지민, 우크라이나 어린이 위해 1억원 기부	https://www.news1.kr/articles/?4608575
4	2	KB국민은행, 우크라이나 사태 피해기업 금융지원	https://www.hankyung.com/economy/article/2022030816256
5	3	포스코인터, 우크라이나 난민 지원 위해 50만달러 기부	http://www.edaily.co.kr/news/newspath.asp?newsid=01357926632261352
6	4	서울광장에서 '우크라이나 평화'를 기원하다	https://www.hani.co.kr/arti/area/capital/1033968.html
7	5	삼정KPMG, 동해안 산불 피해복구 및 우크라이나 긴급구호 지원	https://zdnet.co.kr/view/?no=20220308102904
8	6	은행권, '우크라이나 사태' 피해 기업 금융 지원 실시	https://biz.chosun.com/stock/finance/2022/03/08/ELUO2EHZGREZ5ETM2IVF6CV57U/?utm_source=naver&utm_medium=original&utm_campaign=biz
9	7	헤리, 산불 구호 이어 우크라이나 위해 5천만 원 쾌척[공식]	http://sports.khan.co.kr/news/sk_index.html?art_id=202203081049003&sec_id=540101&pt=nv
10	8	[단독]'의용군 참여' 이근, 우크라이나 입국 확인돼	https://www.news1.kr/articles/?4609094
11	9	'우크라이나에 평화를'...서울시 꿈새김판에 메시지 제작	http://www.newsis.com/view/?id=NISX20220307_0001785048&cID=14001&pID=14000