[Python] BeautifulSoup을 통한 이미지 스크래핑 하기 — 나무늘보의 개발 블로그

노트북:	첫 번째 노트북
만든 날짜:	2020-10-25 오후 5:14
URL:	https://continuous-development.tistory.com/108?category=736681

Python

[Python] BeautifulSoup을 통한 이미지 스크래핑 하기

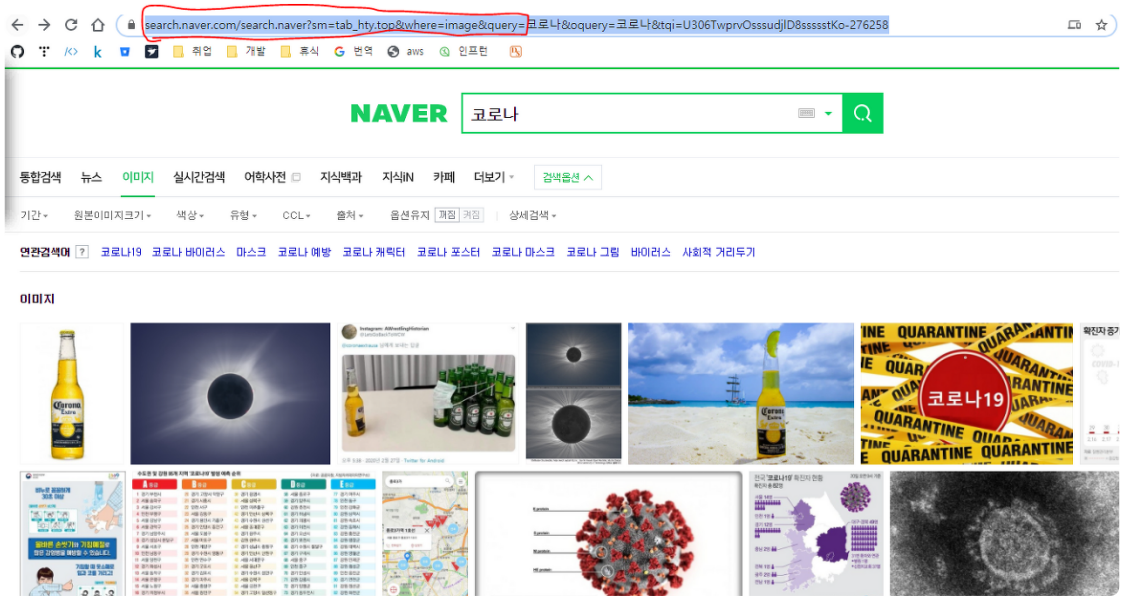
2020. 10. 7. 17:38 수정 삭제 공개

※환경은 anacond + jupyter에서 하였습니다.

In [3]:
```
from urllib.request import urlopen
from bs4 import BeautifulSoup
from urllib.error import HTTPError
from urllib.error import URLError
import pandas as pd
from urllib.parse import quote_plus
```

크롤링을 하는 데 있어서 기본적으로 필요한 함수들이다. 없으면 anacon da prompt에서 pip install을 통해 받도록 하자

내가 크롤링하려는 사이트는 네이버다. 네이버에서 검색어를 입력하고 그 이미지를 가져오기로 하자.
첫 번째로 url을 가져오자.



저렇게 검색어를 기준으로 앞에 url을 가져온다. 검색어를 입력하는 것에 따라 이미지를 가져오기 위해서이다.

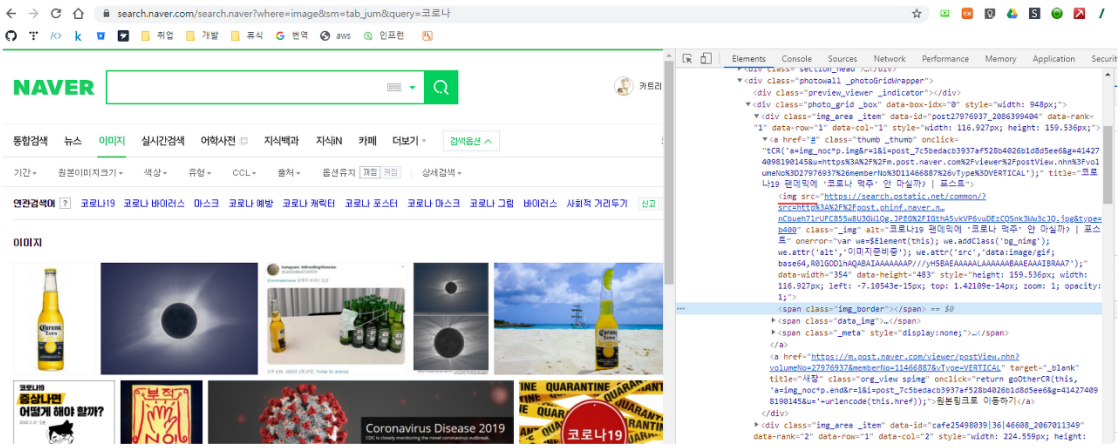
base_url에 검색어 url을 넣어주고
keyword에는 내가 넣을 검색어를 넣어주기 위해 input을 통해 검색어를 입력한다.
ImgCnt는 스크랩할 이미지 개수를 정해준다.

```
In [5]: base_url = 'https://search.naver.com/search.naver?sm=t&where=image&query=' # base url 이다. 여기서 뒤에 어떤 검색어를 넣냐에 따라 달라진다.
keyword = input('검색어 입력 :') # 여기서 네이버에서 이미지 검색하던것 처럼 명령어를 입력해준다.
imgCnt = int(input('스크랩 할 이미지 개수 : ')) # 스크랩 할 이미지 개수를 지정해준다.
try:
    html = urlopen(base_url+quote_plus(keyword)) # quote_plus 를 통해 URL Encoding을 한다.
except HTTPError as he:
    print('http error')
except URLError as us:
    print('url error')
else:
    soup = BeautifulSoup(html.read(),'html.parser')

검색어 입력 :코로나
스크랩 할 이미지 개수 : 50
```

위와 같은 식으로 로직을 만들어준다. 내가 검색어를 입력하는 거에 따라 결과가 달라지는 크롤링 코드를 만들어보자.

사이트를 스크래핑하고 내가 원하는 부분이 어디 있는지 확인한다.



여기서 나는 img를 스크래핑하기 위해 아래와 같이 img를 가져온다.

```
In [7]: img  = soup.find_all('img','_img')
```

```
In [8]: img
```

```
Out[8]: [<img alt="코로나19 팬데믹에 '코로나 맥주' 안 마실까? | 포스트" class="_img" data-height="483" data-source="https://search.pstatic.net/common/?src=http%3A%2F%2Fpost.phinf.naver.net%2FMjAyMDA0MTNfMTgz%2FMDAxNTg2NzM1NzM0ODM4.petATgzOXbpnp4_S35joz2eVvLy26YX8yc59Qx0p0P8g.f83p-xTOuvjclLd7wG6snCbueh7IrUFC855w8U3GWIQg.JPEG%2FIGthA5vkVP6vuDEzCQSnk3Ww3cJQ.jpg&amp;type=b400" data-width="354" onerror="var we=$Element(this); we.addClass('bg_nimg'); we.attr('alt','이미지준비중'); we.attr('src','');" src=""/>,
 <img alt="코로나19는 왜 코로나19인 이유 | 카페" class="_img" data-height="449" data-source="https://search.pstatic.net/common/?src=http%3A%2F%2Fcafefiles.naver.net%2FMjAyMDAyMjJfOTQg%2FMDAxNTgyMzUONTg5MzMz.9maUHUY9Uw4aqeCg-tnsCShALxfcx9GnKm5qg7oa3Fgg.jzVlcswHYV1mPC_NrmNXmVE2N_2j4nVOgNiTKo8bQlQg.PNG%2Fx25C4%25DA%25B7%25CE%25B3%25AA_%25BBx25E7%25C1%25F8.png&amp;type=b400" data-width="632" onerror="var we=$Element(this); we.addClass('bg_nimg'); we.attr('alt','이미지준비중'); we.attr('src','');" src=""/>,
 <img alt='억울한 코로나 맥주…코로나19 확산에 "구매의향 · 이미지↓" | 포토뉴스' class="_img" data-height="389" data-source="https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F001%2F2020%2F02%2F29%2FAKR20200229010000087_01_i_20200229062106385.jpg&amp;type=b400" data-width="500" onerror="var we=$Element(this); we.addClass('bg_nimg'); we.attr('alt','이미지준비중'); we.attr('src','');" src=""/>,
 <img alt="일식과 아름다운 태양 코로나 | 블로그" class="_img" data-height="774" data-source="https://search.pstatic.net/common/?src=http%3A%2F%2Fblogfiles.naver.net%2F20100126_285%2Fhuheun05_1264511922364 7k6sC_jpg%2F%25C0%25CF%25BD%25C4%25B0%25FA_%25C4%25DA%25B7%25CE%25B3%25AA_huheun05.jpg&amp;type=b400" data-width="520" onerror="var we=$Element(this); we.addClass('bg_nimg'); we.attr('alt','이미지준비중'); we.attr('src','');" src="
```

find_all이라는 명령어를 통해 해당하는 부분을 모두 가져온다. 이 find_all은 리스트 형태 값을 하나씩 가져온다.

이 중 data-source 부분이 해당 이미지의 url인데 아래와 같은 방식으로 이 부분에 들어있는 값을 가져온다.

```
 <img alt="일식과 아름다운 태양 코로나 | 블로그" class="_img" data-height="774" data-source="https://search.pstatic.net/common/?src=http%3A%2F%2Fblogfiles.naver.net%2F20100126_285%2Fhuheun05_1264511922364 7k6sC_jpg%2F%25C0%25CF%25BD%25C4%25B0%25FA_%25C4%25DA%25B7%25CE%25B3%25AA_huheun05.jpg&amp;type=b400" data-width="520" onerror="var we=$Element(this); we.addClass('bg_nimg'); we.attr('alt','이미지준비중'); we.attr('src','');" src="
```

아래와 같이 리스트를 for문으로 돌려 각각의 이미지 값을 확인해본다. 여기서 ['data-source']라는 것을 지정해준다. 여기서는 BeutifulSoup의 기능이 작동해 해당 부분을 가져오게 끔 한다.

In [9]:
```
cnt = 1
for i in img:
    imgUrl = i['data-source']
    print(imgUrl)
```

```
https://search.pstatic.net/common/?src=http%3A%2F%2Fpost.phinf.naver.net%2FMjAyMDA0MTNfMTgz%2FMDAxNTg2NzM1NzM0ODM4.petATgz0Xbpnp4_S35joz2eVvLy26VX8yc59Qx0p0P8g.f83b-xT0uvjclLd7wG6snCbueh7IrUFC855wBU3GWIQg.JPEG%2FIGthA5vkVP6vuDEzCQSnk3Ww3cJQ.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fcafefiles.naver.net%2FMjAyMDAyMjJf0TQg%2FMDAxNTgyMzUONTg5MzMz.9maUHUY9Uw4aqeCg-tnsCShALxfcx9GnKm5qg7oa3Fgg.jzVlcswHYV1mPC_NrmNXmVE2N_2j4nVQgNiTKo8bQlQg.PNG%2F%25C4%25DA%25B7%25CE%25B3%25AA_%25BB%25E7%25C1%25F8.png&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F001%2F2020%2F02%2F29%2FAKR20200229010000087_01_i_20200229062106385.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fblogfiles.naver.net%2F20100126_285%2Fhuheun05_126451192236 47k6sC_jpg%2F%25C0%25CF%25BD%25C4%25B0%25FA_%25C4%25DA%25B7%25CE%25B3%25AA_huheun05.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F081%2F2020%2F01%2F30%2F0003062054_001_20200130114108483.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F584%2F2020%2F03%2F05%2F0000007903_001_20200305162019868.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F5472%2F2020%2F02%2F27%2F0000046762_001_20200227000626470.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F5022%2F2020%2F02%2F21%2F0000273802_001_20200221134039400.jpg&type=b400
https://search.pstatic.net/common/?src=http%3A%2F%2Fimgnews.naver.net%2Fimage%2F5461%2F2020%2F04%2F15%2F0000011789_001_20200415091442807.jpg&type=b400
```

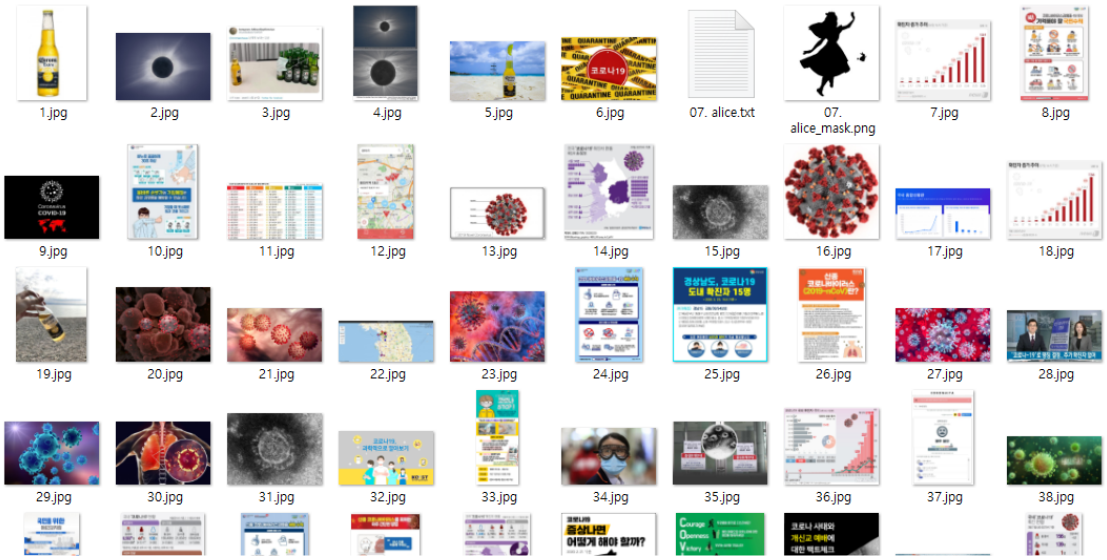
이렇게 이미지가 나오는 걸 확인하고 이제 이 url을 통해 이미지를 스크래핑해서 넣자.

In [13]:
```
cnt = 1
for i in img:
    imgUrl = i['data-source']
#     print(imgUrl)
    with urlopen(imgUrl) as file: # 해당 이미지 파일을 저장하기 위한 로직이다.
        with open('../image/'+str(cnt)+'.jpg','wb')as imgFile: # 해당 파일을 해당 경로에 이름을 만들어 저장한다.wb는 write binary 라는 뜻이
            img = file.read() # 이미지 파일을 읽고
            imgFile.write(img) # 읽은 파일을 write로 저장한다.
    cnt+=1 # 각 이름을 바꾸기 위해 cnt+1 을 한다.
    if cnt>imgCnt: # 우리가 처음 지정했던 50개가 넘어가면 멈추게 break로 멈추게 한다.
        break
print('Image download success')
```

```
Image download success
```

주의할 점은 경로 부분에 image라는 폴더가 있어야 된다. 해당 경로는 꼭 자기가 원하는 곳으로 지정해주자.

이걸 실행하면 아래와 같이 해당 폴더에 image가 저장되는 것을 볼 수 있다.



'**Python**' 카테고리의 다른 글

[Python] python 에서 Seleium을 설치 방법

[Python] BeautifulSoup을 통한 이미지 블로그 스크래핑하기

[Python] BeautifulSoup을 통한 이미지 스크래핑 하기

[python] 영화 리뷰에 대한 자연어 처리분석/ 감성분석하기 feat. 스크래핑

[python] BeautifulSoup를 통한 영화리뷰 scraping 하기

[Python] 파이썬 기초 14 - 아주 기초적인 pandas 사용법과 예제

BeutifulSoup 이미지 가져오기 python 이미지 가져오기 python 이미지 스크래핑 python 이미지 크롤링

나무늘보스

혼자 끄적끄적하는 블로그 입니다.