# Chap 5. 다양한 댓글 및 리뷰 정보 수집하기

## 1. 이번 장에서 배울 내용 소개

이번 장에서는 다양한 신문 기사나 SNS에 쓴 글 아래에 작성되는 댓글이나 리뷰 정보를 수집하는 내용을 살펴보겠습니다. 다양한 마케팅이나 고객들의 생각을 알기 위해서 현업에서도 아주 많이 사용하는 방법이므로 이번장의 내용을 열심히 학습하시고 연습문제까지 스스로의 힘으로 꼭 풀어 보시기 바랍니다.

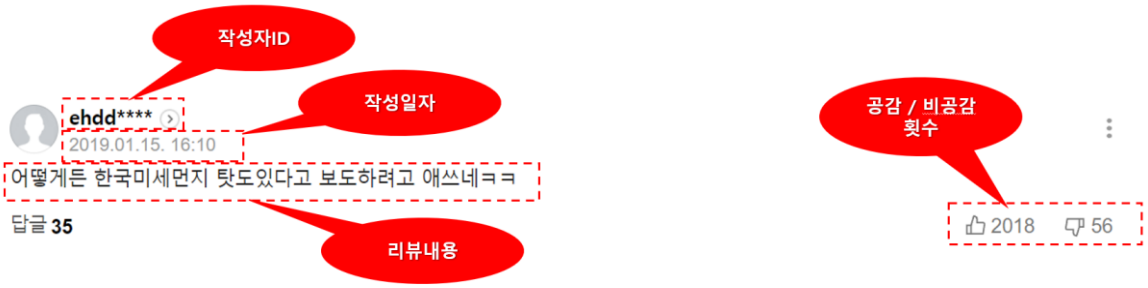
이번 시간에 예제로 사용할 내용은 인터넷 신문 기사에 작성된 댓글을 수집하는 것입니다.

**학습에 사용할 URL 주소:**

**https://news.naver.com/main/read.nhn?mode=LSD&mid=shm&sid1=102&oid=056&aid=0010661268**


위 그림과 같이 미세 먼지 관련 신문 기사에 달린 댓글을 수집하겠습니다.

앞의 그림을 보면 총 댓글이 504 건이 있는데 아래와 같은 형식으로 되어 있습니다.




위 그림과 같이 리뷰가 작성되는데 여기서 우리가 수집할 내용을 정리하면 아래와 같습니다.

1. 리뷰 작성자
2. 리뷰 내용
3. 리뷰 작성 일자
4. 공감 횟수
5. 비공감 횟수

**[ 웹 크롤러를 실행한 화면 예시 ]**

```
================================================================================
 Chap 16.뉴스 기사의 댓글 정보  수집하기
================================================================================

1.댓글을 크롤링할 뉴스의 URL을 입력하세요: https://news.naver.com/main/read.nhn?mode=LSD&mid=shm&sid1=102&oid=056&aid=0010661268
2.크롤링 할 건수는 몇건입니까?(10건단위로 입력): 30
3.파일을 저장할 폴더명만 쓰세요(예:c:₩py_temp₩):
================================================================================
```

위 그림과 같이 크롤링 할 뉴스의 URL 과 크롤링 할 건수와 저장할 폴더명을 입력 받아서 크롤링을 진행하면 됩니다.

**[ 크롤링 되는 화면 예시 ]**

```
전체 검색 결과 건수 : 505 건
실제 최종 출력 건수 30
실제 출력될 최종 페이지수 2


1 번째 댓글 수집 중 ==================
1.작성자ID: ehdd****
2.리뷰: 어떻게든 한국미세먼지 탓도있다고 보도하려고 애쓰네ㅋㅋ
3.작성일자: 2019.01.15. 16:10
4.공감: 2018
5.비공감: 56
```

**[ xls 와 csv 형식으로 저장된 화면 예시 ]**

```
================================================================================
1.요청된 총 30 건의 리뷰 중에서 실제 크롤링 된 리뷰수는 30 건입니다
2.총 소요시간은 29.8 초 입니다
3.파일 저장 완료: txt 파일명 : c:₩py_temp2022-03-21-10-55-34-뉴스기사댓글₩2022-03-21-10-55-34-뉴스기사댓글.txt
4.파일 저장 완료: csv 파일명 : c:₩py_temp2022-03-21-10-55-34-뉴스기사댓글₩2022-03-21-10-55-34-뉴스기사댓글.csv
5.파일 저장 완료: xls 파일명 : c:₩py_temp2022-03-21-10-55-34-뉴스기사댓글₩2022-03-21-10-55-34-뉴스기사댓글.xls
================================================================================
```

| | A | B | C | D | E | F |
|---|---|---|---|---|---|---|
| 1 | | 작성자ID | 리뷰내용 | 작성일자 | 공감횟수 | 비공감횟수 |
| 2 | 0 | ehdd**** | 어떻 | =2019.01.15. 16:10 | 2018 | 56 |
| 3 | 1 | siho**** | 진짜 | 가 2019.01.15. 16:26 | 725 | 37 |
| 4 | 2 | reds**** | 충남 | 2019.01.15. 16:38 | 519 | 4 |
| 5 | 3 | daup**** | 작성 | 2019.12.05. 22:23 | 0 | 0 |
| 6 | 4 | samk**** | 애쓴 | 2019.01.15. 16:55 | 137 | 7 |
| 7 | 5 | mark**** | 기자 | 충 2019.01.15. 17:27 | 71 | 3 |
| 8 | 6 | dusc**** | 중국 | 2019.01.15. 17:28 | 61 | 2 |
| 9 | 7 | hubb**** | 작성 | 2020.03.12. 06:08 | 0 | 0 |
| 10 | 8 | seun**** | 작성 | 2019.04.08. 11:04 | 0 | 0 |
| 11 | 9 | kim7**** | 작성 | 2020.12.14. 16:01 | 0 | 0 |
| 12 | 10 | rua0**** | 이정 | 에 2019.01.15. 17:25 | 28 | 1 |
| 13 | 11 | jihy**** | 충남 | 2019.01.15. 17:22 | 24 | 0 |
| 14 | 12 | summ**** | 작성 | 2019.07.10. 10:59 | 0 | 0 |
| 15 | 13 | slsl**** | 중국 | 2019.01.15. 17:28 | 11 | 0 |
| 16 | 14 | jhak**** | 작성 | 2019.01.15. 21:17 | 0 | 0 |
| 17 | 15 | blas**** | 유독 | 2019.01.15. 18:33 | 8 | 0 |
| 18 | 16 | ziwh**** | KBS | 2019.01.15. 17:32 | 8 | 0 |
| 19 | 17 | gago**** | 단순 | 2019.01.15. 17:12 | 8 | 2 |
| 20 | 18 | hyun**** | 진짜 | 2019.01.15. 18:03 | 6 | 0 |
| 21 | 19 | dlcn**** | 역시 | 2019.01.15. 17:32 | 6 | 0 |
| 22 | 20 | bern**** | 대통 | 2019.01.16. 09:58 | 5 | 0 |
| 23 | 21 | sush**** | Kbs | 2019.01.15. 21:36 | 5 | 0 |
| 24 | 22 | grea**** | 중국 | 2019.01.15. 20:25 | 5 | 0 |
| 25 | 23 | cup9**** | 작성 | 2019.01.15. 22:27 | 0 | 0 |
| 26 | 24 | yein**** | 작성 | 2020.03.19. 22:31 | 0 | 0 |
| 27 | 25 | wass**** | 충남 | 2019.01.15. 17:24 | 5 | 0 |

개인 정보 보호를 위해
이 부분은 일부러 숨겼습니다.

## 2. 전체 소스 코드 미리 보기

```
1   # Chap 16. 다양한 댓글 모으기
2   # 뉴스 기사의 댓글 모으기 - 미세먼지 / 스모그
3   # 테스트 기사 URL :
4 #https://news.naver.com/main/read.nhn?mode=LSD&mid=shm&sid1=102&oid=056&
aid=0010661268
5
6   #Step 1. 필요한 모듈과 라이브러리를 로딩합니다.
7
8   from bs4 import BeautifulSoup
9   from selenium import webdriver
10  from selenium.webdriver.common.by import By
11  from selenium.webdriver.common.keys import Keys
12  from selenium.webdriver.chrome.service import Service
13  import time
14  import math
15  import numpy
16  import pandas as pd
17  import random
18  import os
19  import re
20
21  #Step 2. 사용자에게 검색어 키워드를 입력 받고 저장할 폴더와 파일명을 설정합니다.
22  print("=" *80)
23  print(" Chap 16.뉴스 기사의 댓글 정보  수집하기")
24  print("=" *80)
25  print("₩n")
26
27  query_txt = '뉴스기사댓글'
28  query_url = input('1.댓글을 크롤링할 뉴스의 URL을 입력하세요: ')
29  cnt = int(input('2.크롤링 할 건수는 몇건입니까?(10건단위로 입력): '))
30  page_cnt = math.ceil(cnt / 20)
31
32  f_dir = input("3.파일을 저장할 폴더명만 쓰세요(예:c:₩₩py_temp₩₩):")
33  if f_dir=='' :
34      f_dir='c:₩₩py_temp'
35
36  # 저장될 파일위치와 이름을 지정합니다
```

```
37  n = time.localtime()
38  s = '%04d-%02d-%02d-%02d-%02d-%02d' % (n.tm_year, n.tm_mon, n.tm_mday, n.tm_hour, n.tm_min, n.tm_sec)
39
40  os.makedirs(f_dir+s+'-'+query_txt)
41  os.chdir(f_dir+s+'-'+query_txt)
42
43  ff_name=f_dir+s+'-'+query_txt+'₩₩'+s+'-'+query_txt+'.txt'
44  fc_name=f_dir+s+'-'+query_txt+'₩₩'+s+'-'+query_txt+'.csv'
45  fx_name=f_dir+s+'-'+query_txt+'₩₩'+s+'-'+query_txt+'.xls'
46
47  #Step 3. 크롬 드라이버를 사용해서 웹 브라우저를 실행합니다.
48
49  s_time = time.time( )
50
51  s = Service("c:/py_temp/chromedriver.exe")
52  driver = webdriver.Chrome(service=s)
53
54  driver.get(query_url)
55  driver.maximize_window()
56  time.sleep(5)
57
58  #Step 4. 현재 총 리뷰 건수를 확인하여 사용자의 요청건수와 비교 후 동기화합니다
59  html = driver.page_source
60  soup = BeautifulSoup(html, 'html.parser')
61
62  result= soup.find('div', class_='u_cbox_head').find('span','u_cbox_count')
63  result2 = result.get_text()
64
65  print("=" *80)
66  result3 = result2.replace(",","")
67  result4 = re.search("₩d+",result3)
68  search_cnt = int(result4.group())
69
70  if cnt > search_cnt :
71      cnt = search_cnt
72
73  print("전체 검색 결과 건수 :",search_cnt,"건")
74  print("실제 최종 출력 건수",cnt)
75  print("실체 출력될 최종 페이지수" , page_cnt)
```

```
76
77  # Step 5. 사용자가 요청한 건수가 많을 경우 리뷰 더보기 버튼을 클릭합니다
78  # 최초 10건 수집후 댓글 더보기 버튼 클릭
79  # 아래 버튼을 눌러 첫 화면에 총 20건의 댓글이 나오게 만듦
80  driver.find_element(By.XPATH,'//*[@id="cbox_module"]/div/div[9]/a/span[1]').click()
81  time.sleep(3)
82
83  #Step 6. 리뷰와 점수 등 내용 수집
84  writer_id2=[]          # 리뷰 작성자 ID
85  review2=[]              # 리뷰 내용
86  write_date2=[]         # 리뷰 작성 일자
87  gogam_0=[]              # 공감 횟수
88  gogam_1=[]              # 비공감 횟수
89  count = 0
90
91  for a in range(1,page_cnt+1) :
92
93      if a == page_cnt :
94          break
95      else :
96          driver.find_element(By.XPATH,'//*[@id="cbox_module"]/div/div[9]/a').click()
97          time.sleep(3)
98          print("%s페이지 이동 완료============================" %a)
99          time.sleep(random.randrange(1,3))  # 3-8 초 사이에 랜덤으로 시간 선택
100
101 print('이제 리뷰 정보를 수집합니다. 잠시만 기다려 주세요~~~~~~~~')
102
103 #txt 파일에 저장하기 위해 파일 open하기
104 f = open(ff_name, 'a',encoding='UTF-8')
105
106 html = driver.page_source
107 soup = BeautifulSoup(html, 'html.parser')
108
109 reple_result = soup.find('div', class_='u_cbox_content_wrap').find('ul')
110 slist = reple_result.find_all('li')
111
112 for li in slist:
113     count += 1
114     print("₩n")
```

```
115     print("총 %s건 중 %s번째 댓글 수집 중입니다 ==========" %(cnt,count))
116
117     writer_id = li.find('span', class_='u_cbox_nick').get_text()
118     print("1.작성자ID:", writer_id)
119     f.write("₩n")
120     f.write("총 %s 건 중 %s 번째 리뷰 데이터를 수집합니다====" %(cnt,count) + "₩n")
121     f.write("1.작성자ID:"+writer_id + "₩n")
122     writer_id2.append(writer_id)
123
124     try :
125         review = li.find('span', class_='u_cbox_contents').get_text()
126     except AttributeError :
127         review='작성자에 의해 삭제된 댓글입니다'
128         print("2.리뷰 :",review)
129     else :
130         print("2.리뷰:",review)
131     f.write("2.리뷰:" + review + "₩n")
132     review2.append(review)
133
134     write_date = li.find('span',class_='u_cbox_date').get_text()
135     print('3.작성일자:',write_date)
136     f.write("3.작성일자:" + write_date + "₩n")
137     write_date2.append(write_date)
138
139     gogam = li.find('div', class_='u_cbox_recomm_set').find_all('em')
140
141     try :
142         g_gogam = gogam[0].text
143         print('4.공감:',g_gogam)
144     except IndexError :
145         g_gogam = '0'
146         print('4.공감 :',g_gogam)
147     f.write("4.공감:" + g_gogam + "₩n")
148     gogam_0.append(g_gogam)
149
150     gogam = li.find('div', class_='u_cbox_recomm_set').find_all('em')
151
152     try :
153         b_gogam = gogam[1].text
```

```
154             print('5.비공감:',b_gogam)
155         except IndexError :
156             b_gogam = '0'
157             print('5.비공감 :',b_gogam)
158         f.write("5.비공감:" + b_gogam + "₩n")
159         gogam_1.append(b_gogam)
160
161         time.sleep(0.2)
162
163         if count == cnt :
164             break
165
166 #Step 7. xls 형태와 csv 형태로 저장하기
167 news_reple = pd.DataFrame()
168 news_reple['작성자ID']=pd.Series(writer_id2)
169 news_reple['리뷰내용']=pd.Series(review2)
170 news_reple['작성일자']=pd.Series(write_date2)
171 news_reple['공감횟수']=pd.Series(gogam_0)
172 news_reple['비공감횟수']=pd.Series(gogam_1)
173
174 # csv 형태로 저장하기
175 news_reple.to_csv(fc_name,encoding="utf-8-sig",index=True)
176
177 # 엑셀 형태로 저장하기
178 news_reple.to_excel(fx_name ,index=True , engine='openpyxl')
179
180 # Step 8. 요약 정보 출력하기
181 e_time = time.time( )
182 t_time = e_time - s_time
182 print("₩n")
183 print("=" *120)
184 print("1.요청된 총 %s 건의 리뷰 중에서 실제 크롤링 된 리뷰수는 %s 건입니다" %(cnt,count))
185 print("2.총 소요시간은 %s 초 입니다 " %round(t_time,1))
186 print("3.파일 저장 완료: txt 파일명 : %s " %ff_name)
187 print("4.파일 저장 완료: csv 파일명 : %s " %fc_name)
188 print("5.파일 저장 완료: xls 파일명 : %s " %fx_name)
189print("=" *120)
190 driver.close( )
```

**********

## 3. 주요 소스코드 설명

소스코드의 내용이 길고 앞부분은 이전 챕터의 내용과 중복되는 부분이 많아서 설명을 생략하고 이번 챕터에서 중요한 부분들 위주로 발췌하여 설명하겠습니다.
( 이 챕터의 전체 코드는 저자가 제공하는 소스코드를 참고해주세요)

```
58  #Step 4. 현재 총 리뷰 건수를 확인하여 사용자의 요청건수와 비교 후 동기화합니다
59  html = driver.page_source
60  soup = BeautifulSoup(html, 'html.parser')
61
62  result= soup.find('div', class_='u_cbox_head').find('span','u_cbox_count')
63  result2 = result.get_text()
64
65  print("=" *80)
66  result3 = result2.replace(",","")
67  result4 = re.search("₩d+",result3)
68  search_cnt = int(result4.group())
69
70  if cnt > search_cnt :
71        cnt = search_cnt
72
73  print("전체 검색 결과 건수 :",search_cnt,"건")
74  print("실제 최종 출력 건수",cnt)
75  print("실체 출력될 최종 페이지수" , page_cnt)
```

위 코드의 62-63번 행에서 현재 전체 리뷰수를 가져옵니다.
그리고 66-68번 행까지 정규식을 이용하여 숫자 부분만 추출합니다.
70-71번 행에서 만약 사용자가 요청한 건수(cnt)와 실제 리뷰수(search_cnt) 값 중에서 사용자가 요청한 건수가 더 많다면 실제 리뷰수로 통일합니다. 즉 사용자가 100건의 리뷰 수집을 요청했으나 만약 실제 리뷰가 50건 밖에 없었다면 사용자가 비록 100건을 요청했다고 할지라도 크롤링할 실제 수는 50건으로 지정한다는 의미입니다.
네이버 뉴스 댓글의 경우 맨 처음 기사 아래에 기본적으로 10건의 리뷰가 보이고 댓글 더보기 버튼을 클릭하면 추가로 10건이 더 보입니다.

그래서 소스코드 본문의 80번 행에서 xpath 값으로 댓글 더보기 버튼을 클릭합니다.

```
77  # Step 5. 사용자가 요청한 건수가 많을 경우 리뷰 더보기 버튼을 클릭합니다
78  # 최초 10건 수집후 댓글 더보기 버튼 클릭
79  # 아래 버튼을 눌러 첫 화면에 총 20건의 댓글이 나오게 만듦
80  driver.find_element(By.XPATH,'//*[@id="cbox_module"]/div/div[9]/a/span[1]').click()
81  time.sleep(3)
```

참고로 위 80번행의 댓글 더보기 버튼에 대한 xpath 값은 변경될 수 있으니 실습하는 지금 이 시점의 xpath 값을 확인해서 사용하세요.

그리고 한 페이지에 20개의 댓글이 보이고 추가로 더 보고 싶으면 화면 맨 아래의 더보기 버튼을 클릭하면 20건씩 추가로 내용이 보입니다.
즉 1 페이지에 20건씩 리뷰가 보이는 것입니다.
그래서 아래와 같이 사용자가 요청한 건수를 20으로 나누어서 페이지 번호를 계산하고 현재 페이지 번호가 전체 페이지 번호와 같을 때까지 계속 더보기 버튼을 클릭하도록 93-99번 행까지 코드를 작성하였습니다.

```
91  for a in range(1,page_cnt+1) :
92
93      if a == page_cnt :
94          break
95      else :
96          driver.find_element(By.XPATH,'//*[@id="cbox_module"]/div/div[9]/a').click()
97          time.sleep(3)
98          print("%s페이지 이동 완료============================" %a)
99          time.sleep(random.randrange(1,3))  # 1-3 초 사이에 랜덤으로 시간 선택
```

위 코드의 96번 행이 화면 맨 아래의 더보기 버튼의 xpath 값을 지정하여 클릭하는 부분입니다.
그리고 99번행은 다음 페이지로 넘어갈 때 까지 1초에서 3초 사이에 랜덤한 시간만큼 기다리라는 의미입니다.

네이버 뉴스 리뷰의 경우에는 댓글 더보기 버튼을 여러차례 클릭한 후 한꺼번에 댓글을 다 가져올 수 있습니다.
그래서 한꺼번에 댓글 더보기 버튼을 수차례 클릭한 후 댓글을 한꺼번에 수집하면 됩니다.

```
103 #txt 파일에 저장하기 위해 파일 open하기
104 f = open(ff_name, 'a',encoding='UTF-8')
105
106 html = driver.page_source
107 soup = BeautifulSoup(html, 'html.parser')
108
109 reple_result = soup.find('div', class_='u_cbox_content_wrap').find('ul')
110 slist = reple_result.find_all('li')
111
```

위 코드의 109-110번 행에서 현재 페이지에 있는 리뷰 목록을 전부 다 가져와서 slist 변수에 저장합니다. 이제 아래와 같이 반복문을 활용하여 하나의 댓글마다 원하는 정보를 추출하면 됩니다.

그런데 리뷰 중에서 작성자가 삭제한 리뷰들이 있는데 이 경우를 잘 처리해야 합니다.
아래 코드를 보세요.

```
124     try :
125         review = li.find('span', class_='u_cbox_contents').get_text()
126     except AttributeError :
127         review='작성자에 의해 삭제된 댓글입니다'
128         print("2.리뷰 :",review)
129     else :
130         print("2.리뷰:",review)
131     f.write("2.리뷰:" + review + "₩n")
132     review2.append(review)
```

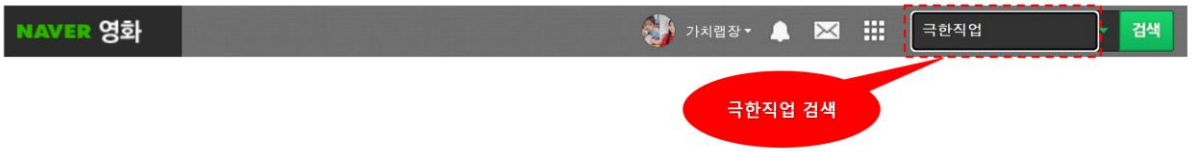
위 코드의 125번 행에서 리뷰 내용 정보를 추출하는데  작성자에 의해 삭제된 댓글일 경우는 해당 값이 없습니다. 이럴 경우 위 코드와 같이 예외처리를 사용해서 처리했습니다.

나머지 코드들은 이전 챕터에서 모두 보았던 내용들이 반복적으로 나오는 것이라서 설명은 생략하겠습니다.

## 4. 연습 문제로 실력 굳히기

### 1) 네이버 영화 리뷰 및 평점 수집하기

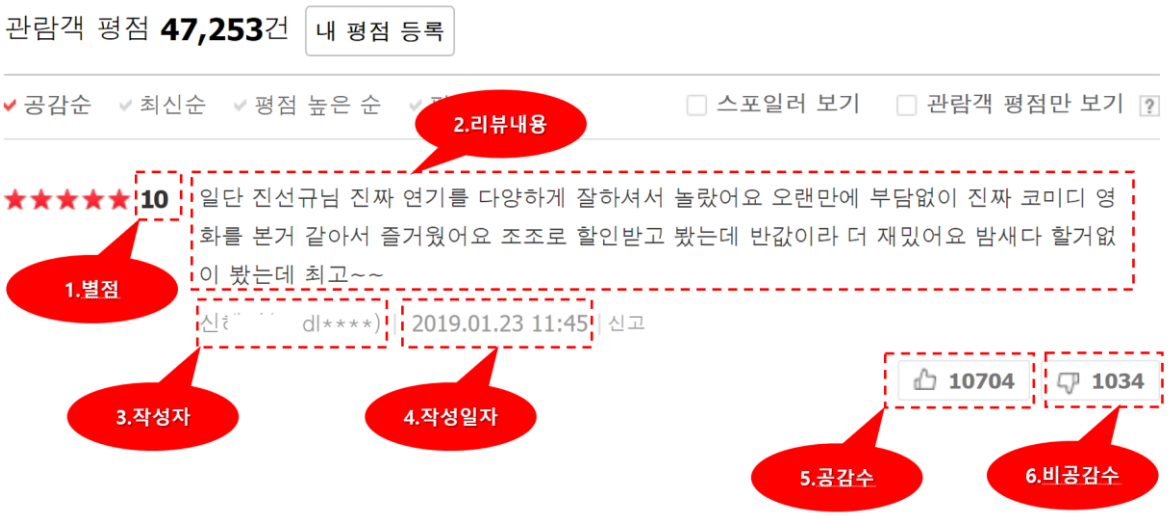
아래의 내용을 참고하여 네이버에서 제공하는 영화 평점 사이트( https://movie.naver.com) 에서 특정 영화를 검색 한 후 해당 영화의 다양한 정보를 수집하는 크롤러를 만드세요.




위 그림에서 오른쪽 상단의 영화 검색 부분에 영화 제목을 입력하여 검색하면 해당 영화가 나옵니다. 그 영화를 선택하면 아래 그림과 같이 상세 정보가 출력되는데 그 중에서 평점 메뉴를 누르세요


위 그림에서 평점 메뉴를 누르면 다양한 정보가 나오는데 아래 그림과 같이 해당 영화의 댓글이 출력됩니다.

아래 그림의 내용을 수집하여 결과를 예시화면과 같이 txt , csv , xls 파일로 저장하세요.



[ 크롤러 실행 화면 예시 화면 ]

```
================================================================================
연습문제 :네이버 영화 리뷰 정보 수집하기
================================================================================
1.크롤링 할 영화의 제목을 입력하세요: 극한직업
2.크롤링 할 리뷰건수는 몇건입니까?: 30
3.파일을 저장할 폴더명만 쓰세요(예:c:₩py_temp₩):
--------------------------------------------------------------------------------
요청하신 데이터를 수집 중이오니 잠시만 기다려 주세요~~~~
================================================================================
전체 검색 결과 건수 : 47253 건
실제 최종 출력 건수 30


크롤링 할 총 페이지 번호:  3
================================================================================
```

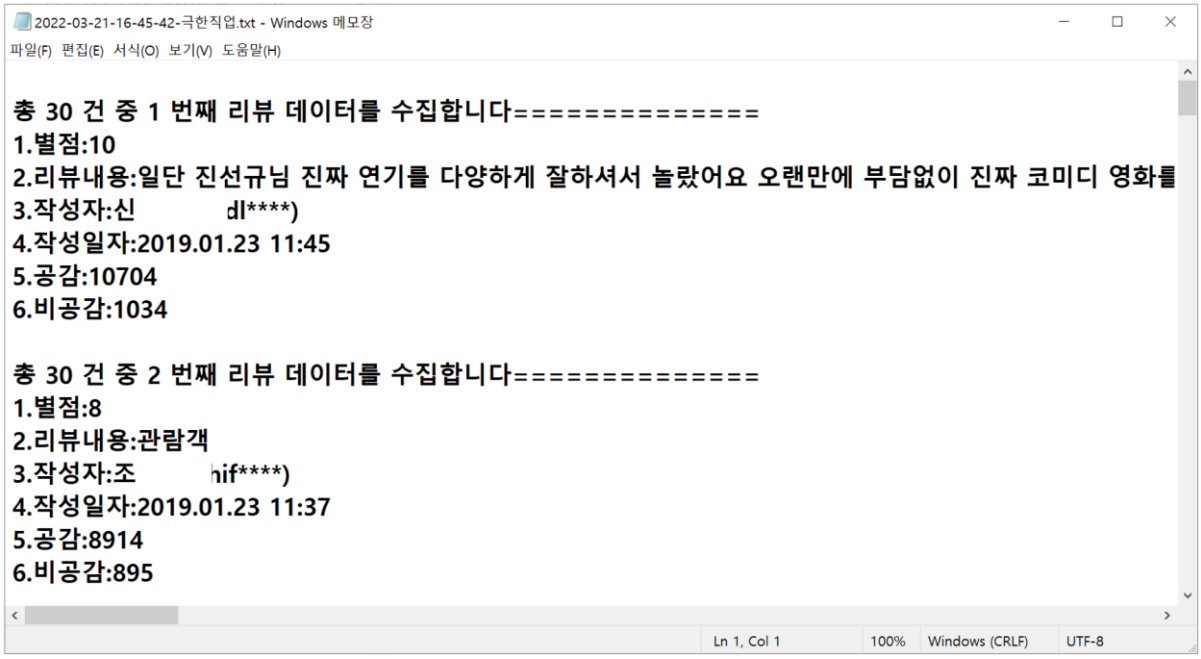
위 그림처럼 1. 영화제목 / 2.크롤링 건수 / 3. 저장폴더명 을 입력 받아서 크롤링 하세요.
전체 리뷰수와 페이지 번호까지 보여주면 더 좋겠죠?
아래 그림과 같이 출력하면 됩니다.

```
총 30 건 중 1 번째 리뷰 데이터를 수집합니다==============
1.별점: ********** :  10
2.리뷰내용: 일단 진선규님 진짜 연기를 다양하게 잘하셔서 놀랐어요 오랜만에 부담없이
할인받고 봤는데 반값이라 더 재밌어요 밤새다 할거없이 봤는데 최고~~
3.작성자: 신        ›dl****)
4.작성일자: 2019.01.23 11:45
5.공감: 10704
6.비공감: 1034
```

( 총 30건을 출력한 후 아래와 같이 결과를 보여주세요~ )

```
================================================================================
1.요청된 총 30 건의 리뷰 중에서 실제 크롤링 된 리뷰수는 30 건입니다
2.총 소요시간은 18.1 초 입니다
3.파일 저장 완료: txt 파일명 : c:₩py_temp₩2022-03-21-16-45-42-극한직업₩2022-03-21-16-45-42-극한직업.txt
4.파일 저장 완료: csv 파일명 : c:₩py_temp₩2022-03-21-16-45-42-극한직업₩2022-03-21-16-45-42-극한직업.csv
5.파일 저장 완료: xls 파일명 : c:₩py_temp₩2022-03-21-16-45-42-극한직업₩2022-03-21-16-45-42-극한직업.xls
================================================================================
```

[ txt 형식으로 저장된 예시 화면 ]




[ xls 와 csv 형식으로 저장된 예시 화면 ]

| | A | B | C | D | E | F | G |
|---|---|---|---|---|---|---|---|
| 1 | | 별점(펑점) | 리뷰내용 | 작성자 | 작성일자 | 공감횟수 | 비공감횟수 |
| 2 | 0 | 10 | 일단 | 신 | 2019.01.23 11:45 | 10704 | 1034 |
| 3 | 1 | 8 | 관람 | 조 | 2019.01.23 11:37 | 8914 | 895 |
| 4 | 2 | 10 | 관람 | 냐 | 2019.01.23 12:47 | 9083 | 1302 |
| 5 | 3 | 10 | 관람 | 아 | 2019.01.23 14:48 | 7323 | 753 |
| 6 | 4 | 10 | 지금 | 박 | 2019.01.23 20:16 | 7003 | 639 |
| 7 | 5 | 10 | 간만 | 11 | 2019.01.23 14:47 | 6123 | 644 |
| 8 | 6 | 8 | 아주 | ST | 2019.01.23 18:00 | 5350 | 536 |
| 9 | 7 | 10 | 류승 | ks | 2019.01.23 17:32 | 4589 | 427 |
| 10 | 8 | 10 | 마약 | 박 | 2019.01.23 09:26 | 4518 | 523 |
| 11 | 9 | 10 | 관람 | 샤 | 2019.01.23 12:48 | 3988 | 521 |
| 12 | 10 | 10 | 조조 | lo | 2019.01.23 11:42 | 3154 | 511 |
| 13 | 11 | 10 | 류승 | ne | 2019.01.23 20:53 | 2850 | 250 |
| 14 | 12 | 9 | 관람 | 쏠 | 2019.01.23 11:42 | 2580 | 339 |
| 15 | 13 | 10 | ㅎㅎ | 옉 | 2019.01.23 13:22 | 2419 | 347 |
| 16 | 14 | 10 | 진선 | 진 | 2019.01.23 21:01 | 2175 | 222 |
| 17 | 15 | 10 | 관람 | 생 | 2019.01.23 13:02 | 2293 | 361 |
| 18 | 16 | 7 | 방금 | kik | 2019.01.23 11:09 | 2333 | 457 |
| 19 | 17 | 5 | 유머 | 쏭 | 2019.01.27 17:23 | 2530 | 770 |
| 20 | 18 | 6 | 글쎄.. | 빈자리(gusq | 2019.01.26 03:28 | 2694 | 1016 |

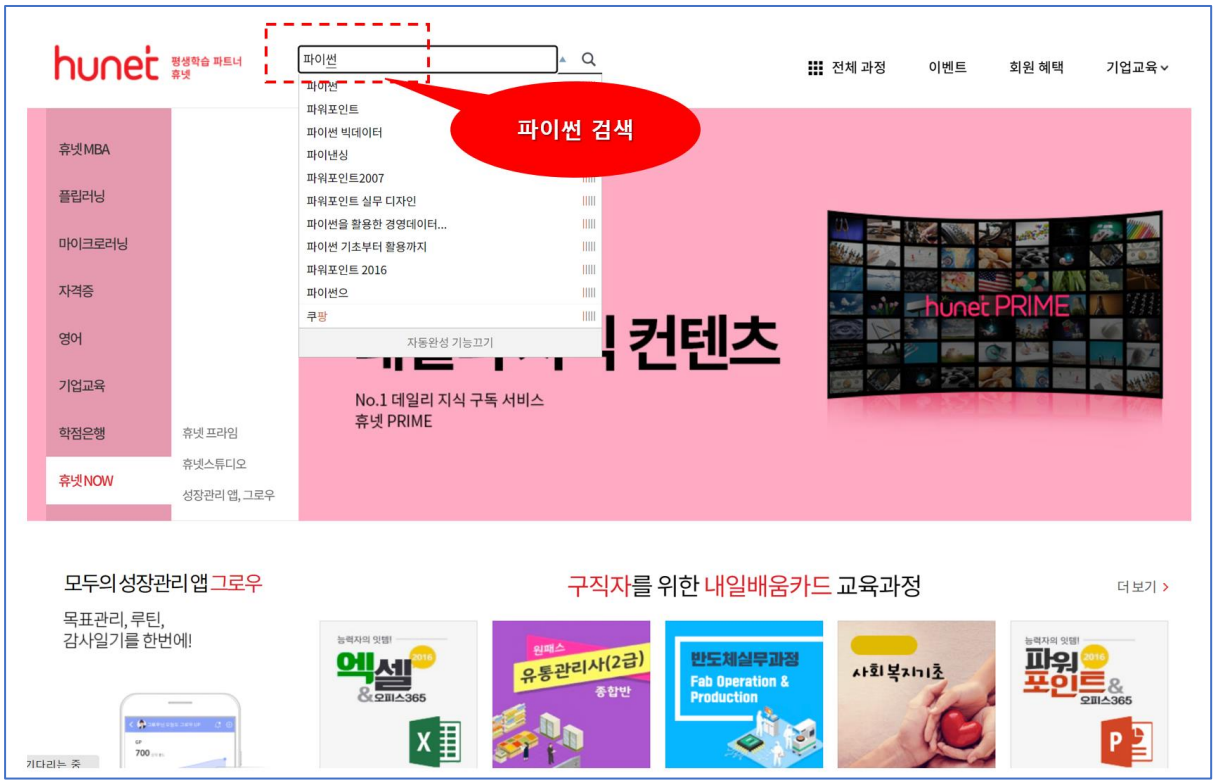
개인 정보 보호를 위해 이 부분은 일부러 숨겼습니다.

개인 정보 보호를 위해 이 부분은 일부러 숨겼습니다.

### 2) 강의 리뷰 정보 수집하기

이번 연습문제는 온라인 강의 분야에서 최고의 회사인 휴넷 사이트에서 특정 과목의 리뷰 정보를 수집하여 xls , csv 형식으로 저장하는 크롤러를 만들어 보는 것입니다.
아래 그림으로 자세히 설명하겠습니다.

먼저 휴넷 ( https://www.hunet.co.kr) 사이트에 접속을 한 후 "파이썬" 과목으로 과정을 검색하면 아래 그림과 같이 다양한 강의 과목과 정보들이 많이 있습니다.



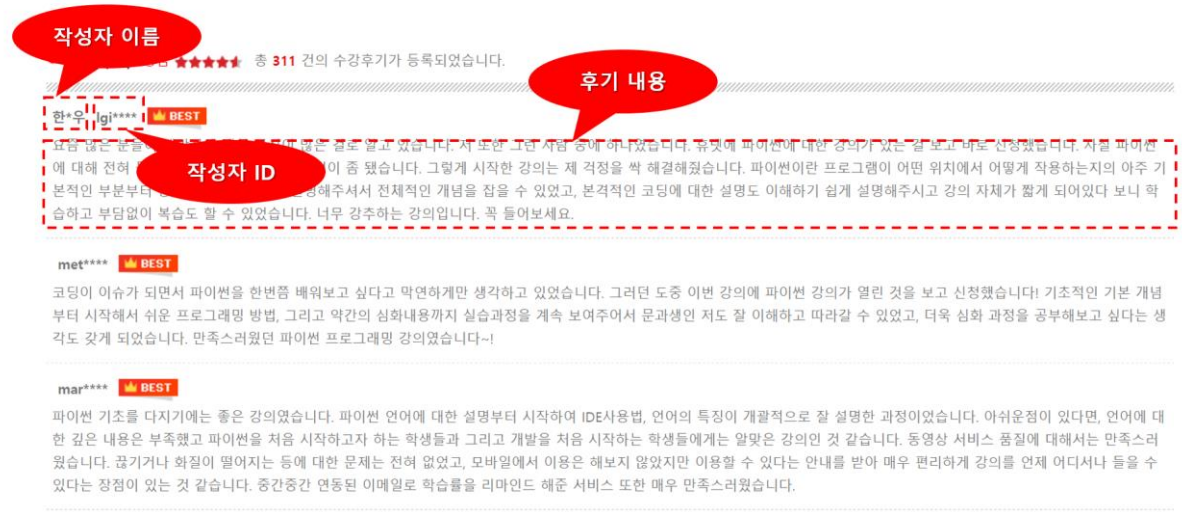


검색된 과정 중에서 리뷰가 많은 과목인 "내 생애 첫 프로그래밍 파이썬" 과목의 리뷰 정보를 수집할 예정입니다.

**수집할 강의 URL : https://hbs.hunet.co.kr/Education/Detail?gid=Y00079478**
코드를 작성할 때 url 주소는 위 강의 url 주소를 사용하세요.

위 강의 URL 주소로 접속하면 아래와 같이 해당 강의 상세 페이지가 나오는데 이 페이지의 오른쪽 Best 수강후기 아래의 더보기 버튼을 클릭하여 전체 리뷰 페이지로 접속하세요.


아래와 같이 리뷰 상세 페이지에서 리뷰 작성자 이름 / 작성자 ID / 리뷰 내용을 수집하여 xls , csv , txt 형식으로 저장하세요. 작성자 이름이 없는 리뷰도 많은데 이런 경우는 작성자 이름 부분에 ' 작성자 이름이 없습니다' 라는 문장으로 이름을 대신하여 저장하세요.



[ 웹 크롤러 실행 화면 예시 ]

```
================================================================================================
 연습문제: 이 크롤러는 휴넷 사이트의 강의 리뷰 수집용 웹크롤러입니다.
================================================================================================
1.수집할 리뷰는 총 몇건입니까?(기본값:10): 25

1 페이지의 리뷰 정보를 수집합니다======
1번째 리뷰 정보를 수집합니다=======
1.번호: 1
2.리뷰 작성자 이름: 한*우
3.리뷰 작성자 ID: lgi****
4.리뷰 내용: 요즘 많은 분들이 개발자에 대한 관심이 많은 걸로 알고 있습니다. 저 또한 그런 사람 중에 하나였습니다. 휴넷에 파이썬에 대한
강의가 있는 걸 보고 바로 신청했습니다. 사실 파이썬에 대해 전혀 몰라서 강의를 신청하고도 걱정이 좀 됐습니다. 그렇게 시작한 강의는 제 걱
정을 싹 해결해줬습니다. 파이썬이란 프로그램이 어떤 위치에서 어떻게 작용하는지의 아주 기본적인 부분부터 강사님께서 세세하게 설명해주셔
서 전체적인 개념을 잡을 수 있었고, 본격적인 코딩에 대한 설명도 이해하기 쉽게 설명해주시고 강의 자체가 짧게 되어있다 보니 학습하고 부담
없이 복습도 할 수 있었습니다. 너무 강추하는 강의입니다. 꼭 들어보세요.
```

[ xls 파일 저장 예시 ]

| | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 번호 | 작성자이름 | 작성자ID | 리뷰내용 | | | | | | | | | | | |
| 2 | 1 | 한*우 | lgi**** | 요즘 | | | | | | | | | | | |
| 3 | 2 | | met**** | 코딩 | | | | | | | | | | | |
| 4 | 3 | | mar**** | 파이 | | | | | | | | | | | |
| 5 | 4 | 조*현 | AMK**** | 왕초 | | | | | | | | | | | |
| 6 | 5 | 민*식 | int**** | 일반 | | | | | | | | | | | |
| 7 | 6 | 김*일 | eas**** | 이 강 | | | | | | | | | | | |
| 8 | 7 | 이*선 | nov**** | 한 달 | | | | | | | | | | | |
| 9 | 8 | 정*훈 | NT1**** | 요즘 | | | | | | | | | | | |
| 10 | 9 | 이*석 | lgc**** | 기초 | | | | | | | | | | | |
| 11 | 10 | 조*진 | gmb**** | 택배 | | | | | | | | | | | |
| 12 | 11 | 박*준 | kbs**** | 강의 | | | | | | | | | | | |
| 13 | 12 | 안*석 | KO2**** | 적절 | | | | | | | | | | | |
| 14 | 13 | 서*배 | lgi**** | 영상 | | | | | | | | | | | |
| 15 | 14 | 심*홍 | gse**** | 파이 | | | | | | | | | | | |
| 16 | 15 | 김*돈 | kim**** | 한달 | | | | | | | | | | | |
| 17 | 16 | 배*만 | nse**** | 따라 | | | | | | | | | | | |
| 18 | 17 | 최*원 | ncc**** | 파이 | | | | | | | | | | | |

개인 정보 보호를 위해
이 부분은 일부러 숨겼습니다.

### 3) 유튜브 영상의 댓글 수집하기

유투브(www.youtube.com)에서 특정 키워드로 영상을 검색한 후 사용자가 요청한 영상의 건수만큼 각 영상 별 댓글을 수집하여 아래의 예시 화면과 같이 txt , csv , xls 형식으로 저장하는 크롤러를 만드세요.

[ 크롤러 실행 화면 예시 ]

```
================================================================================
연습문제 : 유투브 영상의 댓글 수집하기
================================================================================
1.유튜브에서 검색할 주제 키워드를 입력하세요(예:올리브영): 올리브영
2.위 주제로 댓글을 크롤링할 유튜브 영상은 몇건입니까?:5
3.크롤링 결과를 저장할 폴더명만 쓰세요(예:c:₩py_temp₩):
```

위 그림과 같이 사용자에게 영상을 검색할 주제 키워드를 입력 받고 크롤링 할 영상의 건수를 입력 받은 후 각 영상별로 크롤링할 댓글을 해당 영상의 전체 리뷰수로 설정하고 파일을 저장할 폴더명을 입력 받은 후 아래의 항목들을 추출하세요.

```
1 번째 동영상의 정보를 수집합니다.
댓글 112개
첫번째 영상의 댓글은 답글 포함해서 총 112개입니다
================================================================================
1 번째 동영상의 조회수는 109671 회 이고 수집할 댓글은 총 112개 입니다
1 번째 동영상의 제목은 5년차 올리브영 알바생의 3월 올영세일에 꼭 사야할 올리브영 추천템 (광고X,내돈내산) 입니다
================================================================================
```

[ 크롤링 화면 예시 ]

```
1 번째 영상의 1 번째  댓글
--------------------------------------------------------------------------------
1.URL 주소: https://www.youtube.com//watch?v=p7RFMLgNNZM
2.댓글 작성자명: 짱                                  soon
3.댓글 작성일자: 2주 전
15:29 스킬 없이도 사용 가능! 만능 마사저 17:48 참고사항&영상마무리4:36 모든 피부 추천! 딥클렌징+순한 클렌징폼 6:29 나만 쓰기엔 너무 좋은 신상 마스크팩 8:42 아기 피부로 다시 태어나는 초고보습 바디크림 10:54 생리통 필수템.. 제발 써보시면 안될까요 13:15 끈적임 없고 촉촉한 '무향' 핸드크림(ft.대용량)
================================================================================
1번째 영상에서 최종 54건의 리뷰를 수집 완료했습니다~~
현재까지 1개의 영상에서 1건의 누적 리뷰건수를 수집했습니다
================================================================================


1 번째 영상의 2 번째  댓글
--------------------------------------------------------------------------------
1.URL 주소: https://www.youtube.com//watch?v=p7RFMLgNNZM
2.댓글 작성자명: 이
3.댓글 작성일자: 2주 전
4.댓글 내용: 저번 올리브영 추천템때도 도움을 많이 받았는데~ 짱순님이 차근차근 잘 설명해주셔서 너무 도움 많이 받았어요~! 감사해요
================================================================================
1번째 영상에서 최종 54건의 리뷰를 수집 완료했습니다~~
현재까지 1개의 영상에서 2건의 누적 리뷰건수를 수집했습니다
================================================================================
```

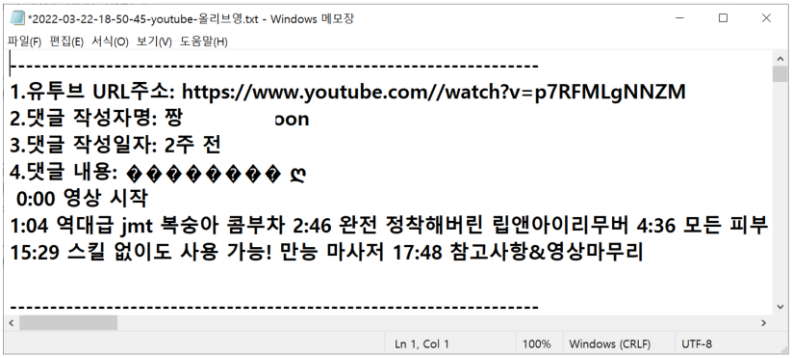
( 중간 내용은 생략합니다 )

```
5 번째 영상의 33 번째  댓글
--------------------------------------------------------------------
1.URL 주소: https://www.youtube.com//watch?v=o56gcd9FJ1k
2.댓글 작성자명: 선연의 채널
3.댓글 작성일자: 5일 전
4.댓글 내용: 안녕하세요 사랑해ෆ
====================================================================================================================
5번째 영상에서 최종 33건의 리뷰를 수집 완료했습니다~~
현재까지 5개의 영상에서 538건의 누적 리뷰건수를 수집했습니다
====================================================================================================================



====================================================================================================
1.요청된 총 5 건 동영상 리뷰 중에서 실제 크롤링 된 리뷰수는 538 건입니다
2.총 소요시간은 396.3 초 입니다
3.파일 저장 완료: txt 파일명 : c:₩py_temp₩2022-03-22-18-50-45-youtube-올리브영₩2022-03-22-18-50-45-youtube-올리브영.txt
4.파일 저장 완료: csv 파일명 : c:₩py_temp₩2022-03-22-18-50-45-youtube-올리브영₩2022-03-22-18-50-45-youtube-올리브영.csv
5.파일 저장 완료: xls 파일명 : c:₩py_temp₩2022-03-22-18-50-45-youtube-올리브영₩2022-03-22-18-50-45-youtube-올리브영.xls
====================================================================================================
```

[ txt 파일 저장 예시 화면 ]




[ csv , xls 파일 저장 예시 화면 ]


이번 챕터에서도 중요한 내용을 많이 공부했습니다.
이번 챕터에서 살펴본 내용 외에도 현업에서는 네이버 카페 등의 글과 리뷰 정보들도 많이 수집하는데 이런 것도 도전해보면 실력이 많이 늘겠죠?
열심히 연습해서 꼭 여러분들의 실력으로 만드세요~~~

[ 파이썬 능력자 너도 될 수 있어~!  - 서진수 저 - ]