네이버 댓글 정보 가져와보기

기본 이해

In [1]:

```
from bs4 import BeautifulSoup
```

In [2]:

```
html = """
<html>
<head><title> test site </title></head>
<p class='class1' align="left">test3</p>
<p class='class1'>test2</p>
<p id='p1'>오늘의 주가지수 1500</p>
<span class='class3'>span tag text</span>
<p class='class4'>test3</p>
</html>
"""
```

In [3]:

```
soup = BeautifulSoup(html, 'lxml')
```

In [4]:

```
print( soup.prettify() )
```

```
<html>
 <head>
  <title>
   test site
  </title>
 </head>
 <body>
  <p align="left" class="class1">
   test3
  </p>
  <p class="class1">
   test2
  </p>
  <p id="p1">
   오늘의 주가지수 1500
  </p>
  <span class="class3">
   span tag text
  </span>
  <p class="class4">
   test3
  </p>
 </body>
</html>
```

children개념 이해

- body 요소의 아이요소 가져오기

In [5]:

```
list(soup.children)
```

Out[5]:

```
[<html>
 <head><title> test site </title></head>
 <body><p align="left" class="class1">test3</p>
 <p class="class1">test2</p>
 <p id="p1">오늘의 주가지수 1500</p>
 <span class="class3">span tag text</span>
 <p class="class4">test3</p>
 </body></html>,
 '₩n']
```

In [6]:

```
list(soup.body.children)
```

Out[6]:

```
[<p align="left" class="class1">test3</p>,
 '₩n',
 <p class="class1">test2</p>,
 '₩n',
 <p id="p1">오늘의 주가지수 1500</p>,
 '₩n',
 <span class="class3">span tag text</span>,
 '₩n',
 <p class="class4">test3</p>,
 '₩n']
```

In [7]:

```
from urllib.request import urlopen
from bs4 import BeautifulSoup
```

스파이더맨 영화정보 7페이지 정도 가져오기

In [44]:

```
# url = 'https://movie.naver.com/movie/point/af/list.nhn?st=mcode&sword=171725&target=after'
basic_url = "https://movie.naver.com/movie/point/af/list.nhn?st=mcode&sword=171725&target=after&page
# 2page
# 3page
```

1페이지 가져오기

In [45]:

```
url1 = "https://movie.naver.com/movie/point/af/list.nhn?st=mcode&sword=171725&target=after&page=1"
page = urlopen(url1)
soup = BeautifulSoup(page, "html.parser")
comment_all = soup.find_all('td', class_='title')
comment_all
```

Out[45]:

```
[<td class="title">
 <a class="movie color_b" href="/movie/bi/mi/basic.nhn?code=171725">스파이더맨:
뉴 유니버스</a>
 <div class="list_netizen_score">
 <span class="st_off"><span class="st_on" style="width:20%">별점 - 총 10점 중</sp
an></span><em>2</em>
 </div>
 <br/>재밌긴개뿔 주인공 ㅈㄴ답답하고 개연성도 없고 뭔

                                <a class="report" href="javascript:report('kar0**
**', 'Wte2RpC0RH5tLV6/TzDCTWKk/vhahxM3nYLUizU0gsg=', '재밌긴개뿔 주인공 ㅈㄴ답답
하고 개연성도 없고 뭔 ', '17195632', 'point_after');" style="color:#8F8F8F" title
="새 창">신고</a>
```

In [46]:

```
list(comment_all[9].children)
```

Out[46]:

```
['₩n',
 <a class="movie color_b" href="/movie/bi/mi/basic.nhn?code=171725">스파이더맨: 뉴
유니버스</a>,
 '₩n',
 <div class="list_netizen_score">
 <span class="st_off"><span class="st_on" style="width:90%">별점 - 총 10점 중</span>
</span><em>9</em>
 </div>,
 '₩n',
 <br/>,
 '코믹스와 카툰과 망가의 통섭 개그 액션, 스토리뿐만 아니라 표현으로도 차원을 넘나든
다. 다양성을 품은 소년의 성장 서사가 심지어 쿨해! ₩n₩t₩t₩t₩n₩t₩t₩t₩n₩t₩t₩t₩n₩t₩t₩t₩t
₩n₩t₩t₩t₩t₩n₩t₩t₩t₩t₩n₩t₩t₩t₩t',
 <a class="report" href="javascript:report('dead****', 'UctHEh05Kz2BDfXohBC12bRXV9o1
i2+DZBNmr7A31/I=', '코믹스와 카툰과 망가의 통섭 개그 액션, 스토리뿐만 아니라 표현으
로도 차원을 넘나든다. 다양성을 품은 소년의 성장 서사가 심지어 쿨해!', '17125953', 'p
oint_after');" style="color:#8F8F8F" title="새 창">신고</a>,
 '₩n']
```

In [47]:

```
print(len( comment_all ))
```

10

In [48]:

```
temp = list(comment_all[5].children)
temp[6]
```

Out[48]:

```
'영화 줄거리가 재밌어요 WnWtWtWtWnWtWtWtWnWtWtWtWnWtWtWtWtWnWtWtWtWtWnWtWtWtWtWnWtWt
WtWt'
```

In [49]:

```
temp = list(comment_all[1].children)
result = temp[6].strip()
result
```

Out[49]:

```
'세상 모든 간지를 다 때려박은 영화'
```

여러개의 커멘트 가져오기

In [50]:

```
cnt = 0
comments = []
for comment in comment_all:
    temp= list(comment.children)
    if len(temp) < 5:
        cnt= cnt + 1
        continue
    else:
        try:
            cnt= cnt + 1
            result = temp[6].strip()
            comments.append(result)
        except:
            print("error cnt count", cnt)
comments
```

Out[50]:

```
['재밌긴개뿔 주인공 ㅈㄴ답답하고 개연성도 없고 뭔',
 '세상 모든 간지를 다 때려박은 영화',
 '1212번봐도존잼;미친영화',
 '',
 '평식이 7점 ㅋ ㅋ ㅋ ㅋ 꼭봐라',
 '영화 줄거리가 재밌어요',
 '개꿀잼완전개꿀잼임ㄹㅇ',
 '끝내주는 OST와 뛰어난 영상미!',
 '와 늦게 봤는데.... 평론레기들이 8점 이상 준거면 진짜 미친거다..재미는 모르겠지만..
여튼 웰메이드 필름',
 '코믹스와 카툰과 망가의 통섭 개그 액션, 스토리뿐만 아니라 표현으로도 차원을 넘나든
다. 다양성을 품은 소년의 성장 서사가 심지어 쿨해!']
```

1-7페이지까지 가져오기

In [52]:

```
comments = [ ]
cnt = 0
for i in range(1,8):
    url = basic_url + str(i)
    page = urlopen(url)
    soup = BeautifulSoup(page, "html.parser")

    comment_all = soup.find_all('td', class_='title')
    for comment in comment_all:
        temp= list(comment.children)
        if len(temp) < 5:
            cnt= cnt + 1
            print("len<5 case :",cnt)
            continue
        else:
            try:
                cnt= cnt + 1
                result = temp[6].strip()
                comments.append(result)
            except:
                cnt= cnt + 1
                print("len>=5 case ",cnt)
                print(temp)
print(len(comments))
print(comments)
print(cnt)
```

```
70
['재밌긴개뿔 주인공 ㅈㄴ답답하고 개연성도 없고 뭔', '세상 모든 간지를 다 때려박은 영화', '1212번봐도존잼;미친영화', '', '평식이 7점 ㅋ ㅋ ㅋ ㅋ 꼭봐라', '영화 줄거리가 재밌어요', '개꿀잼완전개꿀잼임ㄹㅇ', '끝내주는 OST와 뛰어난 영상미!', '와 늦게 봣는데.... 평론레기들이 8점 이상 준거면 진짜 미친거다..재미는 모르겠지만.. 여튼 웰메이드 필름', '코믹스와 카툰과 망가의 통섭 개그 액션, 스토리뿐만 아니라 표현으로도 차원을 넘나든다. 다양성을 품은 소년의 성장 서사가 심지어 쿨해!', '그래픽 연출 스토리등 뭐하나 빠지는게없고  등장인물도다 특징이크게 나타난다 명작중에 명작', '소니에서 만든 최고의 애니메이션', '완전 진짜 댕꿀잼 영화다. 진짜 추천!!!!', '완벽. 굿굿', '최고! 픽사와는 또다른 화끈한 영상미에 탄탄한 스토리, 감각적 ost , 모든 면에서 완벽한 애니', '소니는 일본이니까 당연 에니메이션이~ 그러나 미국식 연출 기법으로 잘 만들었다.', '이걸 이제 보다니 ㅠㅠ 죽이네요', '소니는 앞으로 애니메이션만 만들자', '보다끔 평점너무고평가됨', '영상미랑 ost 너무 좋은데? 엄청 재밌음 그리고 삿대질은 누가 먼저한거야?', '100년에 한번 나올까 말까 한 띵작임 미쳤음', '너무지루하고졸렸어요 대여료 아깝', '친구가 재미있다 재미있다 했는데도 미루다가 지금 보는데 세상 너무 재미있어서 깜짝 놀랐습니다. 영상미도 아주 훌륭하고 나오는 노래들도 너무 취저에 스파이더맨 유니폼도 진짜 개성있고 넘 취향저격..ㅜㅜ 안보신분들 언능 보세요 진짜 후회 안합니다. 킬링 타임으로 볼려다가 집중 빡 해서 보게되는 명작명작...', '진짜 최고임 아직 안봤다면 주저없이 오늘 저녁에 보시길..', '여러번 봤는데 볼때마다 점점 재밌어짐스토리,영상미,OST 다 좋음애니메이션 중에 제일 재밌게 봤음', '', '애니메이션의 카툰화~"신난다~"', '영화보다훨재밌슴지루하지않음', '애니싫어하는분들에게도 존잼일듯그어떤 스파이더맨보다 더 재밌었고 최고였음', '멀티버스를 활용하여 시대 반영까지 한 세련되고 영리한 스파이더맨', '', '말이 필요없다 의심하지마.', '롤러코스터의 속도감에도 장면 하나하나가 잊혀지지 않을 정도로 강렬하다. 장점과 단점이 분명한 게 애니메이션의 매력이라 가벼운 마음으로 보고 만족스러웠다.', '', '절대 보지마세요. 스토리도 막장이고 연출도 정신없고 눈만 아프고 화려하긴 한데 ㄴㅈ임 0.001점 준다', '애니메이션 영화로 나와도 굉장히 재미있는 스파이더맨 영화입니다.', '연출이 특이하고 재밌음', '모든걸 다 잡은 애니메이션 영화. 최고.', '개잼ㅋㅋ도랏음 10자10자', '만화책 컬러 코믹스를 영상 으로 보는 느낌  대박임', '뻔한 스파이더-맨과는 다르다. 처음부터 쉴새없이 밀어 부친다. 색감, 연출이 참 좋다. 다양한 스파이더-맨들을 볼 수 있어서 더 좋았다.', '후속편이 간절하게 기다
```

```
려지는게 얼마만이냐', '스파이더맨 뉴 유니버스 진짜 재밌다', '그래... PC요소를 집
어넣을거면 이렇게 하란 말이야...', '', '스파이더맨이 이렇게 재밌을 줄이야...',
'', '보는 내내 토할뻔 너무 정신없다..', '10점으론 부족한 영화', '돈주고 볼만함 아
니 극장에서 못봐줘서 미안해여', '', '0st도 좋고 스토리도 좋음 괜히 마블이 아니라
구...', '디즈니 픽사영화같은 느낌은 정말 안나고 재미있다.', '진짜안보면 후회  이
게 스파이더맨이지', '연출과 음악이 너무 좋아요. 근데 마일스 모랄레스 퓌웅신 새끼,
왠 쪽바리 스파이더맨? 뭔 2/3를 달라붙지마 달라붙지마 이00만 하다가 끝나냐. 퓌웅신
깜댕이 새꺄', '마블 못지 않다. 여태 보지 못한 애니매이션. 애니보고 감동 먹는 나는
아직 꼰대랑 거리가 멀다 퓨어하다', '스파이더맨 재미없다고 생각했었는데완전 바뀜
이게 진짜 스파이더맨', '내용이 너무 재밌었어요. 스파이더맨을 대표할 영화. 그리고
제가 자막판이랑 더빙을 둘다 봤는데 더빙판이 너무 잘되어 있네요ㅋㅋㅋ성우분들이 캐
릭터를 너무 잘 표현해주셨네요! 더빙판 강추입니다!', '걍 미침... 꼭보셈....', '스
파이더맨 영상화 중 최고!', '쌓아온 업적과 새로운 도전이 합쳐지니 이전에 없던 걸작
이!', '이거 1점주는 애들은 평생 신과함께나 봐야됨 딱 그 수준', '마일즈 모랄레스가
새로운 스파이더맨이 됐다.', '', '음악 아트웍 연출 캐릭터들의 매력 뭐하나 빠짐없이
좋다.', '시간 때우기용 무난한 애니매이션', '소니가 이를 갈고 만든 영화', '영화의
질이 너무 좋아 어지러움을 같이 느낄수 있었다.', '꼬맹이 각성까지 너무 오래 걸렸지
만 볼만했다. 엔딩크레딧에 스파이더맨 2099 등장해서 그런지 후편이 더 기대된다',
'인종차별을 가장 싫어하는데 흑인꼬맹이가 스파이더맨?']
70
```

In [53]:

```
import pandas as pd
```

In [54]:

```
dict_doc = {"text" : comments}
doc = pd.DataFrame(dict_doc)
```

In [55]:

```
doc.to_csv("스파이더맨리뷰.csv", index = False)
```

워드 클라우드

In [56]:

```
from wordcloud import WordCloud, STOPWORDS

import numpy as np
from PIL import Image
```

```
---------------------------------------------------------------------------
ModuleNotFoundError                       Traceback (most recent call last)
<ipython-input-56-5f2af949151d> in <module>
----> 1 from wordcloud import WordCloud, STOPWORDS
      2
      3 import numpy as np
      4 from PIL import Image

ModuleNotFoundError: No module named 'wordcloud'
```

In [57]:

```
f = open("스파이더맨리뷰.csv", encoding="utf-8")
#f = open("SpiderMan.txt", 'r', encoding='utf-8')
text = f.read()
f.close()
```

In [58]:

```
from matplotlib import rc
rc('font', family='NanumGothic')
```

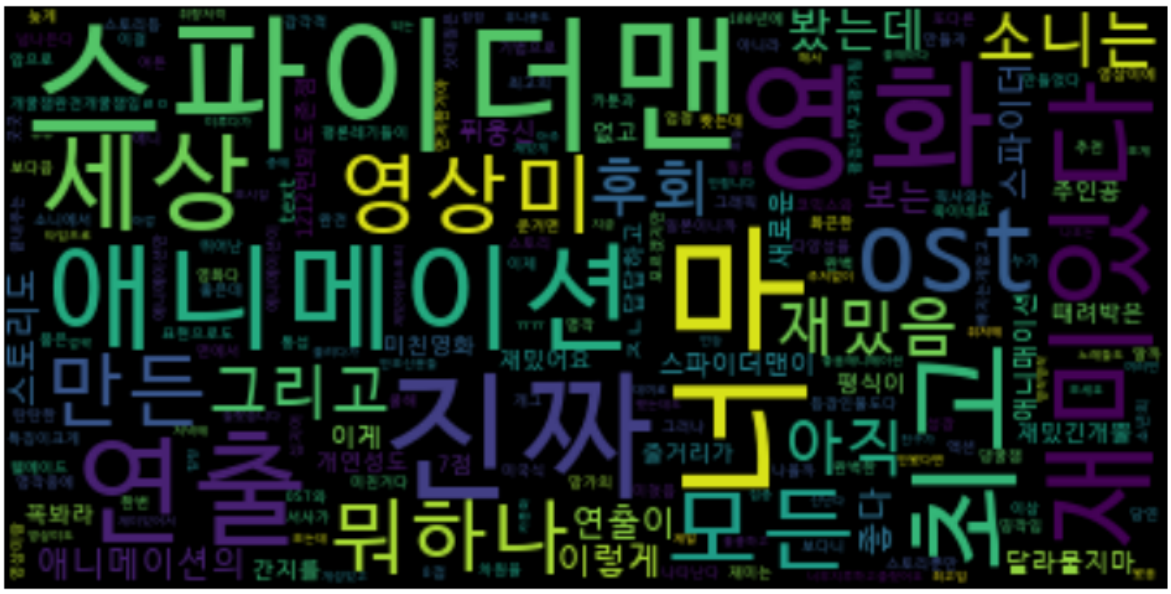
In [61]:

```
%matplotlib inline
from wordcloud import WordCloud
wcloud = WordCloud('./data/D2Coding.ttf', max_words=1000, relative_scaling = 0.2).generate(text)

import matplotlib.pyplot as plt
plt.figure(figsize=(12,12))
plt.imshow(wcloud, interpolation='bilinear')
plt.axis("off")
```

Out[61]:

```
(-0.5, 399.5, 199.5, -0.5)
```



In [ ]:

```

```