2.1.1 유튜브 랭킹 데이터 수집하기

[1]:

```
# 라이브러리 추가하기
from selenium import webdriver
from bs4 import BeautifulSoup
import time
import pandas as pd
```

[2]:

```
# webdriver로 크롬 브라우저 실행하기
browser = webdriver.Chrome('C:/Myexam/chromedriver/chromedriver.exe')
url = "https://youtube-rank.com/board/bbs/board.php?bo_table=youtube"
browser.get(url)
```

```
C:₩Users₩student₩AppData₩Local₩Temp₩ipykernel_5496₩4036677946.p
y:2: DeprecationWarning: executable_path has been deprecated, pl
ease pass in a Service object
  browser = webdriver.Chrome('C:/Myexam/chromedriver/chromedrive
r.exe')
```

[3]:

```
# 페이지 정보 가져오기
html = browser.page_source
soup = BeautifulSoup(html, 'html.parser')
```

[4]:

```python
# BeautifulSoup으로 tr 태그 추출하기
channel_list = soup.select('tr')
print(len(channel_list), '\n')
print(channel_list[0])
```

```
102

<tr>
<th class="rank"><a href="/board/bbs/board.php?bo_table=youtube&
amp;sop=and&amp;sst=rank&amp;sod=desc&amp;sfl=&amp;stx=&amp;sca=
&amp;page=1">순위 <i aria-hidden="true" class="fa fa-sort"></i>
</a></th>
<th class="td_img">이미지</th>
<th class="subject">제목</th>
<th class="subscriber_cnt"><a href="/board/bbs/board.php?bo_tabl
e=youtube&amp;sop=and&amp;sst=subscriber_cnt&amp;sod=desc&amp;sf
l=&amp;stx=&amp;sca=&amp;page=1">구독자순 <i aria-hidden="true"
class="fa fa-sort"></i></a></th>
<th class="view_cnt"><a href="/board/bbs/board.php?bo_table=yout
ube&amp;sop=and&amp;sst=view_cnt&amp;sod=desc&amp;sfl=&amp;stx=&
amp;sca=&amp;page=1">View순 <i aria-hidden="true" class="fa fa-s
ort"></i></a></th>
<th class="video_cnt"><a href="/board/bbs/board.php?bo_table=you
tube&amp;sop=and&amp;sst=video_cnt&amp;sod=desc&amp;sfl=&amp;stx
=&amp;sca=&amp;page=1">Video순 <i aria-hidden="true" class="fa f
a-sort"></i></a></th>
<th class="hit"><a href="/board/bbs/board.php?bo_table=youtube&a
mp;sop=and&amp;sst=wr_hit&amp;sod=desc&amp;sfl=&amp;stx=&amp;sca
=&amp;page=1">조회수 <i aria-hidden="true" class="fa fa-sort"></
i></a></th>
</tr>
```

[5]:

```python
# tr 태그 확인하기
channel_list = soup.select('form > table > tbody > tr')
print(len(channel_list))
```

```
100
```

[6]:

```
# 채널태그출력및태그구조 확인하기
channel = channel_list[0]
print(channel)
```

```
<tr class="aos-init aos-animate" data-aos="fade-up" data-aos-dur
ation="800">
<td class="rank">
                            1                        </td>
<td class="td_img">
<div class="info_img"><a href="https://youtube-rank.com/board/bb
s/board.php?bo_table=youtube&amp;wr_id=3203"><img class="lazyloa
d" data-src="https://yt3.ggpht.com/hZDUwjoeQqigphL4A1tkg9c6hVp5y
XmbboBR7PYFUSFj5PlJSA483NB5v7b0XVoTN9GCku3tqQ=s88-c-k-c0x00fffff
f-no-nd-rj" height="88" src="https://yt3.ggpht.com/hZDUwjoeQqigp
hL4A1tkg9c6hVp5yXmbboBR7PYFUSFj5PlJSA483NB5v7b0XVoTN9GCku3tqQ=s8
8-c-k-c0x00ffffff-no-nd-rj" width="88"/></a></div>
<p class="info_rank">1</p>
</td>
<td class="subject">
<h1>
<p <a="" class="category" href="https://youtube-rank.com/board/b
bs/board.php?bo_table=youtube&amp;sca=%EC%9D%8C%EC%95%85%2F%EB%8
C%84%EC%8A%A4%2F%EA%B0%80%EC%88%98">[음악/댄스/가수]

                                    </p>
<a href="https://youtube-rank.com/board/bbs/board.php?bo_table=y
outube&amp;wr_id=3203">



BLACKPINK
</a>
<span>
<i class="fa fa-comment"></i>

1                                                           </span>
<i aria-hidden="true" class="fa fa-heart"></i> </h1>
<h2><span><a href="https://youtube-rank.com/board/bbs/board.php?
bo_table=youtube&amp;wr_id=3203">"YG Entertainment" YG 와이지 K-
pop BLACKPINK 블랙핑크 블핑 제니 로제 리사 지수 Lisa Jisoo Jenni
e ...</a></span></h2>
<h3>
<i class="fa fa-user"></i>
                            8390만<i class="fa fa-play"></i>286
억8994만                            <i class="fa fa-video-camer
a"></i>
                            467                         <i cl
ass="fa fa-eye"></i>
                            24,555                        </h3>
</td>
<td class="subscriber_cnt">8390만</td>
<td class="view_cnt">286억8994만</td>
<td class="video_cnt">467개</td>
<td class="hit">
<strong>24,555</strong>
<span>HIT</span>
</td>
</tr>
```

[7]:

```python
#  카테고리 정보 추출하기
category = channel.select('p.category')[0].text.strip()
print(category)
```

```
[음악/댄스/가수]
```

[8]:

```python
#  채널명 찾아오기
title = channel.select('h1 > a')[0].text.strip()
print(title)
```

```
BLACKPINK
```

[9]:

```python
#  구독자 수, View 수, 동영상 수 추출하기
subscriber = channel.select('.subscriber_cnt')[0].text
view = channel.select('.view_cnt')[0].text
video = channel.select('.video_cnt')[0].text

print(subscriber)
print(view)
print(video)
```

```
8390만
286억8994만
467개
```

[10]:

```
# 반복문으로 채널 정보 추출하기
channel_list = soup.select('tbody > tr')
for channel in channel_list:
    title = channel.select('h1 > a')[0].text.strip()
    category = channel.select('p.category')[0].text.strip()
    subscriber = channel.select('.subscriber_cnt')[0].text
    view = channel.select('.view_cnt')[0].text
    video = channel.select('.video_cnt')[0].text
    print(title, category, subscriber, view, video)
```

```
BLACKPINK [음악/댄스/가수] 8390만 286억8994만 467개
BANGTANTV [음악/댄스/가수] 7310만 192억2546만 2,089개
HYBE LABELS [음악/댄스/가수] 6960만 259억2514만 1,069개
SMTOWN [음악/댄스/가수] 3140만 262억5559만 4,057개
Boram Tube Vlog [보람튜브 브이로그] [키즈/어린이] 2650만 110억
5288만 223개
JYP Entertainment [음악/댄스/가수] 2620만 184억9032만 1,597개
1MILLION Dance Studio [음악/댄스/가수] 2580만 76억3683만 4,868
개
1theK (원더케이) [음악/댄스/가수] 2440만 232억5543만 17,724개
Mnet K-POP [음악/댄스/가수] 2010만 139억1588만 30,552개
KBS WORLD TV [TV/방송] 1860만 144억9788만 61,172개
officialpsy [음악/댄스/가수] 1780만 102억7292만 123개
JFlaMusic [음악/댄스/가수] 1760만 37억3908만 313개
Jane ASMR 제인 [음식/요리/레시피] 1730만 69억2182만 1,717개
TWICE [음악/댄스/가수] 1530만 43억8322만 972개
BIGBANG [음악/댄스/가수] 1490만 75억7794만 776개
Hongyu ASMR 홍유 [음식/요리/레시피] 1450만 47억2655만 584개
Boram Tube ToysReview [보람튜브 토이리뷰] [키즈/어린이] 1450만
40억6826만 591개
```

[11]:

```
# 페이지별 URL 만들기
page = 1
url = 'https://youtube-rank.com/board/bbs/board.php?bo_table=youtube&
print(url)
```

https://youtube-rank.com/board/bbs/board.php?bo_table=youtube&page=1 (https://youtube-rank.com/board/bbs/board.php?bo_table=youtube&page=1)

[12]:

```python
#  반복문으로 유튜브 랭킹 화면의 여러 페이지를 크롤링하기
results = []
for page in range(1,11):
    url = f"https://youtube-rank.com/board/bbs/board.php?bo_table=you
    browser.get(url)
    time.sleep(2)
    html = browser.page_source
    soup = BeautifulSoup(html, 'html.parser')
    channel_list = soup.select('form > table > tbody > tr')
    for channel in channel_list:
        title = channel.select('h1 > a')[0].text.strip()
        category = channel.select('p.category')[0].text.strip()
        subscriber = channel.select('.subscriber_cnt')[0].text
        view = channel.select('.view_cnt')[0].text
        video = channel.select('.video_cnt')[0].text
        data = [title, category, subscriber, view, video]
        results.append(data)
```

[13]:

```python
# 데이터 칼럼명을 설정하고 엑셀 파일로 저장하기
df = pd.DataFrame(results)
df.columns = ['title', 'category', 'subscriber', 'view', 'video']
df.to_excel('./files/youtube_rank.xlsx', index = False)
```

2.1.2 유튜브 랭킹 데이터 시각화하기

[14]:

```python
# 라이브러리 추가하기
import pandas as pd
import matplotlib.pyplot as plt
```

[15]:

```python
# 그래프에서 한글을 표기하기 위한 글꼴 변경(윈도우, macOS에 대해 각각 처리
from matplotlib import font_manager, rc
import platform
if platform.system() == 'Windows':
    path = 'c:/Windows/Fonts/malgun.ttf'
    font_name = font_manager.FontProperties(fname = path).get_name()
    rc('font', family = font_name)
elif platform.system() == 'Darwin':
    rc('font', family = 'AppleGothic')
else:
    print('Check your OS system')
```

[16]:

```
# 엑셀 파일 불러오기
df = pd.read_excel('./files/youtube_rank.xlsx')
df.head()
```

	title	category	subscriber	view	video
0	BLACKPINK	[음악/댄스/가수]	8390만	286억8994만	467개
1	BANGTANTV	[음악/댄스/가수]	7310만	192억2546만	2,089개
2	HYBE LABELS	[음악/댄스/가수]	6960만	259억2514만	1,069개
3	SMTOWN	[음악/댄스/가수]	3140만	262억5559만	4,057개
4	Boram Tube Vlog [보람튜브 브이로그]	[키즈/어린이]	2650만	110억5288만	223개

[17]:

```
# 데이터 살펴보기
df.tail()
```

	title	category	subscriber	view	video
995	대륙남TV [clark tv]	[BJ/인물/연예인]	69만	4억1652만	3,471개
996	ASMR Boyoung 반보영	[미분류]	69만	1억2674만	219개
997	강하나 스트레칭_stretching	[스포츠/운동]	69만	7429만	363개
998	꾸삐KUPI	[키즈/어린이]	69만	4억5769만	846개
999	안될과학 Unrealscience	[IT/기술/컴퓨터]	69만	8132만	525개

[18]:

```
# 데이터 살펴보기
df['subscriber'][0:10]
```

```
0    8390만
1    7310만
2    6960만
3    3140만
4    2650만
5    2620만
6    2580만
7    2440만
8    2010만
9    1860만
Name: subscriber, dtype: object
```

[19]:

```
# 데이터 살펴보기
df['subscriber'].str.replace('만', '0000')[0:10]
```

```
0     83900000
1     73100000
2     69600000
3     31400000
4     26500000
5     26200000
6     25800000
7     24400000
8     20100000
9     18600000
Name: subscriber, dtype: object
```

[20]:

```
# replaced_subscriber 시리즈 문자열 변경하기
df['replaced_subscriber'] = df['subscriber'].str.replace('만', '0000')
df.head()
```

	title	category	subscriber	view	video	replaced_subscriber
0	BLACKPINK	[음악/댄스/가수]	8390만	286억8994만	467개	83900000
1	BANGTANTV	[음악/댄스/가수]	7310만	192억2546만	2,089개	73100000
2	HYBE LABELS	[음악/댄스/가수]	6960만	259억2514만	1,069개	69600000
3	SMTOWN	[음악/댄스/가수]	3140만	262억5559만	4,057개	31400000
4	Boram Tube Vlog [보람튜브 브이로그]	[키즈/어린이]	2650만	110억5288만	223개	26500000

[21]:

```
# 데이터 상세 정보
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 6 columns):
 #   Column               Non-Null Count  Dtype 
---  ------               --------------  ----- 
 0   title                1000 non-null   object
 1   category             1000 non-null   object
 2   subscriber           1000 non-null   object
 3   view                 1000 non-null   object
 4   video                1000 non-null   object
 5   replaced_subscriber  1000 non-null   object
dtypes: object(6)
memory usage: 47.0+ KB
```

[22]:

```
# Series 데이터 타입 변환하기
df['replaced_subscriber'] = df['replaced_subscriber'].astype('int')
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1000 entries, 0 to 999
Data columns (total 6 columns):
 #   Column               Non-Null Count  Dtype 
---  ------               --------------  ----- 
 0   title                1000 non-null   object
 1   category             1000 non-null   object
 2   subscriber           1000 non-null   object
 3   view                 1000 non-null   object
 4   video                1000 non-null   object
 5   replaced_subscriber  1000 non-null   int32 
dtypes: int32(1), object(5)
memory usage: 43.1+ KB
```

[23]:

```
# 카테고리별 구독자 수, 채널 수 피봇 테이블 생성하기
pivot_df = df.pivot_table(index = 'category', values = 'replaced_subs
pivot_df.head()
```

	sum	count
	replaced_subscriber	replaced_subscriber
category		
[BJ/인물/연예인]	102080000	62
[IT/기술/컴퓨터]	9940000	8
[TV/방송]	265340000	130
[게임]	67350000	53
[교육/강의]	28580000	22

[24]:

```
# 데이터프레임의 칼럼명 변경하기
pivot_df.columns = ['subscriber_sum', 'category_count']
pivot_df.head()
```

	subscriber_sum	category_count
category		
[BJ/인물/연예인]	102080000	62
[IT/기술/컴퓨터]	9940000	8
[TV/방송]	265340000	130
[게임]	67350000	53
[교육/강의]	28580000	22

[25]:

```python
# 데이터프레임의인덱스초기화하기
pivot_df = pivot_df.reset_index()
pivot_df.head()
```

	category	subscriber_sum	category_count
0	[BJ/인물/연예인]	102080000	62
1	[IT/기술/컴퓨터]	9940000	8
2	[TV/방송]	265340000	130
3	[게임]	67350000	53
4	[교육/강의]	28580000	22

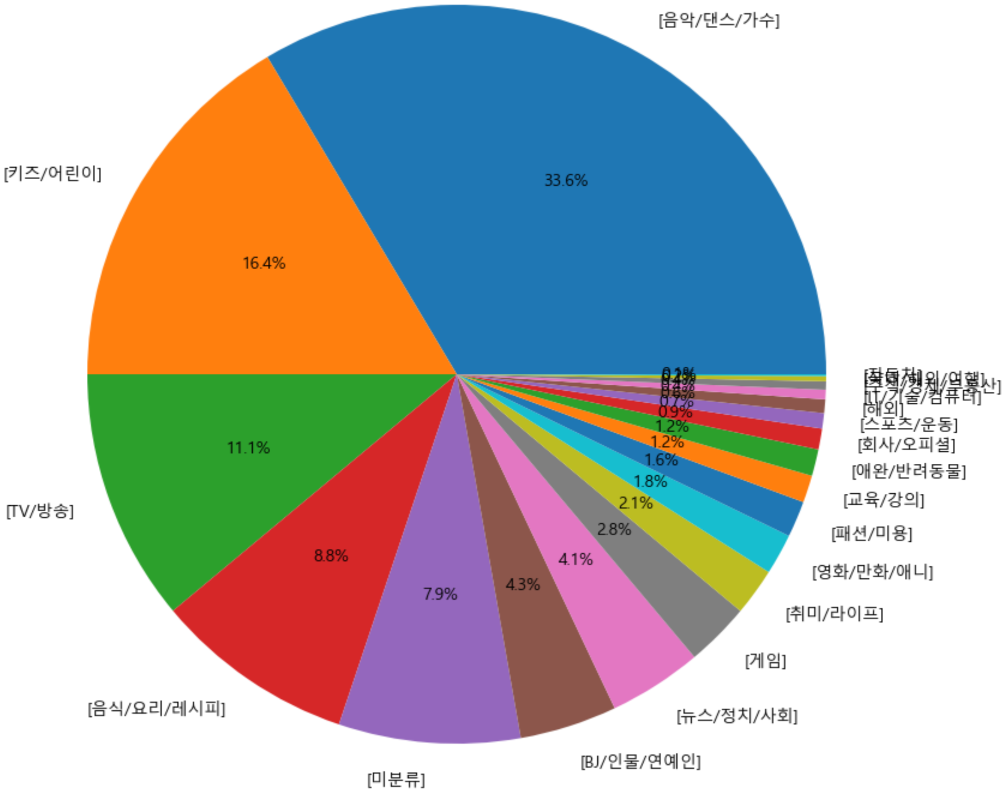
[26]:

```python
# 데이터프레임을내림차순정렬하기
pivot_df = pivot_df.sort_values(by='subscriber_sum', ascending=False)
pivot_df.head()
```

	category	subscriber_sum	category_count
12	[음악/댄스/가수]	805310000	159
16	[키즈/어린이]	394280000	134
2	[TV/방송]	265340000	130
11	[음식/요리/레시피]	210480000	71
7	[미분류]	190610000	159

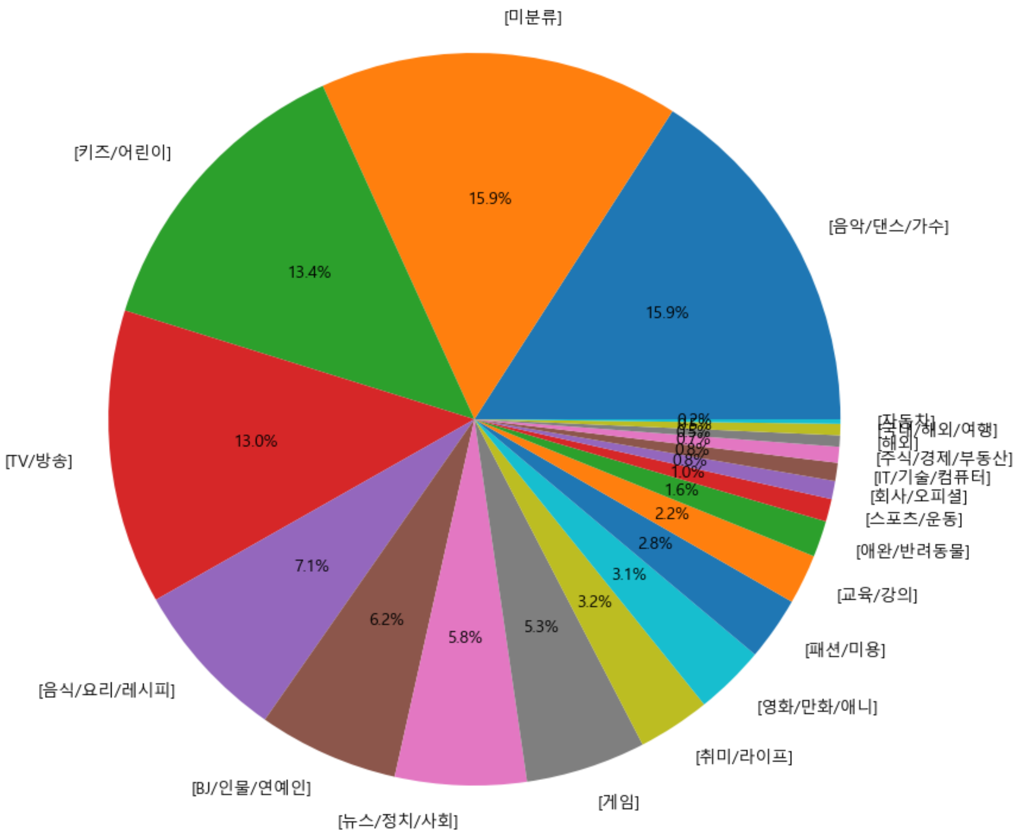
[27]:

```
# 카테고리별구독자수시각화하기
plt.figure(figsize = (30,10))
plt.pie(pivot_df['subscriber_sum'], labels=pivot_df['category'], autop
plt.show()
```



[28]:

```
# 카테고리별 채널 수 시각화하기
pivot_df = pivot_df.sort_values(by='category_count', ascending=False)
pivot_df.head()
plt.figure(figsize = (30,10))
plt.pie(pivot_df['category_count'], labels=pivot_df['category'], auto
plt.show()
```



[ ]:

```

```

[ ]:

```

```