2.1.1 유튜브 랭킹 데이터 수집하기

In [1]:
```
# 라이브러리 추가하기
from selenium import webdriver
from bs4 import BeautifulSoup
import time
import pandas as pd
```

In [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)
```

In [3]:
```
# 페이지 정보 가져오기
html = browser.page_source
soup = BeautifulSoup(html, 'html.parser')
```

In [4]:
```
# 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_table=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=youtube&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-sort"></i></a></th>
<th class="video_cnt"><a href="/board/bbs/board.php?bo_table=youtube&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 fa-sort"></i></a></th>
<th class="hit"><a href="/board/bbs/board.php?bo_table=youtube&amp;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>
```

In [5]:
```
# tr 태그 확인하기
channel_list = soup.select('form > table > tbody > tr')
print(len(channel_list))
```

```
100
```

In [6]:
```
# 채널태그출력및태그구조 확인하기
channel = channel_list[0]
print(channel)
```

```
<tr class="aos-init aos-animate" data-aos="fade-up" data-aos-duration="800">
<td class="rank">
                        1                    </td>
<td class="td_img">
<div class="info_img"><a href="https://youtube-rank.com/board/bbs/board.php?bo_table=youtube&amp;wr_id=3203"><img class="lazyloa
d" data-src="https://yt3.ggpht.com/ytc/AKedOLTat8_vr7h2i29n67IvNLMp1F-9Ch2ejFRIRzt0zQ=s88-c-k-c0x00ffffff-no-rj-mo" height="88"
src="https://yt3.ggpht.com/ytc/AKedOLTat8_vr7h2i29n67IvNLMp1F-9Ch2ejFRIRzt0zQ=s88-c-k-c0x00ffffff-no-rj-mo" width="88"/></a></di
v>
<p class="info_rank">1</p>
</td>
<td class="subject">
<h1>
<p <a="" class="category" href="https://youtube-rank.com/board/bbs/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=youtube&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 Jennie ...</a></span></h2>
<h3>
<i class="fa fa-user"></i>
                            6410만<i class="fa fa-play"></i>190억1809만                                <i class="fa fa-video-camer
a"></i>
                            371                              <i class="fa fa-eye"></i>
                            14,586                          </h3>
</td>
<td class="subscriber_cnt">6410만</td>
<td class="view_cnt">190억1809만</td>
<td class="video_cnt">371개</td>
<td class="hit">
<strong>14,586</strong>
<span>HIT</span>
</td>
</tr>
```

In [7]:
```
#  카테고리 정보 추출하기
category = channel.select('p.category')[0].text.strip()
print(category)
```

```
[음악/댄스/가수]
```

In [8]:
```
#  채널명 찾아오기
title = channel.select('h1 > a')[0].text.strip()
print(title)
```

```
BLACKPINK
```

In [9]:
```
#  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)
```

```
6410만
190억1809만
371개
```

In [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)
```

```
KBS Entertain [TV/방송] 447만 70억6117만 99,977개
떵개떵 [음식/요리/레시피] 447만 25억0821만 4,333개
ASTRO 아스트로 [음악/댄스/가수] 441만 4억1311만 434개
Red Velvet [음악/댄스/가수] 437만 5억9464만 140개
뽀로로(Pororo) [키즈/어린이] 437만 58억7787만 3,597개
MBCdrama [TV/방송] 433만 49억0330만 53,974개
하루한끼 one meal a day [음식/요리/레시피] 431만 4억1059만 162개
TREASURE (트레저) [음악/댄스/가수] 427만 8억2923만 253개
푸메Fume [음식/요리/레시피] 425만 8억5891만 464개
슈슈토이 Shushu ToysReview [키즈/어린이] 420만 15억5498만 581개
채널 NCT DAILY [TV/방송] 419만 7억2412만 467개
Raon Lee [음악/댄스/가수] 419만 9억1695만 295개
EA SPORTS FIFA [미분류] 418만 7억9367만 792개
With Kids Playground [위드키즈 놀이터] [키즈/어린이] 416만 15억7093만 399개
Cooking tree 쿠킹트리 [음식/요리/레시피] 416만 3억7895만 1,134개
[Dorothy]도로시 [음식/요리/레시피] 415만 9억8684만 905개
JTBC Drama [TV/방송] 412만 42억1267만 22,563개
SBS TV동물농장x애니멀봐 [애완/반려동물] 412만 37억4911만 3,002개
NCT [음식/댄스/가수] 411만 2억7111만 219개
tzuyang쯔양 [음식/요리/레시피] 411만 5억0727만 236개
```

In [11]:
```
# 페이지별 URL 만들기
page = 1
url = 'https://youtube-rank.com/board/bbs/board.php?bo_table=youtube&page={}'.format(page)
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)
```

In [12]:
```
#  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=youtube&page={page}"
    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)
```

In [13]:
```
# 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

In [14]:
```
# 라이브러리 추가하기
import pandas as pd
import matplotlib.pyplot as plt
```

In [15]:
```
# 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')
```

In [16]:
```
# 엑셀 파일 불러오기
df = pd.read_excel('./files/youtube_rank.xlsx')
df.head()
```

Out[16]:

	title	category	subscriber	view	video
0	BLACKPINK	[음악/댄스/가수]	6410만	190억1809만	371개
1	HYBE LABELS	[음악/댄스/가수]	6040만	187억5681만	655개
2	BANGTANTV	[음악/댄스/가수]	5650만	122억1297만	1,580개
3	SMTOWN	[음악/댄스/가수]	2850만	219억1233만	3,729개
4	Boram Tube Vlog [보람튜브 브이로그]	[키즈/어린이]	2650만	110억5288만	223개

In [17]:
```
# 데이터 살펴보기
df.tail()
```

Out[17]:

	title	category	subscriber	view	video
995	OGN	[게임]	51만	6억0888만	27,433개
996	밉지않은 관종언니	[미분류]	51만	6795만	170개
997	미소	[게임]	51만	2억1078만	3,084개
998	Muggo	[음식/요리/레시피]	51만	1억8372만	1,388개
999	임선비	[게임]	51만	2억3429만	1,134개

In [18]:
```
# 데이터 살펴보기
df['subscriber'][0:10]
```

Out[18]:
```
0    6410만
1    6040만
2    5650만
3    2850만
4    2650만
5    2420만
6    2330만
7    2160만
8    1930만
9    1840만
Name: subscriber, dtype: object
```

In [19]:
```
# 데이터 살펴보기
df['subscriber'].str.replace('만', '0000')[0:10]
```

Out[19]:
```
0    64100000
1    60400000
2    56500000
3    28500000
4    26500000
5    24200000
6    23300000
7    21600000
8    19300000
9    18400000
Name: subscriber, dtype: object
```

In [20]:
```
# replaced_subscriber 시리즈 문자열 변경하기
df['replaced_subscriber'] = df['subscriber'].str.replace('만', '0000')
df.head()
```

Out[20]:

	title	category	subscriber	view	video	replaced_subscriber
0	BLACKPINK	[음악/댄스/가수]	6410만	190억1809만	371개	64100000
1	HYBE LABELS	[음악/댄스/가수]	6040만	187억5681만	655개	60400000
2	BANGTANTV	[음악/댄스/가수]	5650만	122억1297만	1,580개	56500000
3	SMTOWN	[음악/댄스/가수]	2850만	219억1233만	3,729개	28500000
4	Boram Tube Vlog [보람튜브 브이로그]	[키즈/어린이]	2650만	110억5288만	223개	26500000

In [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
```

In [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
```

In [23]:
```
# 카테고리별 구독자 수, 채널 수 피봇 테이블 생성하기
pivot_df = df.pivot_table(index = 'category', values = 'replaced_subscriber', aggfunc = ['sum','count'])
pivot_df.head()
```

Out[23]:

	sum	count
	replaced_subscriber	replaced_subscriber
category		
[BJ/인물/연예인]	97630000	71
[IT/기술/컴퓨터]	7970000	8
[TV/방송]	229930000	146
[게임]	67190000	74
[교육/강의]	23440000	22

In [24]:
```
# 데이터프레임의 칼럼명 변경하기
pivot_df.columns = ['subscriber_sum', 'category_count']
pivot_df.head()
```

Out[24]:

	subscriber_sum	category_count
category		
[BJ/인물/연예인]	97630000	71
[IT/기술/컴퓨터]	7970000	8
[TV/방송]	229930000	146
[게임]	67190000	74
[교육/강의]	23440000	22

In [25]:
```
# 데이터프레임의인덱스초기화하기
pivot_df = pivot_df.reset_index()
pivot_df.head()
```

Out[25]:

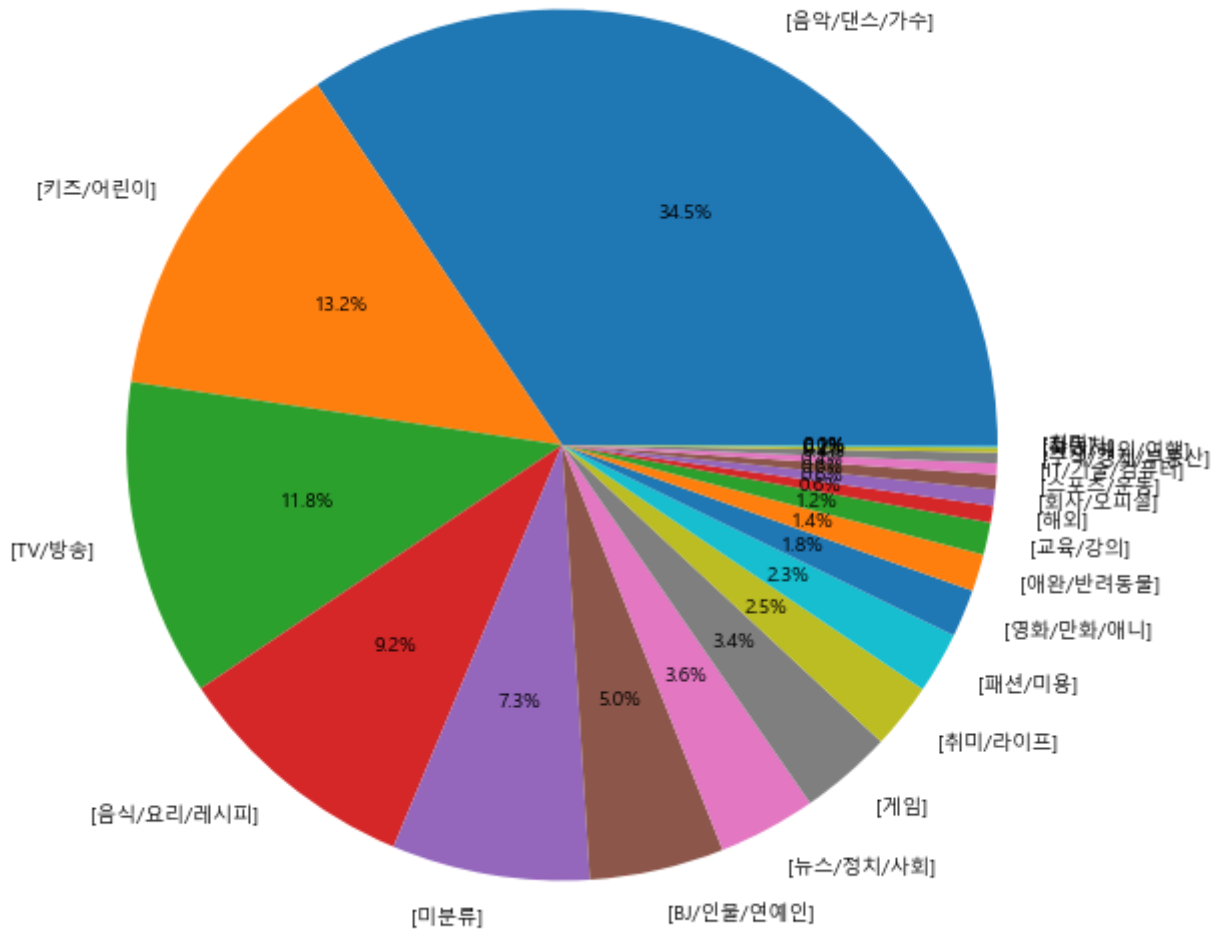
	category	subscriber_sum	category_count
0	[BJ/인물/연예인]	97630000	71
1	[IT/기술/컴퓨터]	7970000	8
2	[TV/방송]	229930000	146
3	[게임]	67190000	74
4	[교육/강의]	23440000	22

In [26]:
```
# 데이터프레임을내림차순정렬하기
pivot_df = pivot_df.sort_values(by='subscriber_sum', ascending=False)
pivot_df.head()
```

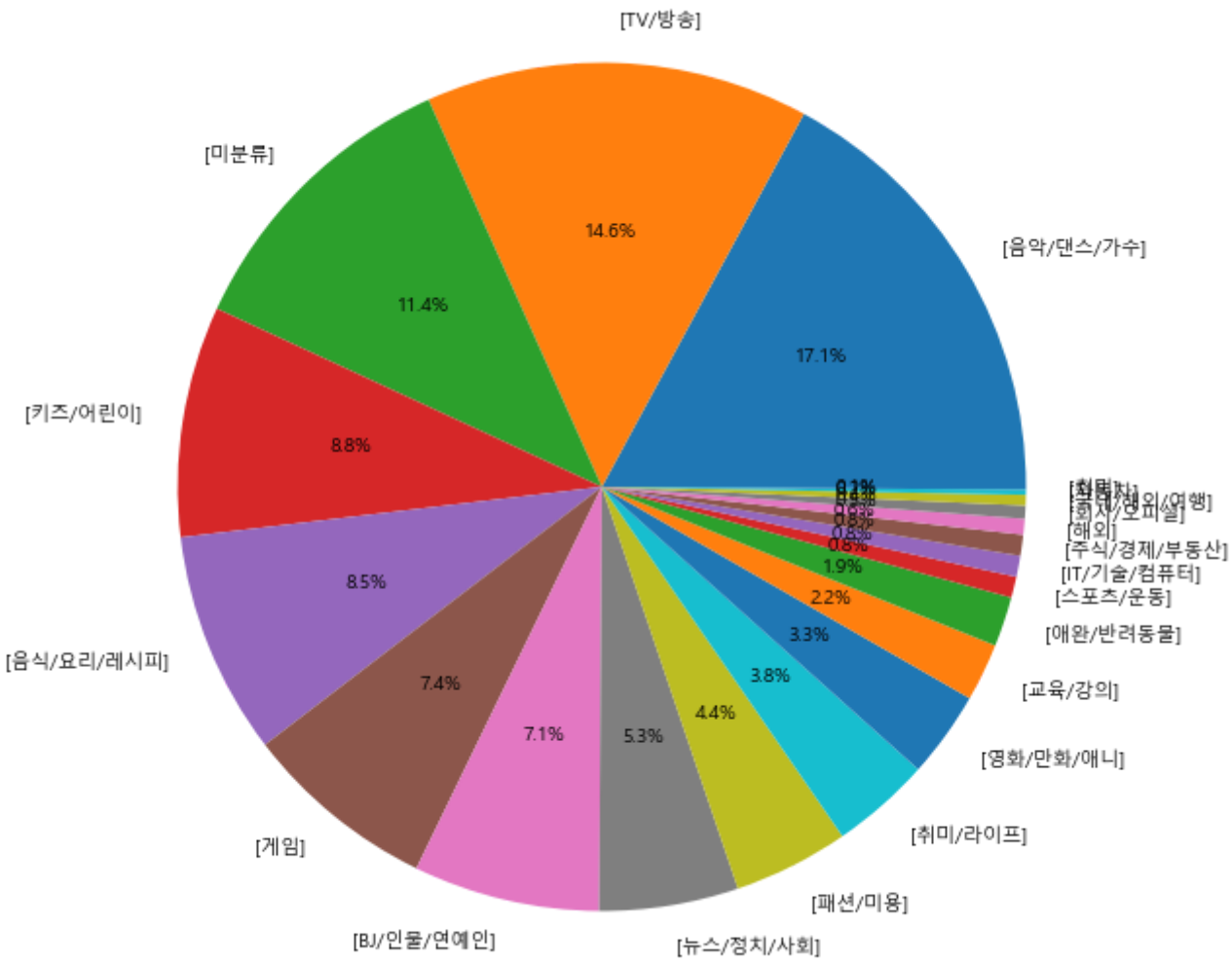
Out[26]:

	category	subscriber_sum	category_count
12	[음악/댄스/가수]	674950000	171
17	[키즈/어린이]	257840000	88
2	[TV/방송]	229930000	146
11	[음식/요리/레시피]	180700000	85
7	[미분류]	143640000	114

In [27]:
```
# 카테고리별구독자수시각화하기
plt.figure(figsize = (30,10))
plt.pie(pivot_df['subscriber_sum'], labels=pivot_df['category'], autopct='%1.1f%%')
plt.show()
```



In [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'], autopct='%1.1f%%')
plt.show()
```



In [ ]:

In [ ]: