

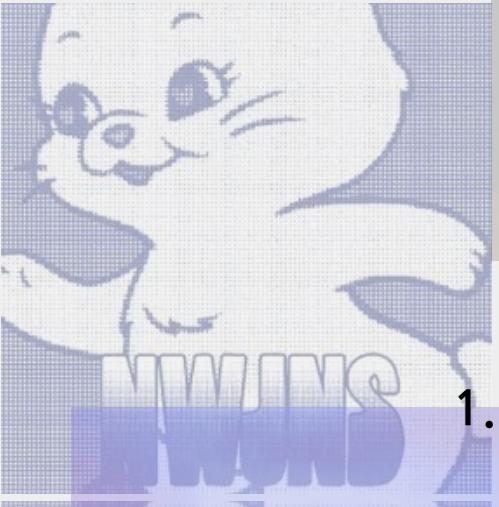


# Melon

음악이 필요한 순간, 멜론



김민호 전현준 주한솔 지우근



1.

프로젝트 개요



2.

질문 소개



3.

Web Scraping



MAP OF THE SOUL PERSONA



4. PyMySQL Table Design



5. 데이터 분석



선



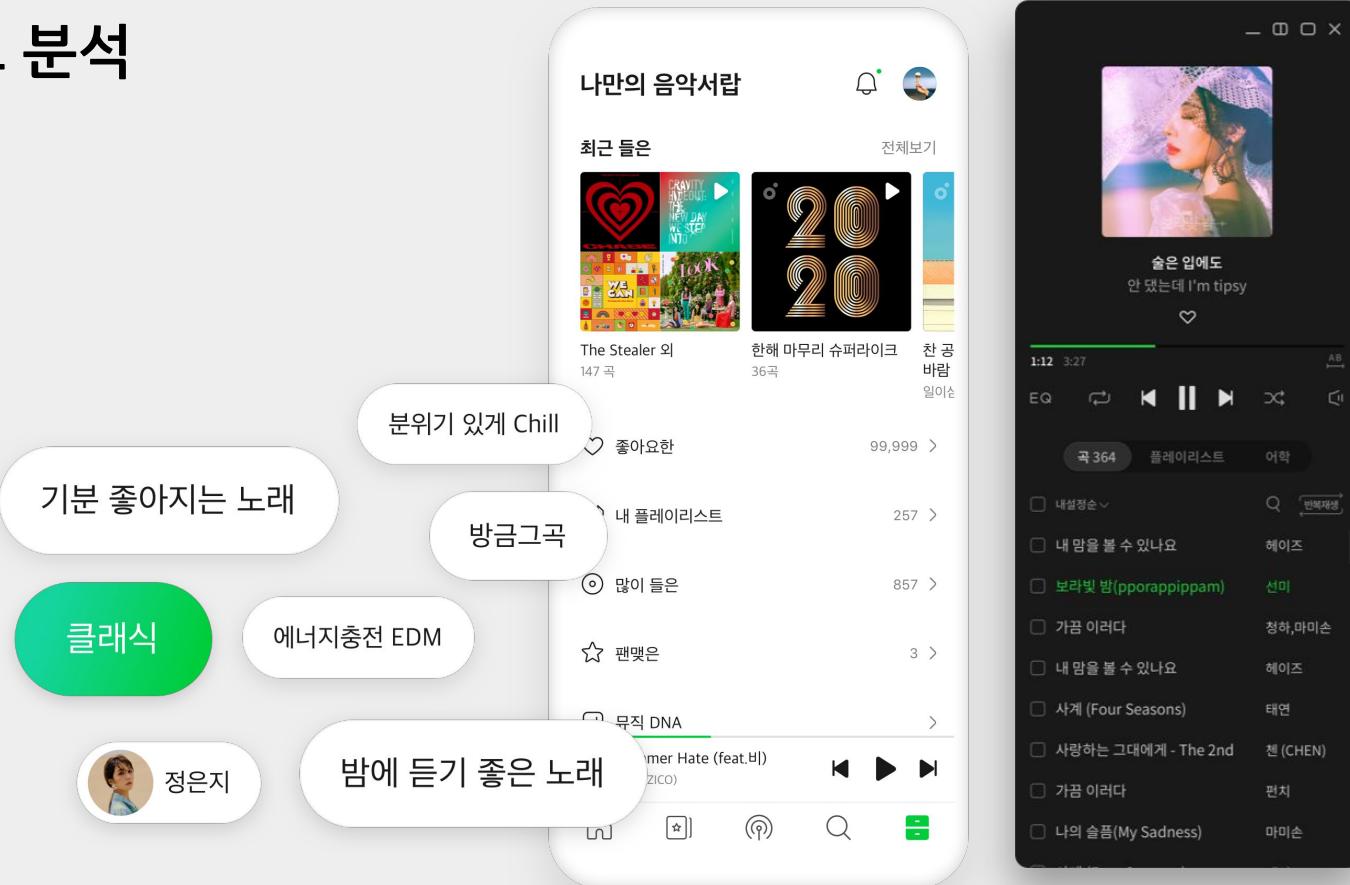
이무진



이우정



## 멜론 차트 분석





# 데이터로부터 얻을 수 있는 분석 결과

## 질문 1

2012~2022 top100 중 가장 많이 순위권 든 노래  
top15, 가수 top15

## 질문 4

가장 많이 차트 인한 가수 top5 중, 연도별 차트에 가장  
많이 올라간 가수와 그들의 총 곡 수

## 질문 2

플랫폼별 2012~2022 1위곡 비교(melon, circle  
chart)

## 질문 5

질문 4에 속하면서 현재 가장 인기있는 아티스트  
top5에도 속하는 가수의 팬 남녀, 연령대 비율

## 질문 3

top3 기획사 중, 가장 많이 이름을 올린 가수 top15

## 질문 6

아티스트 랭킹 1~50위에 속하며 top100에도 속하는  
가수들의 차트 인 곡수와 종합 세부점수



# Web Scraping

## 2012~2021 Top100 Melon

```
veniu
    , bs4 import BeautifulSoup
    , port requests
    , from selenium import webdriver
    , import time
    , from selenium.webdriver.common.by import By
    , from selenium.webdriver.common.keys import Keys
    , import pymysql

    db = pymysql.connect(host='127.0.0.1', port=3306, user='root', passwd='1234', db='melon', charset='utf8')

    cursor = db.cursor()

    for year in range(2012, 2022):
        link = "https://www.melon.com/chart/age/index.htm?chartType=Y&chartGenre=KPOP&chartDate={}#params$5idx50=1".format(year)

        driver = webdriver.Chrome()
        driver.get(link)

        html = driver.page_source

        time.sleep(3)
        driver.close()
        driver.quit()
        bs = BeautifulSoup(html, 'lxml')

        idx = len(bs.find_all('tr', {'class': 'lst50'}))
        idx2 = len(bs.find_all('tr', {'class': 'lst100'}))
        tr_all1 = bs.find_all('tr', {'class': 'lst50'})
        tr_all2 = bs.find_all('tr', {'class': 'lst100'})

        rank = 1

        for i in range(0, idx):
            song_name = tr_all1[i].find('div', {'class': 'ellipsis rank01'}).text.strip()
            singer_name = tr_all1[i].find('div', {'class': 'ellipsis rank02'}).find('a').text.strip()
            likes = tr_all1[i].find('span', {'class': 'cnt'}).text.replace('\n', '').replace('출건수 ', '')
            sql = """INSERT INTO top100 VALUES({0}, {1}, {2}, {3}, {4});""".format(year, rank, song_name, singer_name, likes)

            rank += 1
            cursor.execute(sql)

        for i in range(0, idx2):
            song_name = tr_all2[i].find('div', {'class': 'ellipsis rank01'}).text.strip()
            singer_name = tr_all2[i].find('div', {'class': 'ellipsis rank02'}).find('a').text.strip()
            likes = tr_all2[i].find('span', {'class': 'cnt'}).text.replace('\n', '').replace('출건수 ', '')
            sql = """INSERT INTO top100 VALUES({0}, {1}, {2}, {3}, {4});""".format(year, rank, song_name, singer_name, likes)

            rank += 1
            cursor.execute(sql)

    it()
```

## 2022 Top100 Melon

```
veniu
    , bs4 import BeautifulSoup
    , port requests
    , from selenium import webdriver
    , import time
    , from selenium.webdriver.common.by import By
    , from selenium.webdriver.common.keys import Keys
    , import pymysql

    db = pymysql.connect(host='127.0.0.1', port=3306, user='root', passwd='1234', db='melon', charset='utf8')

    cursor = db.cursor()

    link = "https://www.melon.com/chart/week/index.htm"

    driver = webdriver.Chrome()
    driver.get(link)

    html = driver.page_source

    time.sleep(3)
    driver.close()
    driver.quit()
    bs = BeautifulSoup(html, 'lxml')

    idx = len(bs.find_all('tr', {'class': 'lst50'}))
    idx2 = len(bs.find_all('tr', {'class': 'lst100'}))
    tr_all1 = bs.find_all('tr', {'class': 'lst50'})
    tr_all2 = bs.find_all('tr', {'class': 'lst100'})

    rank = 1
    year = 2022
    for i in range(0, idx):
        song_name = tr_all1[i].find('div', {'class': 'ellipsis rank01'}).text.strip()
        singer_name = tr_all1[i].find('div', {'class': 'ellipsis rank02'}).find('a').text.strip()
        likes = tr_all1[i].find('span', {'class': 'cnt'}).text.replace('\n', '').replace('출건수 ', '')
        sql = """INSERT INTO top100 VALUES({0}, {1}, {2}, {3}, {4});""".format(year, rank, song_name, singer_name, likes)

        rank += 1
        cursor.execute(sql)

    for i in range(0, idx2):
        song_name = tr_all2[i].find('div', {'class': 'ellipsis rank01'}).text.strip()
        singer_name = tr_all2[i].find('div', {'class': 'ellipsis rank02'}).find('a').text.strip()
        likes = tr_all2[i].find('span', {'class': 'cnt'}).text.replace('\n', '').replace('출건수 ', '')
        sql = """INSERT INTO top100 VALUES({0}, {1}, {2}, {3}, {4});""".format(year, rank, song_name, singer_name, likes)

        rank += 1
        cursor.execute(sql)

    it()
```



# Web Scraping

## 2012~2021 Top100 Circle Chart

```
# import 모듈
import pymysql
from bs4 import BeautifulSoup
import requests
from selenium import webdriver
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys

## db 연결 부분
db = pymysql.connect(host='localhost', port=3306, user='root', passwd='1234', db='melon', charset='utf8')
cursor = db.cursor()

#####
## 서브 차트 2012-2021 #####
#####

year = 2012
rank = 1

for i in range(10):
    rank = i
    link = f'https://circlechart.kr/page_chart/onoff.circle?nationGb=K&serviceGb=ALL&targetTime={year}&hitYear={year}&termGb=year&yea
    driver = webdriver.Chrome()
    driver.get(link)

    time.sleep(3)

    html = driver.page_source

    driver.close()
    driver.quit()

    bs = BeautifulSoup(html, 'xml')

    music_names = bs.find_all('div', {'class': 'font-bold mb-2'})[:100]
    names = bs.find_all('div', {'class': 'text-sm text-gray-400 font-bold'})[:100]
    agency = bs.find_all('td', {'class': 'text-left text-xs'})[:100]

    for i in range(len(names)):
        sql = """
        INSERT INTO circle_digital_chart_top100 VALUES(
        {rank}, "{music_names[i].text}", "{names[i].text}", "{agency[i].find('span').text}", {year});
        """
        cursor.execute(sql)
        rank += 1

    year += 1
    commit()
    close()
```

## 2022 Top100 Circle Chart

```
# import 모듈
import pymysql
from bs4 import BeautifulSoup
import requests
from selenium import webdriver
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys

## db 연결 부분
db = pymysql.connect(host='localhost', port=3306, user='root', passwd='1234', db='melon', charset='utf8')
cursor = db.cursor()

#####
## 서브 차트 2022 ###
#####

link = 'https://circlechart.kr/page_chart/onoff.circle?serviceGb=ALL&termGb=week'

driver = webdriver.Chrome()
driver.get(link)

time.sleep(3)

html = driver.page_source

driver.close()
driver.quit()

bs = BeautifulSoup(html, 'xml')

music_names = bs.find_all('div', {'class': 'font-bold mb-2'})[:100]
names = bs.find_all('div', {'class': 'text-sm text-gray-400 font-bold'})[:100]
agency = bs.find_all('td', {'class': 'text-left text-xs'})[:100]
rank = 1

for i in range(len(names)):
    sql = """
    INSERT INTO circle_digital_chart_top100 VALUES(
    {rank}, "{music_names[i].text}", "{names[i].text}", "{agency[i].find('span').text}", 2022);
    """
    cursor.execute(sql)
    rank += 1

cursor.execute(sql)
rank += 1
```



# Web Scraping

## Artist Rank Top 5

```
# import 부른
# mysql
from bs4 import BeautifulSoup
import requests
from selenium import webdriver
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys

### db 선언 부분
db = pymysql.connect(host='127.0.0.1', port=3306, user='root', passwd='1234', db='melon', charset='utf8')
cursor = db.cursor()

#####
## 멜론 아티스트 랭킹 5 ##
#####

### BeautifulSoup 선언 부분

user_agent = "Mozilla/5.0"

header01 = {"User-Agent": user_agent}
response = requests.get('https://www.melon.com/artistplus/artistchart/index.htm?chartGubunCode=DP0000027', headers=header01)
response.raise_for_status()

bs = BeautifulSoup(response.text, 'lxml')
ls = bs.findAll('strong', {'class': 'ellipsis'})
```

lis = bs.findAll('div', {'class': 'graph'})
inx = len(bs.findAll('div', {'class': 'wrap\_artist\_chart'})).find('ul').findAll('li')

```
b = []
rank = 1

for i in range(1, inx+1):
    b = []
    name = ls[i-1].text
    t = f'artist rank{rank}'
    per = bs.find('p', {'class': t}).text.strip()
    for li in lis[i-1].findAll('li'):
        b.append(li.find('span', {'class': 'graph_bar'}).text)
    sql = f"""
INSERT INTO artist_rank_top5 VALUES
({rank}, '{per}', '{name}', '{b[0]}', '{b[1]}', '{b[2]}', '{b[3]}', '{b[4]}', '{b[5]}', '{b[6]}', '{b[7]}', '{b[8]}');
    """
    cursor.execute(sql)
    rank += 1
```

## Artist Rank Top 50

```
# mysql
# import BeautifulSoup
# requests
from selenium import webdriver
import time
from selenium.webdriver.common.by import By
from selenium.webdriver.common.keys import Keys

#####
## db 선언 부분
db = pymysql.connect(host='localhost', port=3306, user='root', passwd='1234', db='melon', charset='utf8')
cursor = db.cursor()

#####
## 멜론 아티스트 랭킹 50 ##
#####

link = 'https://www.melon.com/artistplus/artistchart/index.htm?chartGubunCode=DP0000027', headers=header01'
driver = webdriver.Chrome()
driver.get(link)

for i in range(4):
    driver.find_element(By.XPATH, '// *[ @ id = "conts"] / div[3] / div[3] / button').click()
    time.sleep(1)

time.sleep(3)

html = driver.page_source
driver.close()
driver.quit()

bs = BeautifulSoup(html, 'html.parser')
inx = len(bs.findAll('li', {'class': 'artistplus_li'}))
ar = bs.findAll('li', {'class': 'artistplus_li'})

score_ar = []
rank = 1

for i in range(inx):
    score_ar = []
    scores = ar[i].find('tr').find_next('tr').findAll('div')

    #####
    name = ar[i].find('div', {'class': 'wrap_info'}).find('a').text
    #####
    fan = (ar[i].find('dd', {'class': 'gubun'}).text.strip().replace('\t', '')).replace('\n', '')

    for score in scores:
        score_ar.append(score.text)

    sql = f"""
INSERT INTO artist_rank_top50 VALUES
({rank}, '{name}', {fan}, {float(score_ar[0])}, {float(score_ar[1])}, {float(score_ar[2])}, {float(score_ar[3])}, {float(score_ar[4])});
    """
    cursor.execute(sql)
    rank += 1
```



# MySQL Table Design

## Top100 Melon

```
-- top100 테이블 생성  
create table top100 (  
    years int,  
    ranks int,  
    song_name varchar(300),  
    singer_name varchar(300),  
    likes varchar(100)  
);
```

## Top100 Circle Chart

```
-- artist_rank_top5 테이블 생성  
create table artist_rank_top5 (  
    ranks int,  
    Top10_Share varchar(200),  
    singer_name VARCHAR(200),  
    man_share varchar(200),  
    woman_share varchar(200),  
    teenager VARCHAR(200),  
    twenty VARCHAR(200),  
    thirty VARCHAR(200),  
    forty VARCHAR(200),  
    fifty VARCHAR(200),  
    others VARCHAR(200)  
);
```



# MySQL Table Design

## Artist Rank Top 5

```
-- artist_rank_top50 테이블 생성  
create table artist_rank_top50 (  
    ranks int,  
    singer_name varchar(200),  
    fans VARCHAR(300),  
    detail_score_music float,  
    detail_scoreAscending_fans float,  
    detail_score_likes float,  
    detail_score_photo float,  
    detail_score_video float  
);
```

## Artist Rank Top 50

```
-- circle_digital_chart_top100 테이블 생성  
create table circle_digital_chart_top100 (  
    ranks int,  
    song_name varchar(100),  
    singer_name varchar(200),  
    production varchar(100),  
    year int  
);
```



# 질문 1. 2012~2022 top100 중 가장 많이 순위권 든 노래 top15, 가수 top15

## 코드

```
-- top100 중 2012~2022 가장 많이 순위권에 든 노래 top15
select singer_name 가수명, song_name 곡명, count(song_name) 횟수, likes 좋아요
from top100
group by song_name
order by 3 desc
limit 15;
```

```
-- top100 중 2012~2022 가장 많이 순위권에 든 가수 top15
select singer_name 가수명, count(singer_name) 횟수
from top100
group by singer_name
order by 2 desc
limit 15;
```



# 질문 1. 2012~2022 top100 중 가장 많이 순위권 든 노래 top15, 가수 top15

## 결과

가수명	곡명	횟수	좋아요
방탄소년단	봄날	6	546,918
버스커 버스커	빛꽃 엔딩	5	348,436
풀킴	모든 날, 모든 순간 (Every day, Every Moment)	5	408,593
풀킴	너를 만나	4	312,318
AKMU (악뮤)	어떻게 이별까지 사랑하겠어, 널 사랑하는 거지	4	394,623
효린	안녕	4	86,796
잔나비	주저하는 연인들을 위해	3	352,267
장범준	흔들리는 꽃들 속에서 네 샴푸향이 느껴진거야	3	307,511
아이유	Love poem	3	335,021
아이유	밤편지	3	451,310
풀킴	비	3	298,048
방탄소년단	작은 것들을 위한 시 (Boy With Luv) (Feat. Hal...)	3	439,125
박효신	아생화	3	326,576
마크툽 (MAK...)	오늘도 빛나는 너에게 (To You My Light) (Feat....)	3	270,591
나얼	바람기억	3	222,394

가수명	곡수
아이유	43
방탄소년단	38
볼빨간사춘기	19
TWICE (트와이스)	18
BIGBANG (빅뱅)	17
다비치	17
버스커 버스커	16
풀킴	16
BLACKPINK	14
허각	14
AKMU (악뮤)	14
임영웅	14
Red Velvet (레드...)	13
태연 (TAEYEON)	12
지코 (ZICO)	12



## 질문 2. 플랫폼별 2012~2022 1위곡 비교(melon, circle chart)

### 코드

```
create view melon_v as -- 멜론차트 뷰만들기
with top_rank as
(
select years, ranks, song_name
from top100
where ranks = 1
group by 1, 2
order by 1, 2
)
select ranks,
       case when years = 2012 then song_name else 0 end year_2012,
       case when years = 2013 then song_name else 0 end year_2013,
       case when years = 2014 then song_name else 0 end year_2014,
       case when years = 2015 then song_name else 0 end year_2015,
       case when years = 2016 then song_name else 0 end year_2016,
       case when years = 2017 then song_name else 0 end year_2017,
       case when years = 2018 then song_name else 0 end year_2018,
       case when years = 2019 then song_name else 0 end year_2019,
       case when years = 2020 then song_name else 0 end year_2020,
       case when years = 2021 then song_name else 0 end year_2021,
       case when years = 2022 then song_name else 0 end year_2022
from top_rank;
```

```
create view circle_v as -- 써클차트 뷰 만들기
with top_rank as
(
select years, ranks, song_name
from circle_digital_chart_top100
where ranks = 1
group by 1, 2
order by 1, 2
)
select ranks,
       case when years = 2012 then song_name else 0 end year_2012,
       case when years = 2013 then song_name else 0 end year_2013,
       case when years = 2014 then song_name else 0 end year_2014,
       case when years = 2015 then song_name else 0 end year_2015,
       case when years = 2016 then song_name else 0 end year_2016,
       case when years = 2017 then song_name else 0 end year_2017,
       case when years = 2018 then song_name else 0 end year_2018,
       case when years = 2019 then song_name else 0 end year_2019,
       case when years = 2020 then song_name else 0 end year_2020,
       case when years = 2021 then song_name else 0 end year_2021,
       case when years = 2022 then song_name else 0 end year_2022
from top_rank
group by years;
select * from melon_v;
select ranks 순위, '멜론차트' 차트종류, -- union으로 이어주기 , max로 값이 1인 값만 불러오기
max(year_2012) '2012년', max(year_2013) '2013년', max(year_2014) '2014년', max(year_2015) '2015년',
max(year_2016) '2016년', max(year_2017) '2017년', max(year_2018) '2018년', max(year_2019) '2019년',
max(year_2020) '2020년', max(year_2021) '2021년', max(year_2022) '2022년' from melon_v
union
select ranks 순위, '써클차트' 차트종류,
max(year_2012) '2012년', max(year_2013) '2013년', max(year_2014) '2014년', max(year_2015) '2015년',
max(year_2016) '2016년', max(year_2017) '2017년', max(year_2018) '2018년', max(year_2019) '2019년',
max(year_2020) '2020년', max(year_2021) '2021년', max(year_2022) '2022년' from circle_v;
```



# 그 해의 1위 곡은~~~~

## 결과

순위	차트 종류	2012년	2013년	2014년	2015년	2016년	2017년	2018년	2019년	▼	2020년	2021년	2022년
1	멜론차트	강남스타일	자니 (Feat. ...	썸 (Feat. ...	뱅뱅뱅 (BA... CHEER UP	뱅뱅뱅 (BA... CHEER UP	첫눈처럼 너에게 가겠다	사랑을 했다...	사랑에 연습이 있었다면...	아무노래	Celebrity	Nxde	
1	서클차트	강남스타일	Gentleman	썸 (Feat. ...	뱅뱅뱅 (BA... CHEER UP	뱅뱅뱅 (BA... CHEER UP	첫눈처럼 너에게 가겠다	사랑을 했다...	2002	아무노래	Celebrity	Nxde	

Melon



CIRCLE CHART





## 질문 3. top3 기획사 중, 가장 많이 이름을 올린 가수 top15

### 코드

```
-- circle_digital_chart_top100 중 기획사 많은 순 top15
select production 기획사, count(production) 칫수
from circle_digital_chart_top100
group by production
order by count(production) desc
limit 15;
```

```
-- 기획사 많은 top3에 속하는 가수들 top15
select production, count(production) 칫수, singer_name
from circle_digital_chart_top100
where production = 'YG Entertainment'
      or production = 'Stone Music Entertainment'
      or production = 'SM Entertainment'
group by singer_name
order by count(production) desc
limit 15;
```



# 분발해라 제왑삐

## 결과

기획사	횟수
YG Entertainment	88
Stone Music Entertai	
SM Entertainment	
JYP Entertainment	
SC / HYBE	
소파-한터	
플레이엠엔터	
아메바컬쳐	
EDAM엔터테인먼트	
브랜뉴뮤직	
SBS 콘텐츠허브	
MBC	





## 질문 4. 가장 많이 차트 인한 가수 top5 중, 연도별 차트에 가장 많이 올라간 가수와 그들의 총 곡 수

### 코드

```
-- 가장 많이 차트 인한 가수 top5
select singer_name, count(singer_name)
from top100
group by singer_name
order by count(singer_name) desc
limit 5;
```

- 각 연도별로 가장 많이 차트에 올라간 가수

```
select years, singer_name, count(singer_name) 곡수
from top100
where singer_name = '아이유' or singer_name = '방탄소년단'
      or singer_name = '블빨간사춘기' or singer_name = 'TWICE (트와이스)'
      or singer_name = '다비치'
group by years
order by 1 asc;
```



# 그 해의 가수 - 국내에선 아이유가 짱!!!

## 결과

가수명	횟수	years	singer_name	총곡수
아이유	43	2012	아이유	5
방탄소년단	38	2013	다비치	6
볼빨간사춘기	19	2014	아이유	5
TWICE (트와이스)	18	2015	다비치	6
다비치	17	2016	TWICE (트와이스)	9
		2017	아이유	22
		2018	볼빨간사춘기	16
		2019	방탄소년단	16
		2020	아이유	20
		2021	아이유	19
		2022	볼빨간사춘기	11



5관왕



1관왕



2관왕



1관왕



2관왕



## 질문 5. 질문 4에 속하면서 현재 가장 인기있는 아티스트 top5에도 속하는 가수의 팬 남녀, 연령대 비율

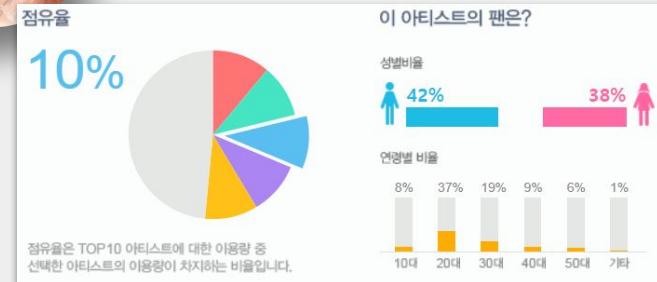
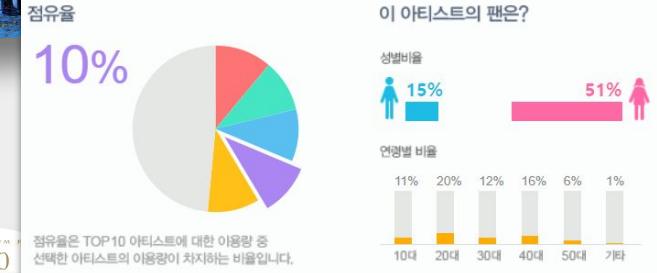
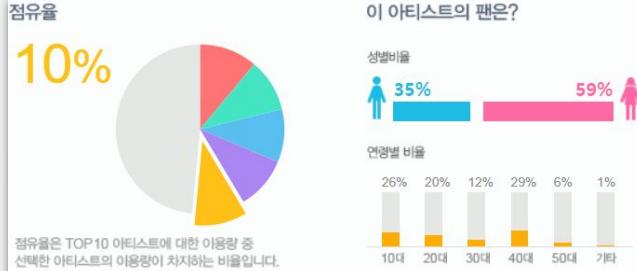
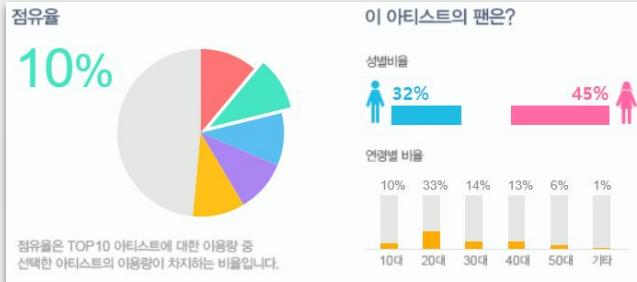
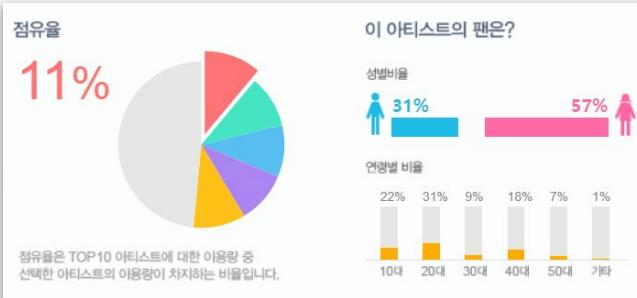
### 코드

```
-- 현재 가장 인기있는 아티스트 top5의 팬 남녀 비율, 연령대 비율  
select * from artist_rank_top5;
```

### 결과

ranks	Top10_Share	singer_name	man_share	woman_share	teenager	twenty	thirty	forty	fifty	others
1	11%	(여자)아이들	31%	57%	22%	31%	9%	18%	7%	1%
2	10%	아이유	32%	45%	10%	33%	14%	13%	6%	1%
3	10%	IVE (아이브)	35%	59%	26%	20%	12%	29%	6%	1%
4	10%	방탄소년단	15%	51%	11%	20%	12%	16%	6%	1%
5	10%	윤하 (YOUNHA)	42%	38%	8%	37%	19%	9%	6%	1%





## 질문 5. 질문 4에 속하면서 현재 가장 인기있는 아티스트 top5에도 속하는 가수의 팬 남녀, 연령대 비율

### 코드

```
역대 가장 많이 차트 인 한 가수 중에 현재 top5인 아티스트의 팬 남녀, 연령대 비율
select a.singer_name, count(a.singer_name) 차트인횟수,
       b.man_share, b.woman_share,
       b.teenager, b.twenty, b.thirty, b.forty, b.fifty, b.others
  from top100 a
 left join artist_rank_top5 b
    on a.singer_name = b.singer_name
 group by singer_name
 order by count(singer_name) desc
 limit 2;
```



# 국힙 TOP '아이유' 세계 TOP 'BTS'

## 결과값

singer_name	차트인 횟수	man_share	woman_share	teenager	twenty	thirty	forty	fifty	others
아이유	43	32%	45%	10%	33%	14%	13%	6%	1%
방탄소년단	38	15%	51%	11%	20%	12%	16%	6%	1%





## 질문 6. 아티스트 랭킹 1~50위에 속하며, top100에도 속하는 가수들의 차트 인 곡수와 종합 세부점수

### 코드

```
-- 아티스트 랭킹 1~50위에 속하며 top100에도 속하는 가수들의 차트 인 곡수와 종합 세부점수  
select a.ranks 'artist rank', b.singer_name 가수명, a.fans, count(b.song_name) 차트인곡수,  
round(detail_score_music + detail_scoreAscending_fans + detail_score_likes + detail_score_photo + detail_score_video, 2) '세부점수 총점 50점'  
from artist_rank_top50 a  
inner join top100 b  
on a.singer_name = b.singer_name  
group by a.singer_name  
order by a.ranks;
```

- 아티스트 랭킹 50위안에 속하나 차트 순위 top100위안에 없다면 제외되어야합니다.



# 꿈나무 (여자)아이들 아이유를 제치다~~~!!!!

artist rank	가수명	fans	차트인곡수	세부점수 총점 50점	artist rank	가수명	fans	차트인곡수	세부점수 총점 50점	
1	(여자)아이들	팬 120,337명 (+ 350)	5	46.4	21	결과값	Crush	팬 213,929명 (+ 62)	8	31.2
2	아이유	팬 775,704명 (+ 267)	43	43.2	22	지코 (ZICO)	팬 185,568명 (+ 43)	12	30.9	
3	IVE (아이브)	팬 61,665명 (+ 319)	3	45.2	23	경서	팬 13,708명 (+ 56)	4	31.3	
4	방탄소년단	팬 739,871명 (+ 309)	38	42.8	25	세부점수 측정기준(여자아이들기준)	팬 10,700명 (+ 56)	8	29.9	
5	윤하 (YOUNHA)	팬 110,333명 (+ 296)	2	42.4	26	플컴	팬 159,790명 (+ 31)	16	29.6	
6	LE SSERAFIM (르세라핌)	팬 25,252명 (+ 460)	2	41.8	27	음원 9.2	하이라이트 (Highlight)	팬 82,642명 (+ 41)	1	29.3
7	임영웅	팬 123,855명 (+ 119)	14	38.7	28	BIGBANG (빅뱅)	팬 264,312명 (+ 53)	17	28.4	
8	NewJeans	팬 55,844명 (+ 317)	3	40.1	31	팬증가수 9.7	Justin Bieber	팬 89,532명 (+ 27)	1	27.8
9	BLACKPINK	팬 236,175명 (+ 102)	14	37.7	32	영탁	팬 34,841명 (+ 11)	1	33.3	
11	잔나비	팬 136,846명 (+ 189)	5	35.7	35	좋아요 8.1	이무진	팬 55,556명 (+ 18)	8	29.0
12	정은지	팬 59,016명 (+ 145)	1	37.8	36	진	팬 133,352명 (+ 21)	8	29.0	
13	BIG Naughty (서동현)	팬 60,104명 (+ 130)	1	35.2	37	멜로망스	팬 85,527명 (+ 43)	8	29.0	
14	10CM	팬 180,157명 (+ 124)	8	33.4	38	포토 9.7	TWICE (트와이스)	팬 205,140명 (+ 22)	25	28.3
15	Charlie Puth	팬 158,215명 (+ 122)	4	32.1	39	성시경	팬 118,961명 (+ 20)	12	28.3	
16	태연 (TAEYEON)	팬 249,873명 (+ 110)	12	33.3	40	비디오 9.6	Red Velvet (레드벨벳)	팬 251,376명 (+ 23)	12	28.3
17	BE'0 (비오)	팬 52,777명 (+ 90)	2	33.1	41	작재	팬 43,996명 (+ 4)	1	28.5	
19	aespa	팬 88,881명 (+ 77)	5	32.8	43	세부점수 총점 46.4 / 50	팬 125,688명 (+ 7)	17	28.4	
20	볼빨간사춘기	팬 271,074명 (+ 62)	19	31.4	46	백예린 (Yerin Baek)	팬 176,598명 (+ 67)	5	26.4	
					48	박재범	팬 117,987명 (+ 36)	3	27	





# Thank you

10cm



# SHE

Knows  
where  
the  
rainbow



IVE 3RD SINGLE ALBUM

Afie