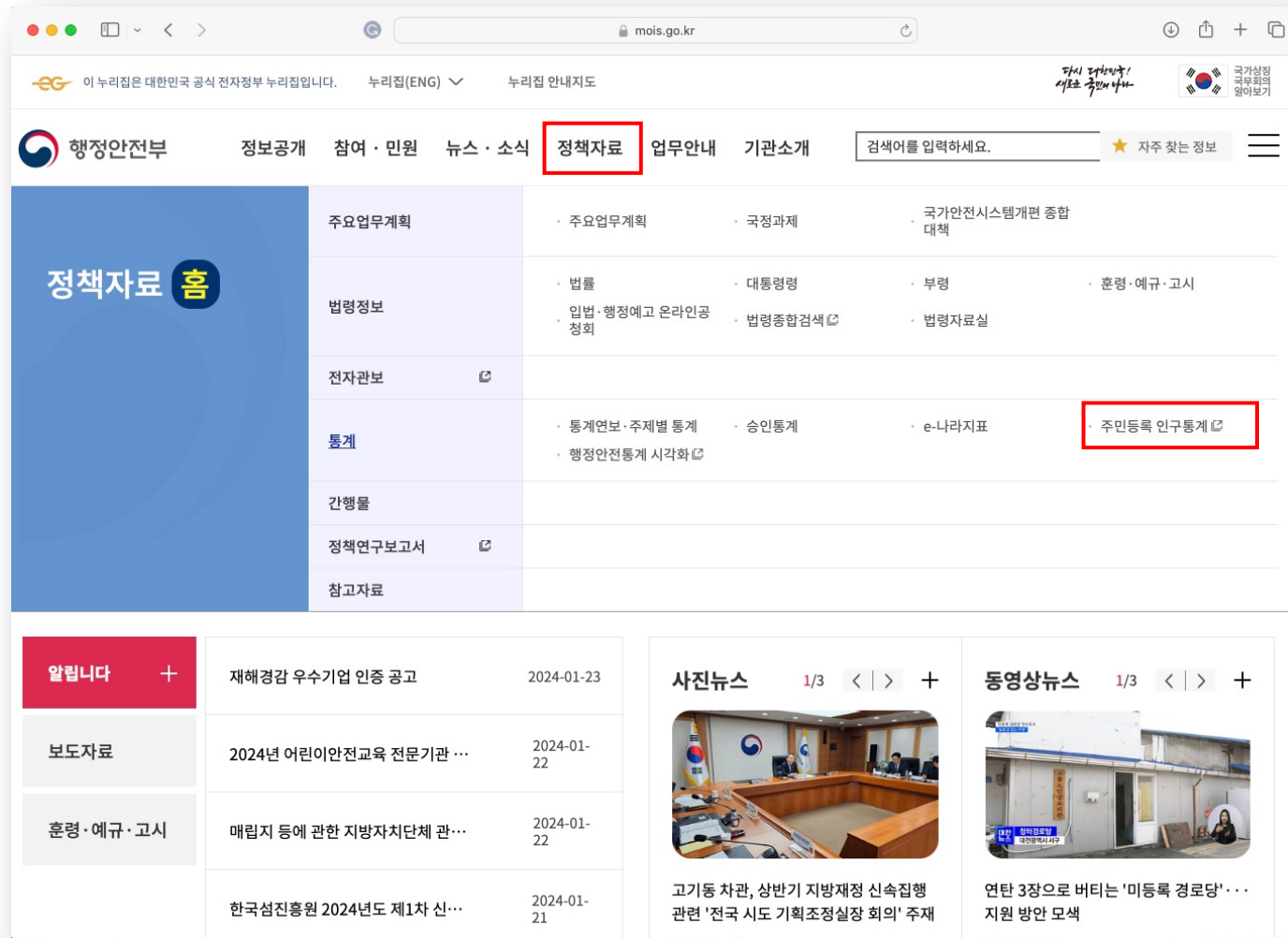


공공데이터 분석

인구 데이터 분석

인구 공공데이터 내려 받기 #1

- 행정안전부 홈페이지: <https://www.mois.go.kr>
 - 정책자료 > 주민등록 인구 통계 메뉴 선택



인구 공공데이터 내려 받기 #2

- 주민등록 인구통계
 - 연령별 인구현황 선택

행정안전부

주민등록 인구통계

주민등록 인구 및 세대현황

연령별 인구현황

주민등록 인구 기타현황

연령별 인구현황

통계표

그래프

행정구역: 전국, 시·군·구

등록구분: 전체

조회기간: 월간, 연간, 2023년 12월 ~ 2023년 12월

*매월 말일 작성/공표일시: 매월 1일 12시 이후(공표일이 주말, 공휴일인 경우에는 다음 평일에 공표)

구분: ☒ 계, ☐ 남·여 구분

연령 구분 단위: 1세

만 연령구분: 0, 100이상

검색, 초기화

☒ 현재화면, ☐ 전체시군구현황, ☐ 전체읍면동현황

csv 파일 다운로드, xlsx 파일 다운로드

인구 공공데이터 내려 받기: 연령별 인구 현황 (남여 구분 없음)

■ 연령별 인구 현황

- 조회기간
 - 월간 2023년 12월 ~ 2023년 12월
- 구분: 남·여구분 없음
- 연령 구분 단위: 1세
- 만 연령구분: 0 ~ 100이상
- 전체 읍면동 현황 선택
- 검색 버튼 클릭
- xlsx 파일 다운로드
 - age.xlsx 로 저장 후 age.csv로 변환

The screenshot shows the '연령별 인구현황' (Age Population Status) page on the 'jumin.mois.go.kr' website. The page has a sidebar with '주민등록 인구통계' (Residential Registration Population Statistics) and a main content area with '연령별 인구현황' (Age Population Status). The main area has tabs for '통계표' (Table) and '그래프' (Graph). The '통계표' tab is active, showing search filters. The filters include '행정구역' (Administrative Area) set to '전국' (All Korea), '등록구분' (Registration Category) set to '전체' (All), '조회기간' (Search Period) set to '월간' (Monthly) for '2023년 12월 ~ 2023년 12월', '구분' (Category) set to '계' (Total), '연령 구분 단위' (Age Unit) set to '1세' (1 year), and '만 연령구분' (Age Group) set to '0 ~ 100이상'. At the bottom, there are buttons for '검색' (Search), '초기화' (Reset), and two download buttons: 'csv 파일 다운로드' (Download CSV File) and 'xlsx 파일 다운로드' (Download XLSX File). Red boxes and numbers 1 through 6 highlight specific elements: 1. '전체' (All) under '등록구분'; 2. '계' (Total) under '구분'; 3. '1세' (1 year) under '연령 구분 단위'; 4. '0' (0) under '만 연령구분'; 5. '전체읍면동현황' (All Eupmyeondong Status) under '현재화면'; 6. 'xlsx 파일 다운로드' (Download XLSX File) button.

인구 공공데이터 내려 받기: 연령별 인구 현황 (남여 구분)

■ 연령별 성별 인구현황

- 조회기간:
 - 월간 2023년 12월 ~ 2023년 12월
- 구분: 남·여구분 선택
- 연령 구분 단위: 1세
- 만 연령구분: 0 ~ 100이상
- 전체 시군구 현황 선택
- csv 파일 다운로드
 - gender.xlsx 로 저장

행정안전부

주민등록 인구통계

연령별 인구현황

통계표 | 그래프

행정구역: 전국 | 시·군·구: [선택]

등록구분: 전체

조회기간: 월간 | 연간 | 2023년 12월 ~ 2023년 12월

구분: ☒ 계 ☒ 남·여 구분

연령 구분 단위: 1세

만 연령구분: 0 ~ 100이상

검색 | 초기화

☐ 현재화면 ☒ 전체시군구현황 ☐ 전체읍면동현황

☒ csv 파일 다운로드 ☒ xlsx 파일 다운로드

전국 읍면동, 연령별 인구 현황

- age.csv 파일:
 - 데이터에 남·여 구분 없음
- 헤더 정보
 - A열: 행정기관 코드
 - B열: 행정기관
 - C~D열: 총 인구수 및 연령구간 인구수
 - E열 ~ CZ열: 연령별 인구 수(0세 ... 99세, 100세 이상)

	A	B	C	D	E	F	G	H	I	J	K	L	M	연령별 인구수			
1	※ 매월 말일자 통계 현황																
2			2023년12월														
3			계														
4	행정기관코드	행정기관	총 인구수	연령구간인구	0세	1세	2세	3세	4세	5세	6세	7세	8세	9세	10세	11세	12세
5	1100000000	서울특별시	9,386,034	9,386,034	37,771	41,087	43,013	43,637	47,506	50,291	55,100	62,036	66,691	66,430	66,672	74,287	71,911
6	1111000000	서울특별시 종로구	139,417	139,417	390	460	479	470	543	552	655	726	834	810	819	937	891
7	1111051500	서울특별시 종로구 청운효자동	11,349	11,349	40	44	44	48	51	50	68	84	87	80	90	120	98
8	1111053000	서울특별시 종로구 사직동	8,998	8,998	24	27	45	32	46	47	51	57	66	67	67	62	65
9	1111054000	서울특별시 종로구 삼청동	2,248	2,248	6	5	7	5	5	11	14	7	16	10	13	16	17
10	1111055000	서울특별시 종로구 부암동	9,085	9,085	18	21	29	20	30	36	48	57	65	57	70	84	65
11	1111056000	서울특별시 종로구 평창동	17,387	17,387	63	65	90	84	95	90	108	110	143	148	148	130	152
12	1111057000	서울특별시 종로구 무악동	7,961	7,961	28	33	45	49	38	52	71	79	72	79	75	84	109
13	1111058000	서울특별시 종로구 교남동	9,715	9,715	51	60	65	56	96	76	82	90	101	64	84	93	56
14	1111060000	서울특별시 종로구 가회동	3,860	3,860	12	13	13	7	13	15	16	21	23	15	26	23	46
15	1111061500	서울특별시 종로구 종로1.2.3.4가	6,683	6,683	11	21	16	19	16	17	5	16	26	23	14	26	22
16	1111063000	서울특별시 종로구 종로5.6가동	5,384	5,384	6	10	6	17	14	18	24	20	22	19	14	19	17
17	1111064000	서울특별시 종로구 이화동	6,899	6,899	11	16	12	14	22	12	15	21	20	28	24	36	23
18	1111065000	서울특별시 종로구 혜화동	15,869	15,869	37	41	33	34	45	40	60	56	78	88	82	93	100
19	1111067000	서울특별시 종로구 창신제1동	4,521	4,521	12	9	11	6	11	9	7	10	17	15	13	16	16
20	1111068000	서울특별시 종로구 창신제2동	7,510	7,510	11	16	9	16	10	20	17	15	21	22	21	25	35

연령별 인구수

대구 산격동 인구 현황

```
import csv
```

<day03_age_01.py>

```
f = open('age.csv', encoding='utf-8-sig')
```

```
data = csv.reader(f)
```

```
header = next(data)
```

```
print(header)
```

```
# row[0]: 행정기관
```

```
for row in data:
```

```
    if '산격3동' in row[0]: # '산격3동'이 포함된 자료만 출력
```

```
        print(row)
```

```
f.close()
```

```
['행정기관', '총 인구수', '연령구간인구수', '0세', '1세', '2세', '3세', '4세', '5세', '6세', '7세', '8세', '9세', '10세',  
'11세', '12세', '13세', '14세', '15세', '16세', '17세', '18세', '19세', '20세', '21세', '22세', '23세', '24세', '25세', '26세', '27세', '28세', '29세', '30세',  
'31세', '32세', '33세', '34세', '35세', '36세', '37세', '38세', '39세', '40세', '41세', '42세', '43세', '44세', '45세', '46세',  
'47세', '48세', '49세', '50세', '51세', '52세', '53세', '54세', '55세', '56세', '57세', '58세', '59세', '60세', '61세', '62세', '63세', '64세', '65세', '66세',  
'67세', '68세', '69세', '70세', '71세', '72세', '73세', '74세', '75세', '76세', '77세', '78세', '79세', '80세', '81세', '82세', '83세', '84세', '85세', '86세',  
'87세', '88세', '89세', '90세', '91세', '92세', '93세', '94세', '95세', '96세', '97세', '98세', '99세', '100세 이상']  
['대구광역시 북구 산격3동', '8,750', '8,750', '13', '13', '5', '15', '15', '22', '15', '22', '27', '25', '23', '28', '28', '35',  
'33', '26', '43', '31', '37', '99', '190', '207', '268', '379', '348', '7', '260', '223', '205', '173', '172', '135', '121', '99',  
'93', '64', '72', '73', '65', '59', '86', '86', '92', '116', '69', '73', '86', '97', '123', '96', '106', '90', '107', '119', '101',  
'118', '105', '120', '113', '112', '137', '143', '146', '122', '123', '131', '109', '123', '111', '97', '116', '81', '77', '92',  
'92', '89', '48', '45', '60', '57', '62', '45', '52', '44', '26', '36', '33', '23', '17', '7', '11', '8', '11', '2', '7', '2', '0',  
'2', '0', '0']
```

모든 데이터는 문자열
형태로 저장되어 있음

인구수 출력

- age.csv 데이터 header (연령 구분: 1세)

[0]	[1]	[2]	[3]	[4]	...	[102]	[103]
행정기관	총 인구수	연령 구간 인구수	0세	1세	...	99세	100세 이상

- 산격3동 (경북대 인근)의 인구 데이터 출력

```
import csv <day03_age_02.py>

f = open('age.csv', encoding='utf-8-sig')
data = csv.reader(f)
header = next(data)

result = []
for row in data:
    if '산격3동' in row[0]: # '산격3동'이 포함된 자료만 출력
        for data in row[3:]: # 0세 ~ 100세 이상까지 자료를 리스트에 추가
            result.append(data)
print(result)
f.close()
```

```
['13', '13', '5', '15', '15', '22', '15', '22', '27', '25', '23', '28', '28', '35', '33', '26', '43', '31', '37', '99', '190', '207', '268',
'379', '348', '340', '247', '260', '223', '205', '173', '172', '135', '121', '99', '93', '64', '72', '73', '65', '59', '86', '86', '92', '116',
'69', '73', '86', '97', '123', '96', '106', '90', '107', '119', '101', '118', '105', '120', '113', '112', '137', '143', '146', '122', '123',
'131', '109', '123', '111', '97', '116', '81', '77', '92', '92', '89', '48', '45', '60', '57', '62', '45', '52', '44', '26', '36', '33', '23',
'17', '7', '11', '8', '11', '2', '7', '2', '0', '2', '0', '0']
```


대구시 산격3동의 인구 분포 그래프 그리기

```
import csv
import matplotlib.pyplot as plt
import platform
import koreanize_matplotlib
```

```
f = open('age.csv', encoding='utf-8-sig')
data = csv.reader(f)
result = []
city = ''
for row in data:
    if '산격3' in row[0]:
        city = row[0]
        for data in row[3:]: # 0세부터 100세 이상까지 데이터
            if ',' in data:
                data = data.replace(',', '')
            result.append(int(data)) # 숫자로 변환

f.close()
print(result)
```

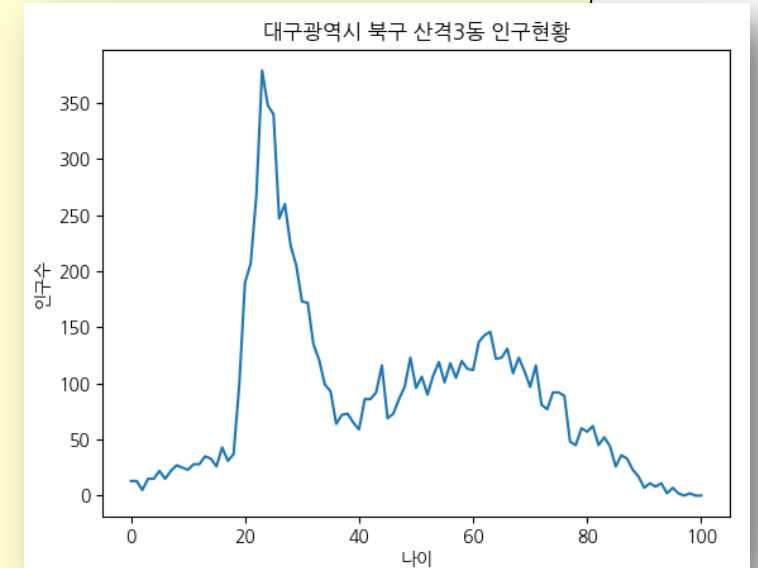
```
plt.title(f'{city} 인구현황')
plt.xlabel('나이')
plt.ylabel('인구수')
plt.plot(result)
plt.show()
```

<day03_age_03.py>

[0]	[1]	[2]	[3]	[4]	...	[102]	[103]
행정기관	총 인구수	연령 구간 인구수	0세	1세	...	99세	100세 이상

if ',' in data:
data = data.replace(',', '')
result.append(int(data)) # 숫자로 변환

천 단위 콤마 삭제 및
숫자로 변환



```
[13, 13, 5, 15, 15, 22, 15, 22, 27, 25, 23, 28, 28, 35, 33, 26, 43, 31, 37, 99, 190, 207, 268, 379, 348, 340,
247, 260, 223, 205, 173, 172, 135, 121, 99, 93, 64, 72, 73, 65, 59, 86, 86, 92, 116, 69, 73, 86, 97, 123, 96,
106, 90, 107, 119, 101, 118, 105, 120, 113, 112, 137, 143, 146, 122, 123, 131, 109, 123, 111, 97, 116, 81, 77,
92, 92, 89, 48, 45, 60, 57, 62, 45, 52, 44, 26, 36, 33, 23, 17, 7, 11, 8, 11, 2, 7, 2, 0, 2, 0, 0]
```

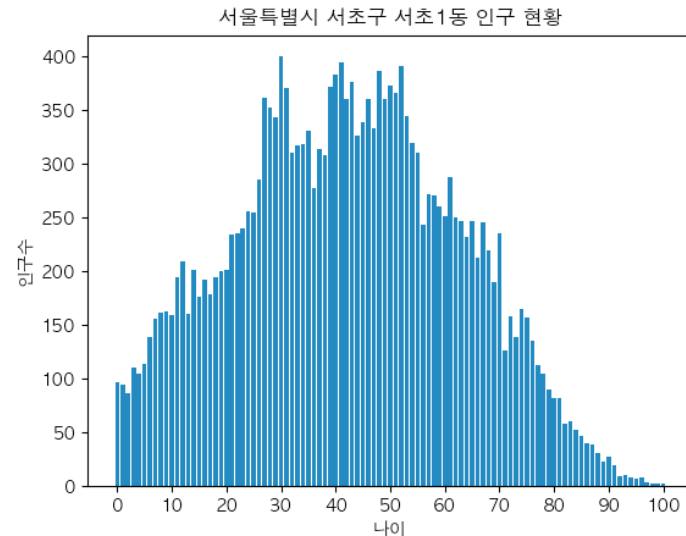
실습: 인구 구조 그래프 함수 구현

- 동 이름을 입력하면 해당 동의 인구 분포를 그리는 함수 구현

인구 구조를 알고 싶은 지역의 이름(읍면동 단위)을 입력하세요: 서초1

인구 구조를 알고 싶은 지역의 이름(읍면동 단위)을 입력하세요: 서초1

0세:	81명	1세:	95명	2세:	93명	3세:	98명	4세:	105명	5세:	92명	6세:	1036명	8세:	154명	9세:	159명
10세:	173명	11세:	154명	12세:	201명	13세:	200명	14세:	179명	15세:	193명	16세:	1781명	18세:	187명	19세:	214명
20세:	198명	21세:	186명	22세:	246명	23세:	224명	24세:	264명	25세:	263명	26세:	2718명	28세:	349명	29세:	335명
30세:	356명	31세:	363명	32세:	379명	33세:	293명	34세:	317명	35세:	326명	36세:	3495명	38세:	299명	39세:	301명
40세:	365명	41세:	378명	42세:	384명	43세:	367명	44세:	358명	45세:	312명	46세:	3540명	48세:	342명	49세:	369명
50세:	349명	51세:	393명	52세:	371명	53세:	355명	54세:	330명	55세:	321명	56세:	2935명	58세:	263명	59세:	259명
60세:	256명	61세:	257명	62세:	276명	63세:	238명	64세:	236명	65세:	229명	66세:	2512명	68세:	244명	69세:	206명
70세:	182명	71세:	215명	72세:	136명	73세:	144명	74세:	139명	75세:	160명	76세:	1626명	78세:	103명	79세:	93명
80세:	89명	81세:	80명	82세:	80명	83세:	60명	84세:	60명	85세:	46명	86세:	434명	88세:	34명	89세:	29명
90세:	15명	91세:	23명	92세:	14명	93세:	8명	94세:	9명	95세:	3명	96세:	6명	98세:	5명	99세:	1명
100세:	3명																



인구 구조 그래프 함수 구현 #1

<day03_age_04.py>

```
import csv
import matplotlib.pyplot as plt
import platform
import matplotlib.font_manager as fm

def print_population(population):
    '''
    특정 지역의 인구 현황을 화면에 출력함
    '''
    for i in range(len(population)):
        print(f'{i:3d}세: {population[i]:6d}명', end=' ')
        if (i + 1) % 10 == 0:
            print()
```

```
def draw_population(district_name, population_list):
    '''
    특정 지역에 대한 인구 분포를 그래프로 나타냄(plot)
    - district_name: 지역 이름
    - population_list: 0~100세 이상까지 인구수 리스트
    '''
    # 그래프 출력
    plt.style.use('ggplot')
    plt.title('{} 인구 현황'.format(district_name))
    plt.xlabel('나이')
    plt.ylabel('인구수')

    plt.bar(range(101), population_list)
    plt.xticks(range(0, 101, 10)) # 0세 ~ 100세 이상

    plt.plot(population_list)
    plt.show()
```

인구 구조 그래프 함수 구현 #2

```
def get_population(city):  
    f = open('age.csv', encoding='utf-8-sig')  
    data = csv.reader(f)  
    next(data) # 헤더 정보 건너뛰
```

<day03_age_04.py>

[0]	[1]	[2]	[3]	[4]	...	[102]	[103]
행정기관	총 인구수	연령 구간 인구수	0세	1세	...	99세	100세 이상

```
population_list = []  
district_name = ''  
for row in data:
```

```
    if city in row[0]:
```

```
        district_name = row[0]
```

```
        for data in row[3:]:
```

```
            if ',' in data:
```

```
                data = data.replace(',', '') # 천 단위 콤마 제거
```

```
                population_list.append(int(data))
```

```
            break # 처음으로 일치하는 도시명만 검색하기 위함
```

```
f.close()
```

```
print_population(population_list)
```

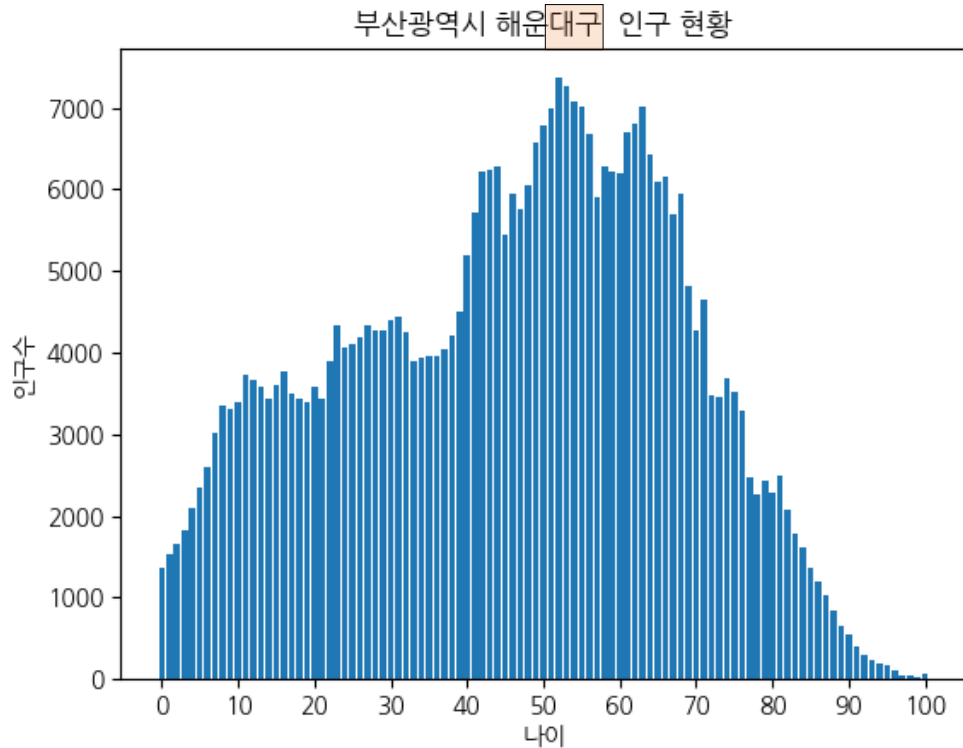
```
draw_population(district_name, population_list)
```

```
city = input('인구 구조를 알고 싶은 지역의 이름(읍면동 단위)을 입력하세요: ')  
get_population(city)
```

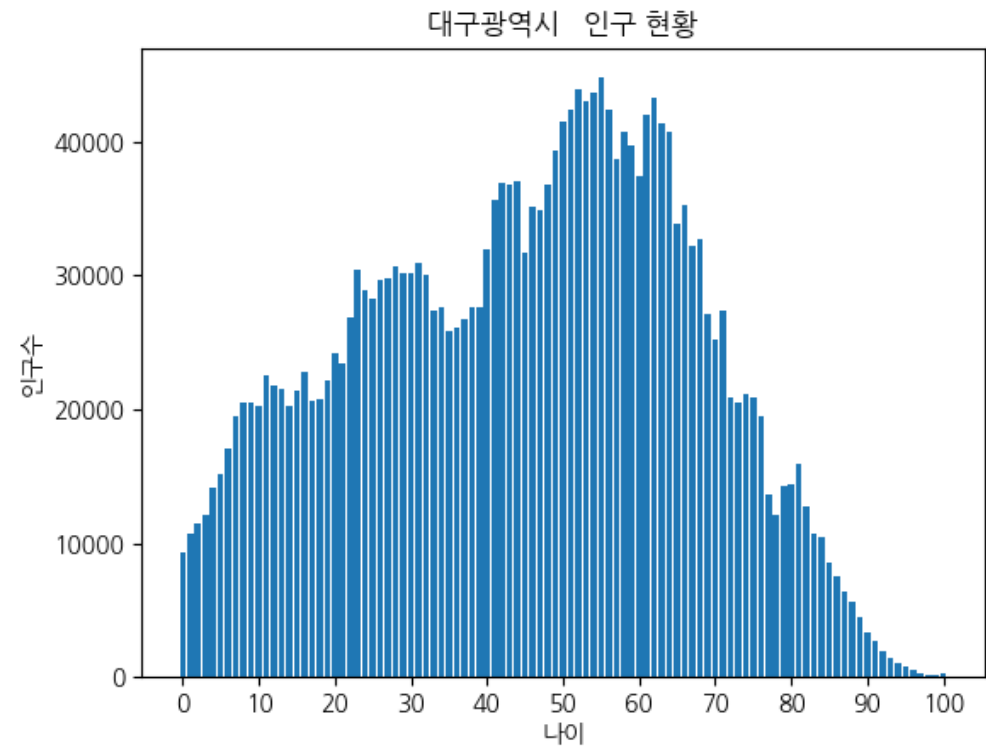
row[0]에 입력한 도시 정보를
포함하고 있는 경우

막대 그래프: 읍면동 입력 #2 (실행 결과)

인구 구조를 알고 싶은 지역의 이름(읍면동 단위)을 입력하세요: 대구



‘대구광역시’ 입력



투표 가능 인구 비율 분석 #1

■ 대구 광역시 투표 가능 인구 분석

- 만 18세 이상 ~
- csv 파일에서 처음으로 ‘**대구광역시**’가 나오는 인구수 활용: 전체 대구 인구수 정보 포함

676	대구광역시	2,374,960	2,374,960	9,288	10,675	11,476	12,056	14,092	15,212	17,059	19,530	20,557	20,458	20,203	22,532	21,803	21,565
677	대구광역시 중구	89,064	89,064	560	648	606	596	701	688	742	806	756	706	627	690	643	582
678	대구광역시 중구 동인동	8,372	8,372	39	38	32	22	27	24	24	34	29	33	47	40	47	51
679	대구광역시 중구 삼덕동	6,643	6,643	33	29	22	33	28	26	36	49	48	46	46	51	59	53
680	대구광역시 중구 성내1동	4,868	4,868	4	4	6	10	10	12	10	16	11	15	12	24	11	18
681	대구광역시 중구 성내2동	5,872	5,872	18	31	39	36	51	29	34	43	24	28	23	28	28	24
682	대구광역시 중구 성내3동	9,030	9,030	97	156	124	115	123	127	135	116	116	104	86	90	50	47
683	대구광역시 중구 대신동	7,558	7,558	70	55	68	57	70	69	71	90	75	70	65	72	51	42
684	대구광역시 중구 남산1동	4,260	4,260	7	16	10	14	16	13	17	29	20	23	24	26	27	25
685	대구광역시 중구 남산2동	7,272	7,272	58	57	66	55	76	93	88	103	91	83	62	67	60	41
686	대구광역시 중구 남산3동	6,203	6,203	26	35	27	25	35	33	41	36	59	49	44	60	61	62
687	대구광역시 중구 남산4동	16,476	16,476	129	138	127	137	175	162	183	185	175	164	128	152	140	126
688	대구광역시 중구 대봉1동	7,823	7,823	35	44	52	49	47	60	57	68	73	57	59	54	84	65
689	대구광역시 중구 대봉2동	4,687	4,687	44	45	33	43	43	40	46	37	35	34	31	26	25	28
690	대구광역시 동구	342,813	342,813	1,509	1,745	1,856	1,866	2,179	2,289	2,384	2,693	2,762	2,717	2,636	2,879	2,667	2,665
691	대구광역시 동구 신암1동	10,945	10,945	53	87	97	89	96	103	98	100	100	82	64	60	57	47
692	대구광역시 동구 신암2동	11,562	11,562	93	97	82	102	95	106	120	132	122	107	97	110	101	104
693	대구광역시 동구 신암3동	10,917	10,917	90	122	121	106	122	126	125	118	131	119	88	89	61	60

- 그래프 출력
 - 전체 인구대비 투표 가능 인구 비율: pie chart

투표 가능 인구수 분석 #2

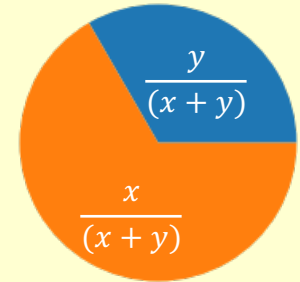
<day03_age_05.py>

```
import csv
import matplotlib.pyplot as plt
import koreanize_matplotlib

def draw_piechart(city_name, city_population, voting_population):
    '''
    전체 인구수 대비 투표 가능 인구의 파이차트 작성
    '''
    non_voting_population = city_population - voting_population
    population = [non_voting_population, voting_population]

    color = ['tomato', 'royalblue']
    plt.pie(population, labels=['18세 미만', '투표가능인구'],
            autopct='%1f%%', colors=color, startangle=90)

    plt.legend(loc=1)
    plt.title(city_name + " 투표 가능 인구 비율")
    plt.show()
```



[18세 미만 인구, 투표가능 인구]
를 pie chart에 전달

투표 가능 인구수 분석 #3

<day03_age_05.py>

```
def get_voting_population(city):  
    '''  
    투표 가능 인구수 분석 row[21:]  
    전체 인구수 : row[1]  
    '''  
  
    f = open('age.csv', encoding='utf-8-sig')  
    data = csv.reader(f)  
    header = next(data) # 헤더 정보 건너뛰  
  
    voting_number_list = []  
    city_name = ''  
    city_population = 0 # 도시 전체 인구수  
    voting_population = 0  
    for row in data:  
        if city in row[0]:  
            city_population = row[1]  
            if ',' in city_population:  
                # 도시 전체 인구수에서 천단위 콤마 제거  
                city_population = city_population.replace(',', '')  
                city_population = int(city_population)
```

```
        city_name = row[0]  
        for data in row[21:]: # 18세 이상  
            if ',' in data:  
                data = data.replace(',', '') # 천단위 콤마 제거  
            voting_num = int(data)  
            # 각 연령대별 투표 인구수를 리스트에 추가  
            voting_number_list.append(voting_num)  
            # 누적된 투표 가능 인구수  
            voting_population += voting_num
```

break

특정 도시의 데이터 중에서 제일 먼저 나오는 데이터만 분석하기 위함

```
f.close()  
print(f'{city_name}전체 인구수:{city_population:,}명,  
      투표 가능 인구수: {voting_population:,}명')  
  
draw_piechart(city_name, city_population, voting_population)
```

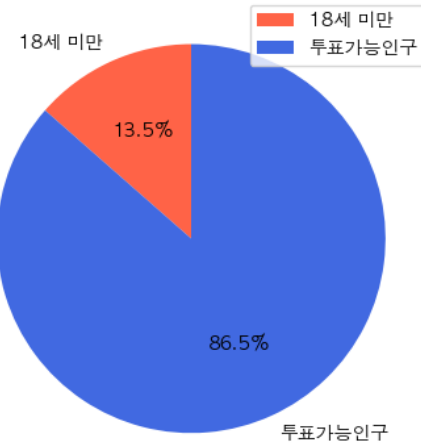
```
city = input('투표 가능 인구수를 확인할 도시이름을 입력하시오: ')  
get_voting_population(city)
```

[0]	[1]	[2]	[3]	[4]	...	[102]	[103]
행정기관	총 인구수	연령 구간 인구수	0세	1세	...	99세	100세 이상

투표 가능 인구수 분석: 실행 결과

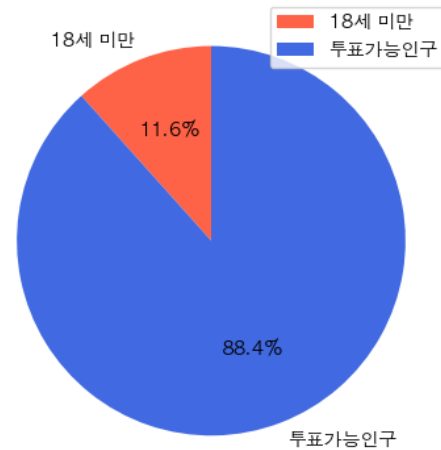
투표 가능 인구수를 확인할 도시이름을 입력하시오: **대구광역시**
대구광역시 전체 인구수: 2,374,960명, 투표 가능 인구수: 2,053,380명

대구광역시 투표 가능 인구 비율



투표 가능 인구수를 확인할 도시이름을 입력하시오: **서울**
서울특별시 전체 인구수: 9,386,034명, 투표 가능 인구수: 8,293,025명

서울특별시 투표 가능 인구 비율



학령 인구 비율

- 학령 인구: 6세 ~ 21세
 - 이론적 입학연령과 수학기간을 고려하여 정해지는 연령 집단
 - https://kostat.go.kr/board.es?mid=a10502020600&bid=3207&act=view&list_no=423716&tag=&nPage=1&ref_bid=
 - 초등학교: 6~11세
 - 중학교: 12~14세
 - 고등학교: 15~17세
 - 대학교: 18~21세

학령 인구 비율 예제 #1

```
import csv
import matplotlib.pyplot as plt
import koreanize_matplotlib

def draw_pie_chart(city, population_list, label_list):
    plt.pie(population_list, labels=label_list, autopct='%.1f%%', startangle=90)

    plt.legend(loc=1)
    plt.title(city + " 학령인구 비율")
    plt.show()

def get_population(row, start, end):
    population = 0
    for num in row[start:end+1]:
        if ',' in num:
            num = num.replace(',', '')
        num = int(num)
        population += num
    return population
```

학령 인구 비율 예제 #2

```
def school_age_population(city):
    city_population = 0
    non_school_pop = 0
    school_age_pop = 0

    label_list = ['초등학생', '중학생', '고등학생',
                  '대학생', '비학령인구']
    population_list = []

    f = open('age.csv', encoding='utf-8-sig')
    data = csv.reader(f)
    header = next(data) # 헤더 정보 건너뛰

    for row in data:
        if city in row[0]:
            city_population = row[1]
            if ',' in city_population:
                # 도시 전체 인구수에서 천단위 콤마 제거
                city_population = city_population.replace(',', '')
            city_population = int(city_population)

            # 각 구간별 인구 계산
            elementary_pop = get_population(row, 9, 14)
            population_list.append(elementary_pop)
```

```
            middle_school_pop = get_population(row, 15, 17)
            population_list.append(middle_school_pop)

            high_school_pop = get_population(row, 18, 20)
            population_list.append(high_school_pop)

            university_pop = get_population(row, 21, 24)
            population_list.append(university_pop)

            school_age_pop = (elementary_pop + middle_school_pop +
                              high_school_pop + university_pop)

            non_school_pop = city_population - school_age_pop
            population_list.append(non_school_pop)
            break

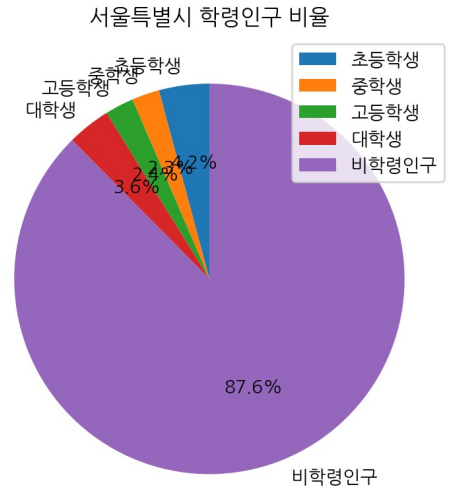
    print(f'전체인구수{city_population}
          학령인구 비율:{round((school_age_pop*100)/city_population, 1)}%')
    draw_pie_chart(city, population_list, label_list)

city = input("학령인구를 분석할 도시 이름: ")
school_age_population(city)
```

학령 인구 비율 예제 #3

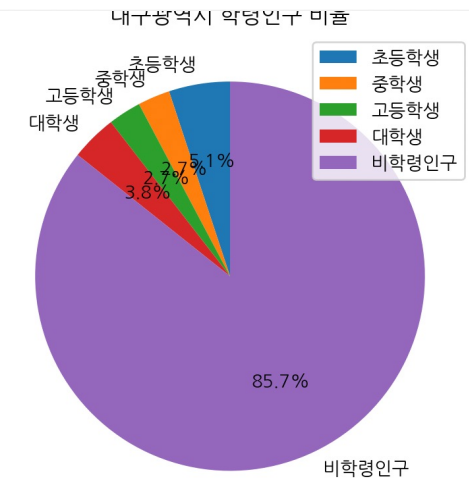
학령인구를 분석할 도시 이름: **서울특별시**

전체인구수: 9,386,034 학령인구수: 1,165,539 학령인구 비율: 12.4%



학령인구를 분석할 도시 이름: **대구광역시**

전체인구수: 2,374,960 학령인구수: 339,389 학령인구 비율: 14.3%



gender.csv 데이터 분석

- 성별 인구 현황
 - 전국 시군구 단위, 1세 단위

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q
1	행정기관	총 인구수	연령구간인구	0세	1세	2세	3세	4세	5세	6세	7세	8세	9세	10세	11세	12세	13세
2	서울특별시	9,386,034	9,386,034	37,771	41,087	43,013	43,637	47,506	50,291	55,100	62,036	66,691	66,430	66,672	74,287	71,911	72,773
3	서울특별시 종로구	139,417	139,417	390	460	479	470	543	552	655	726	834	810	819	937	891	994
4	서울특별시 중구	121,312	121,312	472	535	591	547	542	524	577	639	627	576	621	658	553	554
5	서울특별시 용산구	213,151	213,151	927	1,008	1,005	1,034	1,094	1,110	1,112	1,225	1,288	1,286	1,251	1,330	1,249	1,301
6	서울특별시 성동구	277,361	277,361	1,409	1,558	1,604	1,625	1,660	1,736	1,741	1,872	1,854	1,834	1,778	1,780	1,659	1,571
7	서울특별시 광진구	335,554	335,554	1,120	1,204	1,261	1,271	1,434	1,460	1,577	1,839	2,046	1,996	2,130	2,319	2,274	2,306
8	서울특별시 동대문구	341,149	341,149	1,390	1,497	1,549	1,682	1,822	1,814	1,996	2,133	2,249	2,170	2,094	2,346	2,155	2,165
9	서울특별시 중랑구	382,155	382,155	1,702	1,755	1,654	1,580	1,828	1,910	1,973	2,248	2,326	2,138	2,269	2,450	2,324	2,290
10	서울특별시 성북구	425,602	425,602	1,696	1,878	2,008	1,988	2,226	2,390	2,594	2,989	3,213	3,115	3,091	3,416	3,431	3,361
11	서울특별시 강북구	288,113	288,113	791	808	945	998	1,004	1,178	1,343	1,426	1,551	1,531	1,693	1,880	1,777	1,835
12	서울특별시 도봉구	306,948	306,948	952	1,070	1,126	1,112	1,308	1,406	1,571	1,773	1,922	2,002	1,918	2,171	2,165	2,188
13	서울특별시 노원구	498,213	498,213	2,000	2,137	2,053	2,147	2,285	2,371	2,843	3,261	3,620	3,708	3,734	4,310	4,375	4,539
14	서울특별시 은평구	466,770	466,770	1,740	2,020	2,053	2,102	2,220	2,399	2,554	2,936	3,196	3,054	3,100	3,499	3,354	3,514

gender.csv 데이터 분석

- 성별 인구 자료 내부 구조
 - 1세 기준, 전체 시군구 현황

연령별 총 인구수						남성 연령별 인구수					여성 연령별 인구수				
행정 구역	총인구 수	연령 구간 인구수	계_0세	...	계100세	남자 총인구 수	남자 연령 구간 총인구수	남자 0세	...	남자 100세 이상	여자 총인구수	여자 연령 구간 총인구수	여자 0세	...	여자 100세 이상
[0]	[1]	[2]	[3]	...	[103]	[104]	[105]	[106]		[206]	[207]	[208]	[209]		[309]

- 남성 데이터
 - 총 인구수: [인덱스\[104\]](#)
 - 연령별 인구수(0세~100세 이상): [106] ~ [206]
- 여성 데이터
 - 총 인구수: [인덱스\[207\]](#)
 - 연령별 인구수(0세~100세 이상): [209] ~ [309]

gender.csv 헤더 정보

```
import csv
```

<day03_gender_01.py>

```
f = open('gender.csv', encoding='utf-8-sig')
data = csv.reader(f)
header = next(data)
```

```
for i in range(len(header)):
    print(f'{i:3d}: {header[i]}', end=', ')
```

```
    if (i+1) % 10 == 0:
        print()
```

```
f.close()
```

[0]: 행정기관, [1]: 총 인구수, [2]: 연령구간인구수, [3]: 0세, [4]: 1세, [5]: 2세, [6]: 3세, [7]: 4세, [8]: 5세, [9]: 6세,
[10]: 7세, [11]: 8세, [12]: 9세, [13]: 10세, [14]: 11세, [15]: 12세, [16]: 13세, [17]: 14세, [18]: 15세, [19]: 16세,

. . .
[90]: 87세, [91]: 88세, [92]: 89세, [93]: 90세, [94]: 91세, [95]: 92세, [96]: 93세, [97]: 94세, [98]: 95세, [99]: 96세,
[100]: 97세, [101]: 98세, [102]: 99세, [103]: 100세 이상,

[104]: 남 인구수, [105]: 연령구간인구수, [106]: 0세, [107]: 1세, [108]: 2세, [109]: 3세,
[110]: 4세, [111]: 5세, [112]: 6세, [113]: 7세, [114]: 8세, [115]: 9세, [116]: 10세, [117]: 11세, [118]: 12세, [119]: 13세,

. . .
[200]: 94세, [201]: 95세, [202]: 96세, [203]: 97세, [204]: 98세, [205]: 99세, [206]: 100세 이상,

[207]: 여 인구수, [208]: 연령구간인구수, [209]: 0세,
[210]: 1세, [211]: 2세, [212]: 3세, [213]: 4세, [214]: 5세, [215]: 6세, [216]: 7세, [217]: 8세, [218]: 9세, [219]: 10세,

. . .
[220]: 11세, [221]: 12세, [222]: 13세, [223]: 14세, [224]: 15세, [225]: 16세, [226]: 17세, [227]: 18세, [228]: 19세, [229]: 20세,
[300]: 91세, [301]: 92세, [302]: 93세, [303]: 94세, [304]: 95세, [305]: 96세, [306]: 97세, [307]: 98세, [308]: 99세, [309]: 100세 이상,

연령별 성별 데이터 시각화 #1

- `barh(y, data)` 그래프: 수평 막대 그래프

<day03_gender_02.py>

```
import csv
import matplotlib.pyplot as plt
import platform
import koreanize_matplotlib

def print_population(population):
    '''
    특정 지역의 인구 현황을 화면에 출력함
    '''
    for i in range(len(population)):
        print(f'{i:3d}세: {population[i]:4d}명', end=' ')
        if (i + 1) % 10 == 0:
            print()
    print()
```

```
def draw_geneder_population(title, male_num_list, female_num_list):

    if platform.system() == 'Windows':
        plt.rc('font', family='Malgun Gothic')
    else:
        plt.rc('font', family='AppleGothic')

    # barh(y축 범위, data)
    plt.barh(range(len(male_num_list)), male_num_list, label='남성')
    plt.barh(range(len(female_num_list)), female_num_list, label='여성')
    plt.rcParams['axes.unicode_minus'] = False
    plt.title(title + ' 성별 인구 비율')
    plt.legend()
    plt.show()
```

연령별 성별 데이터 시각화 #2

```
def calculate_population():
    f = open('gender.csv', encoding='utf-8-sig')
    data = csv.reader(f)
    male_num_list = []
    female_num_list = []
```

<day03_gender_02.py>

행정 구역	총인구 수	연령 구간 인구수	계_0세	...	계100세	남자 총인구 수	남자 연령 구간 총인구수	남자 0세	...	남자 100세 이상	여자 총인구수	여자 연령 구간 총인구수	여자 0세	...	여자 100세 이상
[0]	[1]	[2]	[3]	...	[103]	[104]	[105]	[106]		[206]	[207]	[208]	[209]		[309]

```
district = input('시군구 이름을 입력하세요: ')
```

```
for row in data:
```

```
    if district in row[0]:
```

```
        for male in row[106:207]: # 남성 연령별 인구수 구간
            if ',' in male:
                male = male.replace(',', '') # 천단위 콤마 제거
                male_num_list.append(int(male))
```

```
        for female in row[209:310]:
            if ',' in female:
                female = female.replace(',', '')
                female_num_list.append(int(female))
```

```
    break
```

```
f.close()
```

```
print(f'남성 총 인구수:{sum(male_num_list):,} ')
```

```
print_population(male_num_list)
```

```
print('-----')
```

```
print(f'여성 총 인구수:{sum(female_num_list):,} ')
```

```
print_population(female_num_list)
```

```
draw_geneder_population(district, male_num_list, female_num_list)
```

```
calculate_population()
```

연령별 성별 데이터 시각화 #3: 실행 결과

시군구 이름을 입력하세요: 대구광역시 북구

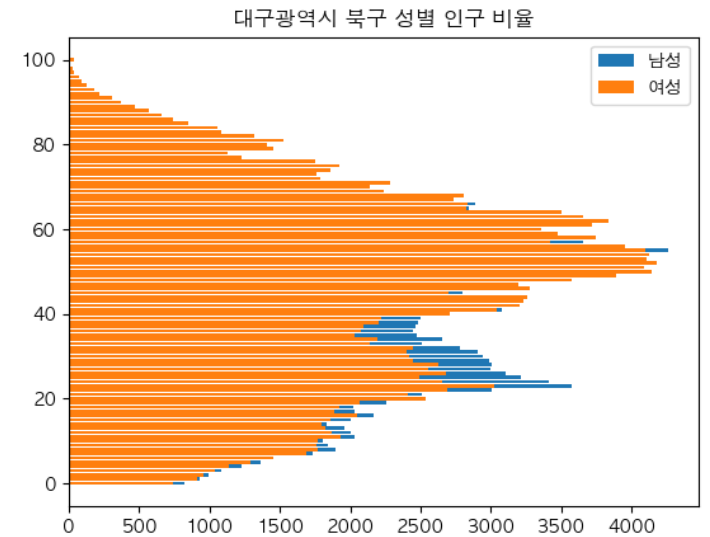
남성 총 인구수: 208,178

0세: 822명	1세: 933명	2세: 994명	3세: 1080명	4세: 1230명	5세: 1362명	6세: 1445명	7세: 1732명	8세: 1807명
10세: 1807명	11세: 2029명	12세: 2007명	13세: 1960명	14세: 1830명	15세: 2004명	16세: 2170명	17세: 2027명	18세: 2469명
20세: 2469명	21세: 2508명	22세: 3004명	23세: 3570명	24세: 3410명	25세: 3214명	26세: 3103명	27세: 2998명	28세: 2939명
30세: 2939명	31세: 2908명	32세: 2781명	33세: 2512명	34세: 2650명	35세: 2471명	36세: 2442명	37세: 2462명	38세: 2683명
40세: 2683명	41세: 3075명	42세: 3113명	43세: 3142명	44세: 3167명	45세: 2799명	46세: 3082명	47세: 3151명	48세: 3282명
50세: 3753명	51세: 3989명	52세: 4079명	53세: 4008명	54세: 3932명	55세: 4263명	56세: 3727명	57세: 3151명	58세: 2504명
60세: 3177명	61세: 3648명	62세: 3620명	63세: 3465명	64세: 3279명	65세: 2841명	66세: 2891명	67세: 2469명	68세: 2170명
70세: 1793명	71세: 1876명	72세: 1520명	73세: 1464명	74세: 1467명	75세: 1429명	76세: 1273명	77세: 998명	78세: 861명
80세: 861명	81세: 1092명	82세: 720명	83세: 595명	84세: 568명	85세: 480명	86세: 367명	87세: 246명	88세: 123명
90세: 123명	91세: 104명	92세: 65명	93세: 38명	94세: 22명	95세: 24명	96세: 13명	97세: 5명	98세: 5명
100세: 5명								

여성 총 인구수: 211,446

0세: 744명	1세: 910명	2세: 956명	3세: 1038명	4세: 1138명	5세: 1290명	6세: 1449명	7세: 1605명	8세: 1765명
10세: 1765명	11세: 1935명	12세: 1870명	13세: 1821명	14세: 1792명	15세: 1860명	16세: 2052명	17세: 2052명	18세: 2533명
20세: 2533명	21세: 2412명	22세: 2689명	23세: 3024명	24세: 2651명	25세: 2489명	26세: 2676명	27세: 2676명	28세: 2417명
30세: 2417명	31세: 2405명	32세: 2446명	33세: 2143명	34세: 2194명	35세: 2032명	36세: 2074명	37세: 2074명	38세: 2709명
40세: 2709명	41세: 3042명	42세: 3208명	43세: 3228명	44세: 3258명	45세: 2702명	46세: 3276명	47세: 3276명	48세: 4145명
50세: 4145명	51세: 4084명	52세: 4178명	53세: 4109명	54세: 4127명	55세: 4100명	56세: 3957명	57세: 3957명	58세: 3360명
60세: 3360명	61세: 3722명	62세: 3837명	63세: 3651명	64세: 3497명	65세: 2824명	66세: 2832명	67세: 2832명	68세: 2139명
70세: 2139명	71세: 2287명	72세: 1791명	73세: 1763명	74세: 1863명	75세: 1920명	76세: 1751명	77세: 1751명	78세: 1410명
80세: 1410명	81세: 1521명	82세: 1319명	83세: 1085명	84세: 1053명	85세: 846명	86세: 736명	87세: 605명	88세: 572명
90세: 374명	91세: 307명	92세: 215명	93세: 181명	94세: 122명	95세: 94명	96세: 74명	97세: 34명	98세: 29명
100세: 38명								99세: 10명

2개의 데이터가 겹쳐 있어
구분이 어려움



연령별 성별 데이터 시각화: 항아리 모양

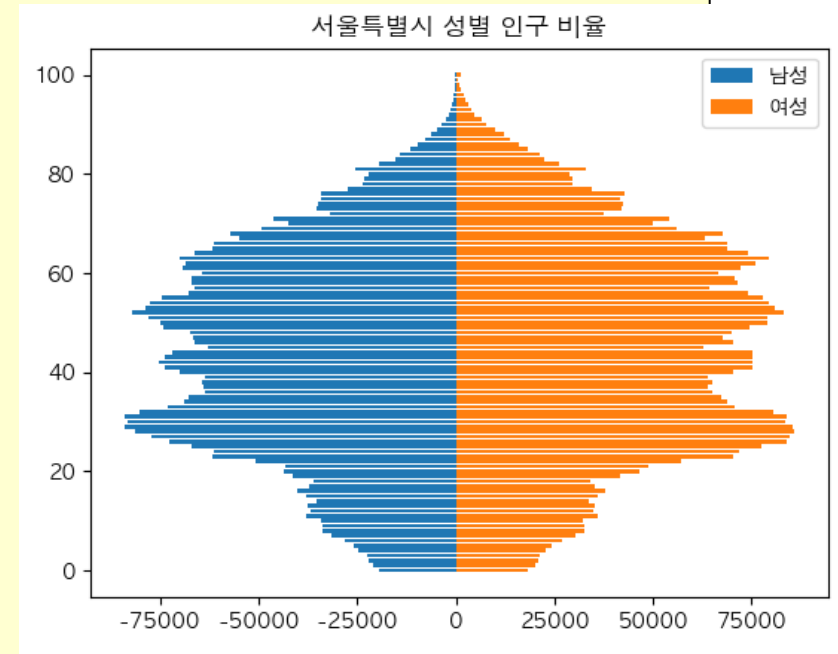
- 여성 데이터: 오른쪽, 남성 데이터: 왼쪽 표시
 - `male_num_list.append(-int(m))`: 음수로 변환

<day03_gender_02.py>

```
def calculate_population():
    f = open('gender.csv', encoding='utf-8-sig')
    data = csv.reader(f)
    male_num_list = []
    female_num_list = []

    district = input('지역(동) 이름을 입력하세요: ')
    for row in data:
        if district in row[0]:
            for male in row[106:207]: # 남성 연령별 인구수 구간
                if ',' in male:
                    male = male.replace(',', '') # 천단위 콤마 제거
                    male_num_list.append(-int(male))

            for female in row[209:310]:
                if ',' in female:
                    female = female.replace(',', '')
                    female_num_list.append(int(female))
            break
    f.close()
```



제주도의 성별 인구 비율 표현

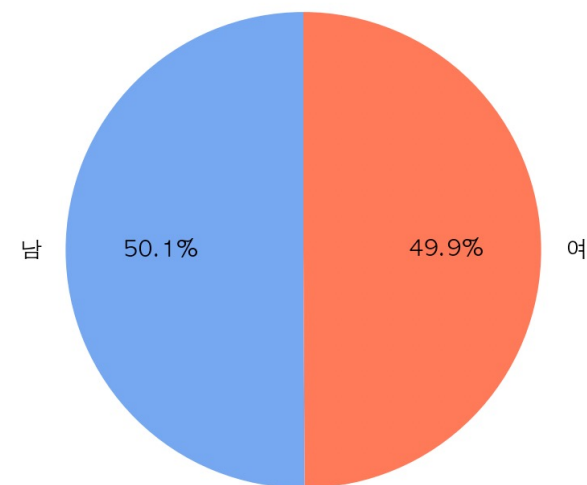
■ 제주특별자치도 인구 현황

행정구역	...	남 인구수	남 연령구간 인구수	...	여 인구수	여 연령구간 인구수	...
제주특별자치 도	...	337,976	337,976		337,276	337,276	...
[0]	...	[104]	[105]	...	[207]	[208]	...

찾고 싶은 지역의 이름을 입력하세요: 제주특별자치도

제주특별자치도 남자 인구수: 337,976명, 여자 인구수: 337,276명

제주특별자치도 남녀 성별 비율



특정 지역의 남녀 인구 비율 예제

```
import csv
import matplotlib.pyplot as plt
import koreanize_matplotlib

f = open('gender.csv', encoding='utf-8-sig')
data = csv.reader(f)
population=[] # Pie chart에 넣을 데이터 (남, 여 인구수)
city = input('찾고 싶은 지역의 이름을 입력하세요: ')
male_count = 0
female_count = 0

for row in data:
    if city in row[0]:
        male_count = int(row[104].replace(',','')) # 자리수 문자열 제거 및 숫자로 변환
        female_count = int(row[207].replace(',',''))
        break # 도시별 하위 목록이 많음. 처음에 나오는 데이터가 전체 총합

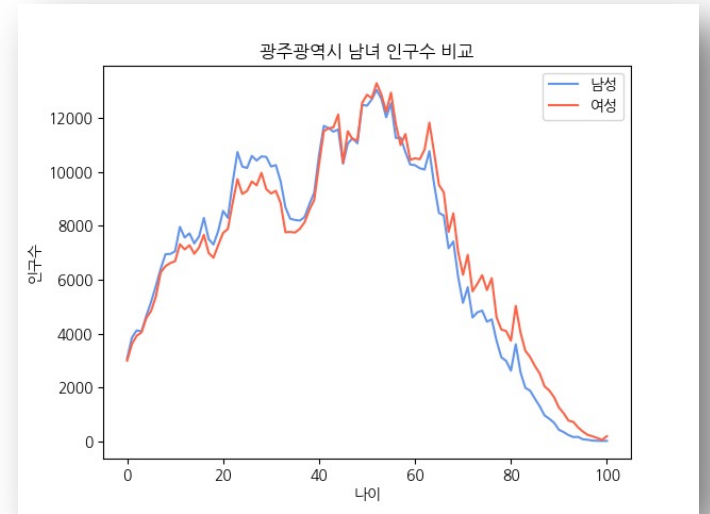
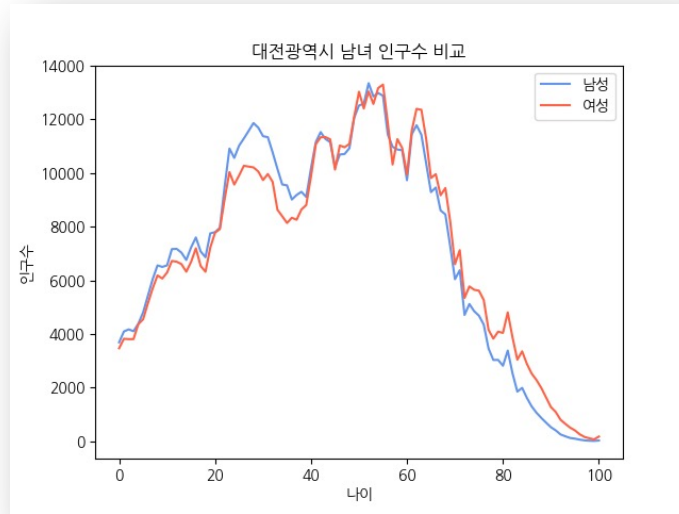
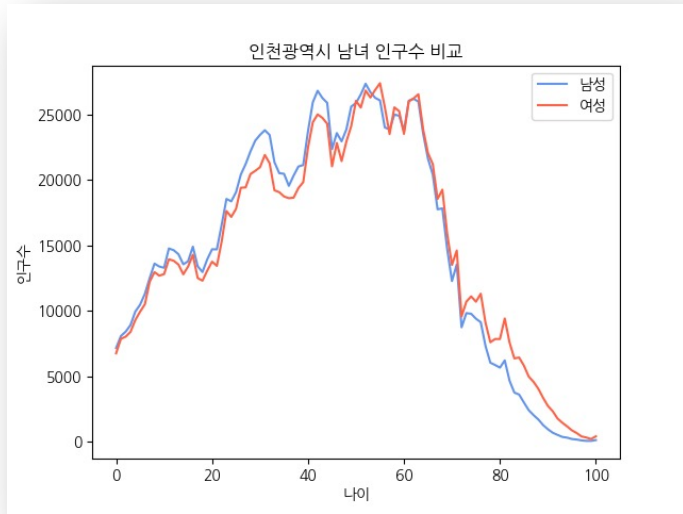
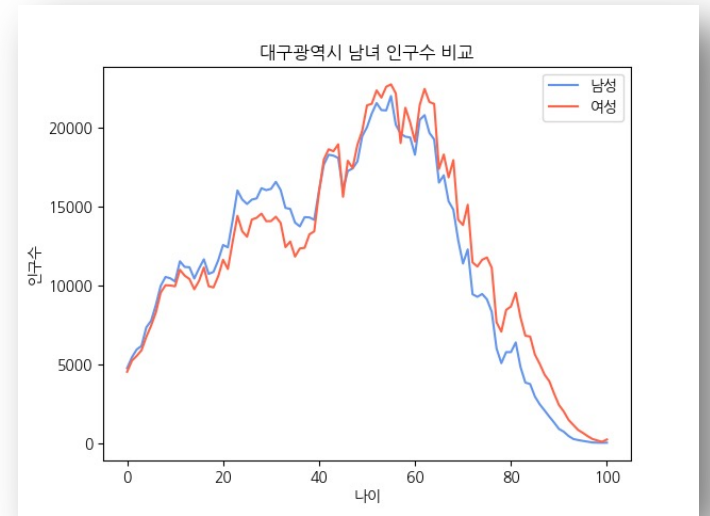
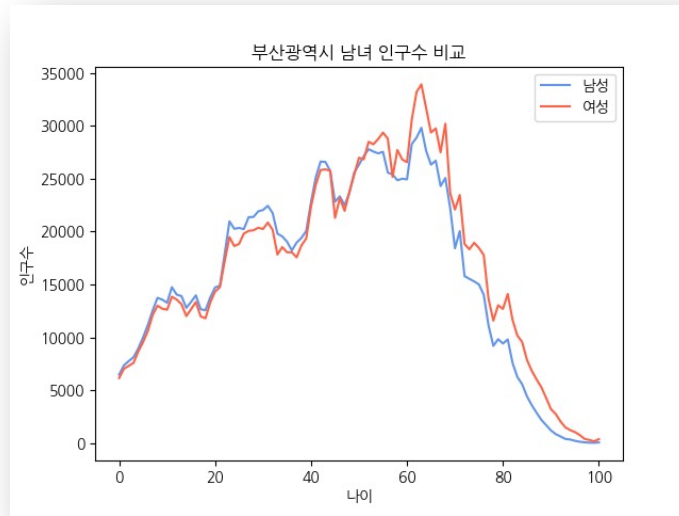
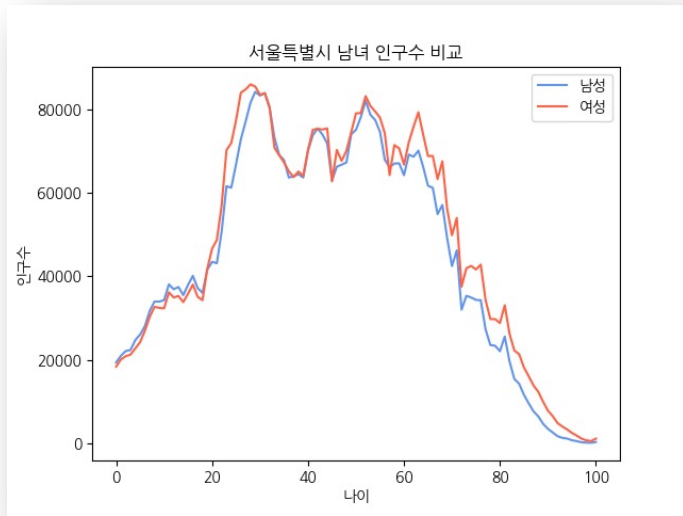
print(f'{city} 남자 인구수: {male_count:,}명, 여자 인구수: {female_count:,}명')

population = [male_count, female_count]
color = ['cornflowerblue', 'tomato']
plt.pie(population, labels=['남', '여'], autopct='%.1f%%', colors=color, startangle=90)
plt.title(city + " 남녀 성별 비율")
plt.show()
```

<day03_gender_03.py>

파이 차트의 시작
각도 설정(90도)

서울특별시 및 5대 광역시 연령대별 남녀 인구수 비교



서울특별시 및 5대 광역시 의 남녀 인구수 비교 #1

```
import csv
import matplotlib.pyplot as plt
import platform
import koreanize_matplotlib

f = open('gender.csv', encoding='utf-8-sig')
data = csv.reader(f)
city_list = ['서울특별시', '부산광역시', '대구광역시', '인천광역시', '광주광역시', '대전광역시' ]
for city in city_list:
    male_list = [] # 리스트 데이터 초기화
    female_list = [] # 리스트 데이터 초기화
    for row in data:
        if city in row[0]:
            for i in range(106, 207):
                male_list.append(int(row[i].replace(',','')))
                female_list.append(int(row[i+103].replace(',','')))
            break # 도시 하위 목록이 많음.

color = ['cornflowerblue', 'tomato']
plt.plot(male_list, label='남성', color=color[0])
plt.plot(female_list, label='여성', color=color[1])
plt.title(city + " 남녀 인구수 비교")
plt.xlabel('나이')
plt.ylabel('인구수')
plt.legend()
plt.savefig('img/' + city + '.png', dpi=100)
plt.close()
```

<day03_gender_04.py>

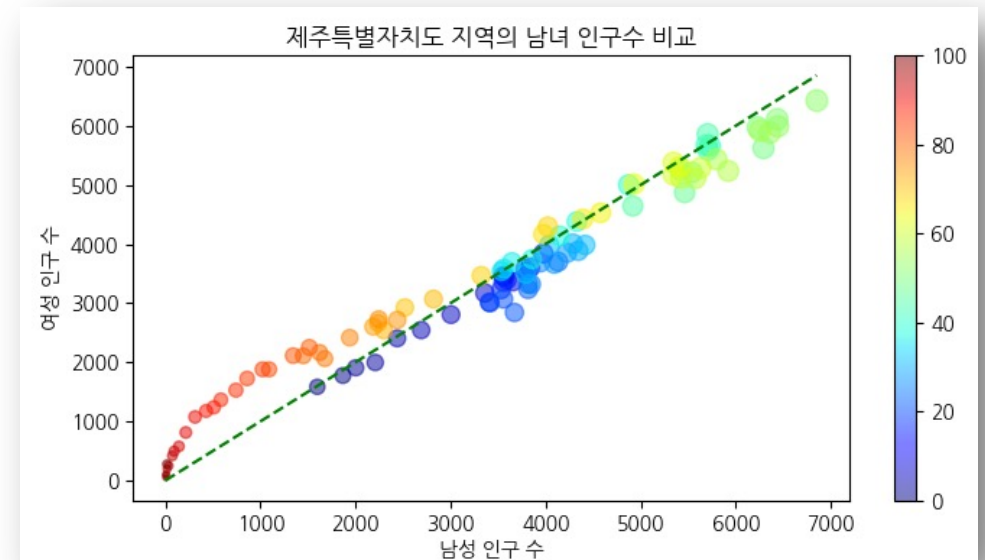
산점도(scatter)로 표현하기

■ 산점도

- 가로축과 세로축을 기준으로 두 요소가 서로 어떤 관계를 맺고 있는지를 파악하기 쉽게 나타낸 그래프

■ 산점도 x축, y축 상관 관계

- 각 점들은 오른쪽 컬러바를 참고하여 색깔별로 나이를 표시함
- 버블의 위치: 남녀 성비
 - 기준선 아래: 남성의 비율이 높음
 - 기준선 위쪽: 여성의 비율이 높음
- 버블의 색상: 연령대
- 버블의 크기: 연령대별 인구수를 표현

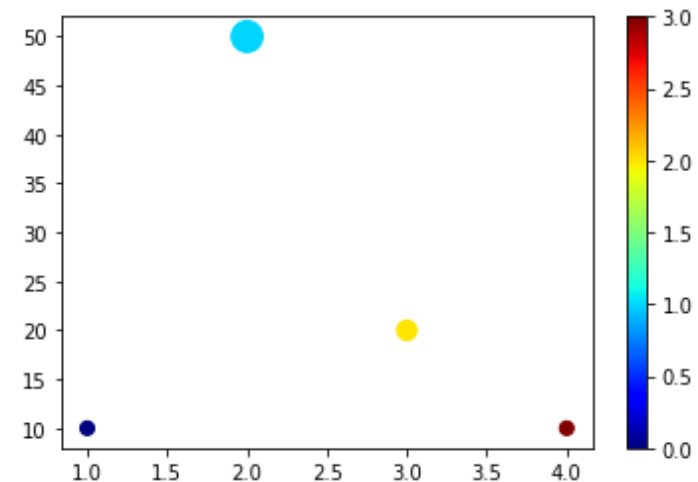


산점도에 color bar 추가하기

- `colorbar()` 함수
 - 그래프 우측에 color bar를 추가함
- `scatter()` 함수 속성 추가
 - `c=range(색상 개수)`
 - 각 데이터에 해당하는 color bar의 색으로 정해짐
 - `cmap`: 컬러맵 속성 사용 (`cmap='jet'`) – 무지개색
 - <https://matplotlib.org/tutorials/colors/colormaps.html?highlight=colormap>

```
import matplotlib.pyplot as plt
y_value = [10, 50, 20, 10]
x_value = [1, 2, 3, 4]
size = []
for y in y_value:
    size.append(y * 5)

plt.scatter(x_value, y_value, s=size, c=range(4), cmap='jet')
plt.colorbar()
plt.show()
```



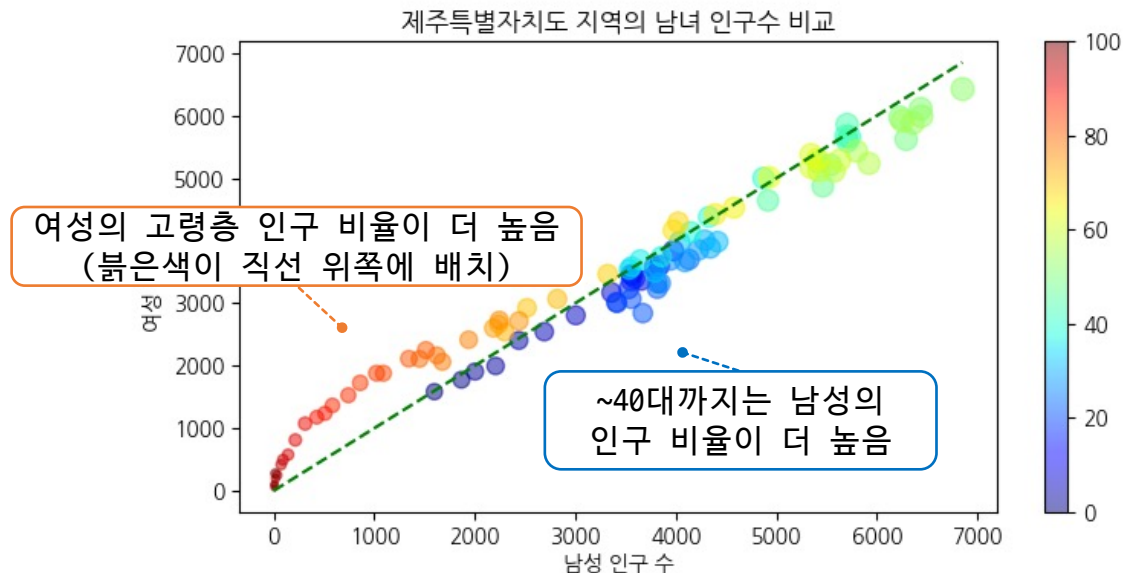
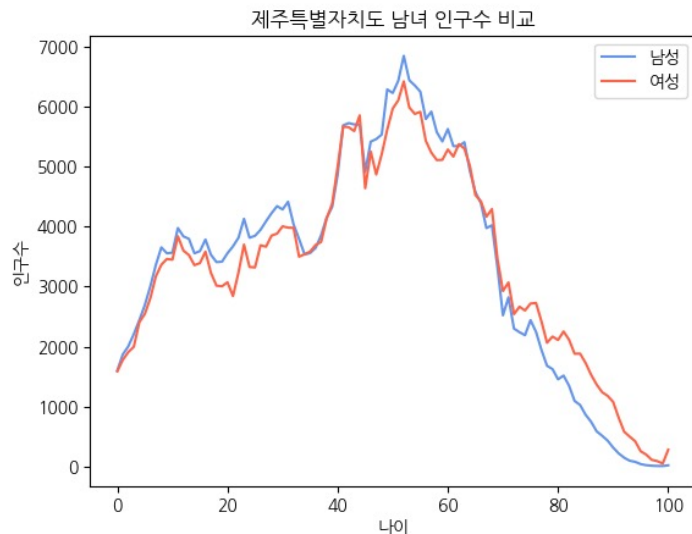
인구 분포: 산점도

■ 특정 지역의 인구 관계

찾고 싶은 지역의 이름을 입력하세요: **제주특별자치도**

[여성 인구]: [1585, 1776, 1907, 1996, 2405, 2544, 2806, 3172, 3365, 3461, 3449, 3840, 3601, 3527, 3358, 3392, 3584, 3229, 3016, 3005, 3072, 2842, 3235, 3701, 3327, 3320, 3690, 3666, 3852, 3884, 4006, 3984, 3980, 3499, 3540, 3585, 3695, 3745, 4139, 4381, 5001, 5672, 5660, 5596, 5858, 4642, 5255, 4871, 5211, 5623, 5972, 6110, 6426, 5990, 5883, 5917, 5433, 5236, 5112, 5116, 5288, 5170, 5379, 5303, 5013, 4528, 4423, 4166, 4294, 3465, 2926, 3069, 2540, 2664, 2601, 2717, 2727, 2416, 2062, 2166, 2109, 2251, 2113, 1880, 1881, 1726, 1528, 1366, 1238, 1176, 1075, 813, 578, 496, 418, 251, 196, 108, 87, 46, 278]

[남성 인구]: [1599, 1868, 2003, 2208, 2439, 2692, 3005, 3361, 3655, 3556, 3566, 3977, 3838, 3798, 3556, 3592, 3786, 3531, 3409, 3415, 3557, 3672, 3816, 4132, 3816, 3848, 3948, 4091, 4225, 4343, 4288, 4416, 4038, 3793, 3537, 3560, 3646, 3857, 4149, 4329, 4872, 5692, 5729, 5707, 5703, 4917, 5420, 5462, 5538, 6291, 6231, 6436, 6852, 6444, 6358, 6254, 5797, 5921, 5576, 5423, 5630, 5345, 5345, 5410, 4930, 4579, 4392, 3976, 4024, 3323, 2520, 2821, 2299, 2239, 2188, 2440, 2245, 1940, 1678, 1623, 1452, 1515, 1345, 1092, 1024, 861, 743, 585, 510, 429, 313, 216, 144, 93, 75, 37, 19, 10, 6, 5, 17]



제주도의 연령대별 성별 비율 산점도 #1

```
import csv
import matplotlib.pyplot as plt
import platform
import math
import koreanize_matplotlib

<day03_gender_05.py>

def draw_scatter(city, male_list, female_list, bubble_size_list):
    if platform.system() == 'Windows':
        plt.rc('font', family='Malgun Gothic')
    else:
        plt.rc('font', family='AppleGothic')

    plt.figure(figsize=(8, 4), dpi=100)
    plt.scatter(male_list, female_list, s=bubble_size_list, c=range(101), alpha=0.5, cmap='jet')
    plt.colorbar()
    plt.plot(range(max(male_list)), range(max(male_list)), 'g--') # 추세선 추가(녹색 점선)

    plt.title(city + " 지역의 남녀 인구수 비교")
    plt.xlabel('남성 인구 수')
    plt.ylabel('여성 인구 수')
    plt.show()
```

제주도의 연령대별 성별 비율 산점도 #2

```
def calculate_population():  
    f = open('gender.csv', encoding='utf-8-sig')  
    data = csv.reader(f)  
    male_list = []  
    female_list = []  
    bubble_size_list = []  
    city = input('찾고 싶은 지역의 이름을 입력하세요: ')  
  
    for row in data:  
        if city in row[0]:  
            for i in range(106, 207):  
                male_num = int(row[i].replace(',',''))  
                female_num = int(row[i + 103].replace(',',''))  
                # 버블의 사이즈 조절  
                bubble_size_list.append(math.sqrt(male_num + female_num))  
                #bubble_size_list.append(math.log(pow(male_num,2) + pow(female_num,2)))  
                male_list.append(male_num)  
                female_list.append(female_num)  
            break  
  
    f.close()  
    print(f'[여성 인구]: {female_list}')    print(f'[남성 인구]: {male_list}')    draw_scatter(city, male_list, female_list, bubble_size_list)  
  
calculate_population()
```

<day03_gender_05.py>



Questions?