

# 서울시 기온데이터

## ■ 서울시 기온데이터

- 기상자료개방포털 (<https://data.kma.go.kr>)
- 기후통계분석 → 기후분석 → 기온분석
- 데이터 가져오기

```
import pandas as pd
df = pd.read_csv('data/seoul.csv', encoding='cp949')
print(df)
```

|       | 날짜         | 지점  | 평균기온(℃) | 최저기온(℃) | 최고기온(℃) |
|-------|------------|-----|---------|---------|---------|
| 0     | 1907-10-01 | 108 | 13.5    | 7.9     | 20.7    |
| 1     | 1907-10-02 | 108 | 16.2    | 7.9     | 22.0    |
| 2     | 1907-10-03 | 108 | 16.2    | 13.1    | 21.3    |
| 3     | 1907-10-04 | 108 | 16.5    | 11.2    | 22.0    |
| 4     | 1907-10-05 | 108 | 17.6    | 10.9    | 25.4    |
| ...   | ...        | ... | ...     | ...     | ...     |
| 40930 | 2020-12-27 | 108 | 5.8     | 1.4     | 10.0    |
| 40931 | 2020-12-28 | 108 | 6.7     | 4.2     | 11.4    |
| 40932 | 2020-12-29 | 108 | 0.1     | -6.2    | 4.3     |
| 40933 | 2020-12-30 | 108 | -10.9   | -12.9   | -6.2    |
| 40934 | 2020-12-31 | 108 | -8.9    | -12.9   | -5.0    |

40935 rows × 5 columns

## ■ 서울시 기온데이터

- 1907년 ~ 2020년 중 가장 더웠던 날

```
df[df['최고기온(℃)'] == df['최고기온(℃)'].max()]
```

|       | 날짜         | 지점  | 평균기온(℃) | 최저기온(℃) | 최고기온(℃) |
|-------|------------|-----|---------|---------|---------|
| 40051 | 2018-08-01 | 108 | 33.6    | 27.8    | 39.6    |

- 일교차가 가장 큰 날

```
df['일교차'] = df['최고기온(℃)'] - df['최저기온(℃)']  
df[df['일교차'] == df['일교차'].max()]
```

|   | 날짜         | 지점  | 평균기온(℃) | 최저기온(℃) | 최고기온(℃) | 일교차  |
|---|------------|-----|---------|---------|---------|------|
| 0 | 1907-10-01 | 108 | 13.5    | 7.9     | 20.7    | 12.8 |
| 1 | 1907-10-02 | 108 | 16.2    | 7.9     | 22.0    | 14.1 |
| 2 | 1907-10-03 | 108 | 16.2    | 13.1    | 21.3    | 8.2  |
| 3 | 1907-10-04 | 108 | 16.5    | 11.2    | 22.0    | 10.8 |
| 4 | 1907-10-05 | 108 | 17.6    | 10.9    | 25.4    | 14.5 |

|       | 날짜         | 지점  | 평균기온(℃) | 최저기온(℃) | 최고기온(℃) | 일교차  |
|-------|------------|-----|---------|---------|---------|------|
| 12619 | 1942-04-19 | 108 | 12.7    | 2.5     | 24.3    | 21.8 |

## ■ 서울시 기온데이터

### ● 누락값 확인

```
df.count()
```

```
날짜      40935
지점      40935
평균기온(℃)  40179
최저기온(℃)  40178
최고기온(℃)  40177
dtype: int64
```

```
df[ pd.isnull(df['평균기온(℃)']) ]
```

|       | 날짜         | 지점  | 평균기온(℃) | 최저기온(℃) | 최고기온(℃) | 일교차 |
|-------|------------|-----|---------|---------|---------|-----|
| 15676 | 1950-09-01 | 108 | NaN     | NaN     | NaN     | NaN |
| 15677 | 1950-09-02 | 108 | NaN     | NaN     | NaN     | NaN |
| 15678 | 1950-09-03 | 108 | NaN     | NaN     | NaN     | NaN |
| 15679 | 1950-09-04 | 108 | NaN     | NaN     | NaN     | NaN |
| 15680 | 1950-09-05 | 108 | NaN     | NaN     | NaN     | NaN |
| ...   | ...        | ... | ...     | ...     | ...     | ... |
| 16427 | 1953-11-26 | 108 | NaN     | NaN     | NaN     | NaN |
| 16428 | 1953-11-27 | 108 | NaN     | NaN     | NaN     | NaN |
| 16429 | 1953-11-28 | 108 | NaN     | NaN     | NaN     | NaN |
| 16430 | 1953-11-29 | 108 | NaN     | NaN     | NaN     | NaN |
| 16431 | 1953-11-30 | 108 | NaN     | NaN     | NaN     | NaN |

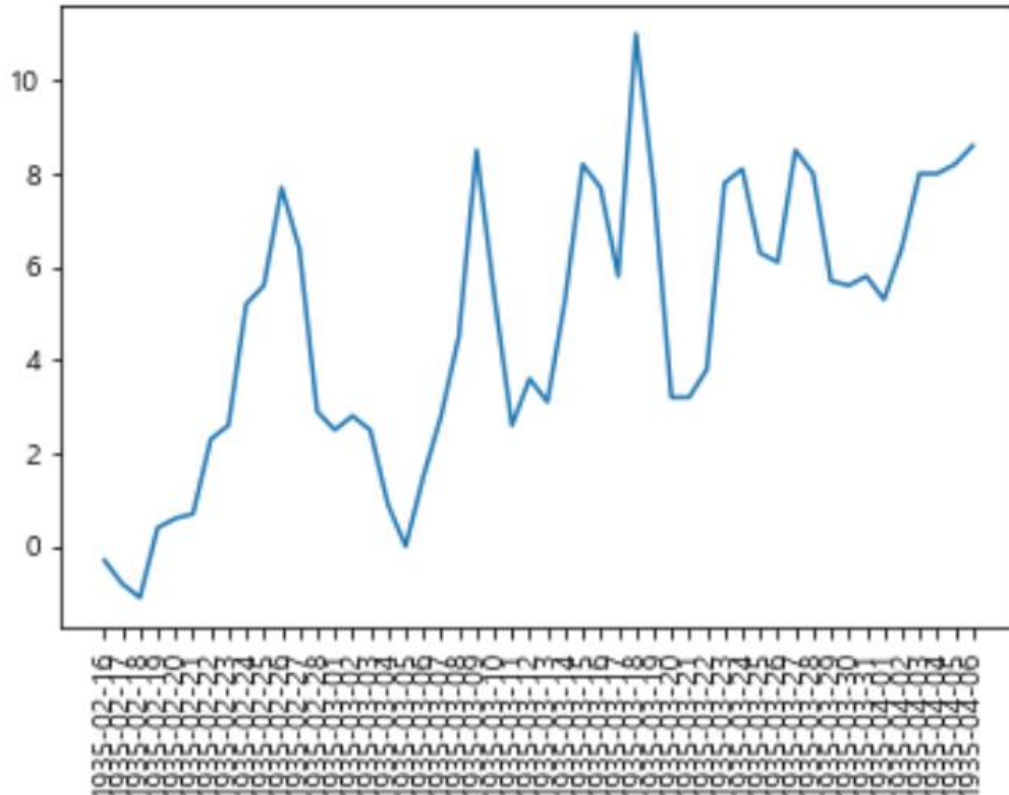
평균기온이 누락된 데이터 확인

756 rows x 6 columns

## ■ 서울시 기온데이터

### ● 그래프 표현

```
import matplotlib.pyplot as plt
plt.plot(df['날짜'][10000:10050], df['평균기온(°C)'][10000:10050])
plt.xticks(df['날짜'][10000:10050], rotation=90)
plt.show()
```



## ■ 서울시 기온데이터

### ● 2000년 이후 3월 1일 이후 최고 / 최저 기온 확인

- 연도 / 월 / 일 분리

```
df3['연도'] = df3['날짜'].apply(lambda x : x.split('-')[0])
df3['월'] = df3['날짜'].apply(lambda x : x.split('-')[1])
df3['일'] = df3['날짜'].apply(lambda x : x.split('-')[2])
print(df3.head())
```

|   | 날짜         | 지점  | 평균기온(℃) | 최저기온(℃) | 최고기온(℃) | 일교차  | 연도   | 월  | 일  |
|---|------------|-----|---------|---------|---------|------|------|----|----|
| 0 | 1907-10-01 | 108 | 13.5    | 7.9     | 20.7    | 12.8 | 1907 | 10 | 01 |
| 1 | 1907-10-02 | 108 | 16.2    | 7.9     | 22.0    | 14.1 | 1907 | 10 | 02 |
| 2 | 1907-10-03 | 108 | 16.2    | 13.1    | 21.3    | 8.2  | 1907 | 10 | 03 |
| 3 | 1907-10-04 | 108 | 16.5    | 11.2    | 22.0    | 10.8 | 1907 | 10 | 04 |
| 4 | 1907-10-05 | 108 | 17.6    | 10.9    | 25.4    | 14.5 | 1907 | 10 | 05 |

## ■ 서울시 기온데이터

- 2000년 이후 3월 1일 이후 최고 / 최저 기온 확인
  - 2000년 이후 연도만 리스트로 저장

```
year = df3[ df3['연도'] >= '2000' ]['연도'].unique()  
ticks = range(len(year))  
print(year)  
print(ticks)
```

```
['2000' '2001' '2002' '2003' '2004' '2005' '2006' '2007' '2008' '2009'  
 '2010' '2011' '2012' '2013' '2014' '2015' '2016' '2017' '2018' '2019'  
 '2020']  
range(0, 21)
```

## ■ 서울시 기온데이터

- 2000년 이후 3월 1일 이후 최고 / 최저 기온 확인
  - 매년 3월 1일의 최고 기온 확인

```
최고기온 = df3[
    (df3['연도'] >= '2000') &
    (df3['월'] == '03') &
    (df3['일'] == '01')]['최고기온(°C)']
최고기온.index = ticks
print(최고기온)
```

```
0      7.7
1      8.5
2     13.1
3     10.2
4      6.0
5      4.5
6      2.5
7     15.3
8      7.8
9      8.6
10     6.3
11     3.7
12    14.6
13     7.7
14     9.4
15     5.9
16     2.1
17    11.4
18     3.0
19    13.0
20    10.1
```

Name: 최고기온(°C), dtype: float64



## ■ 서울시 기온데이터

- 2000년 이후 3월 1일 이후 최고 / 최저 기온 확인
  - 매년 3월 1일의 최저 기온 확인

```
최저기온 = df3[
    (df3['연도'] >= '2000') &
    (df3['월'] == '03') &
    (df3['일'] == '01')]['최저기온(°C)']
최저기온.index = ticks
print(최저기온)
```

```
0    -3.2
1     1.7
2     1.1
3     4.0
4    -1.1
5    -4.5
6    -2.4
7     3.9
8    -0.5
9    -1.6
10    1.1
11   -2.7
12    4.0
13   -2.4
14    3.3
15   -0.2
16   -7.5
17    2.0
18   -4.3
19    1.2
20    3.2
```

Name: 최저기온(°C), dtype: float64

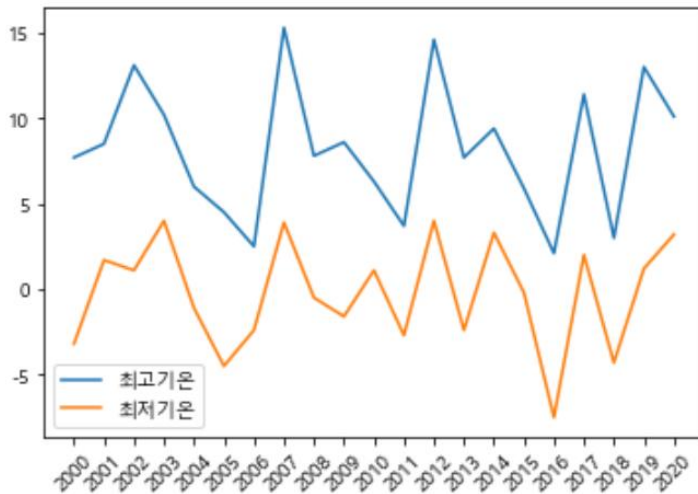
## ■ 서울시 기온데이터

- 2000년 이후 3월 1일 이후 최고 / 최저 기온 확인

```
import matplotlib.pyplot as plt

plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False

plt.plot(최고기온, label='최고기온')
plt.plot(최저기온, label='최저기온')
plt.xticks(ticks, year, rotation=45)
plt.legend()
plt.show()
```



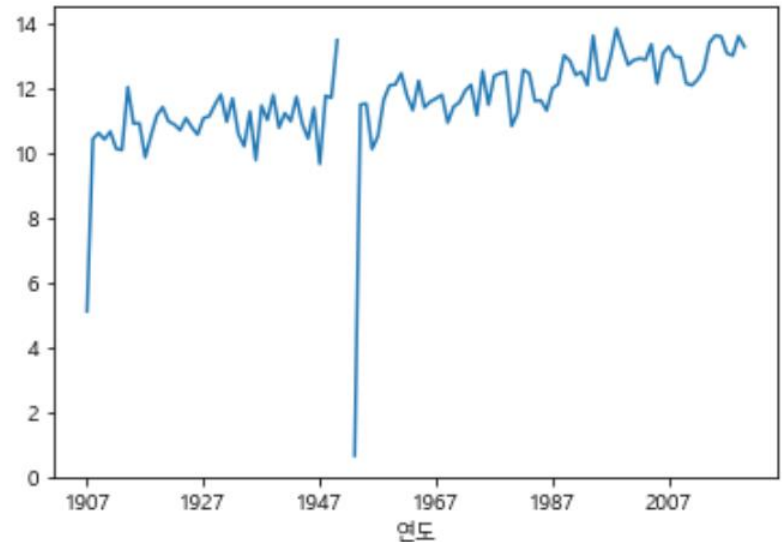
## ■ 서울시 기온데이터

### ● 연도별 평균기온

```
df3_group = df3.groupby('연도')['평균기온(℃)'].mean()  
print(df3_group)  
df3_group.plot()
```

| 연도   | 평균기온(℃)   |
|------|-----------|
| 1907 | 5.103261  |
| 1908 | 10.427322 |
| 1909 | 10.608767 |
| 1910 | 10.412055 |
| 1911 | 10.654795 |
| ...  |           |
| 2016 | 13.592896 |
| 2017 | 13.073151 |
| 2018 | 13.002466 |
| 2019 | 13.598904 |
| 2020 | 13.271858 |

Name: 평균기온(℃), Length: 114, dtype: float64

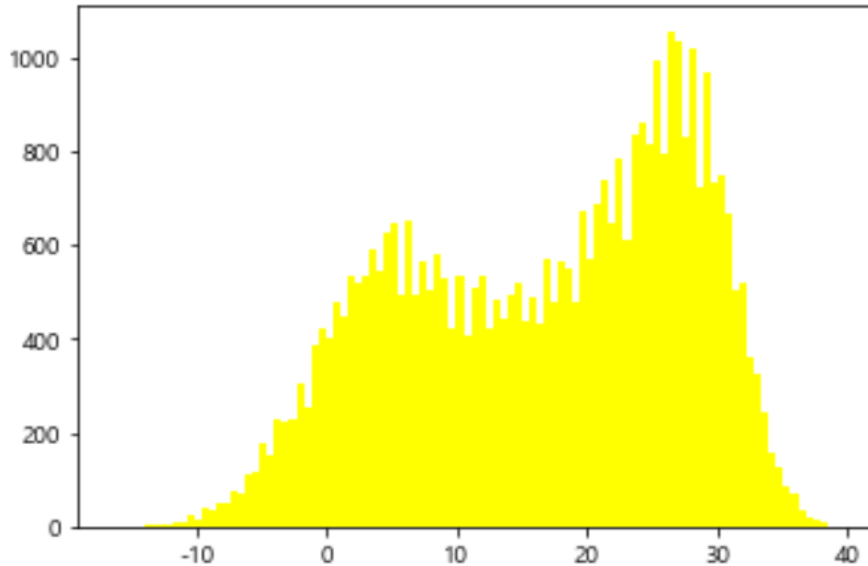


## ■ 서울시 기온데이터

- 연도별 최고기온 데이터를 히스토그램으로 표현

```
import matplotlib.pyplot as plt
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False

plt.hist(df['최고기온(°C)'], bins=100, color='yellow')
plt.show()
```



# 인구 공공데이터

## ■ 인구 공공데이터

- 행정안전부 (<https://www.mois.go.kr>)
- 정책자료 → 통계 → 주민등록 인구 통계
- 데이터 가져오기

```
df = pd.read_csv(  
    'data/people_202101.csv', encoding='cp949', thousands=',')  
print(df.head())
```

|   | 행정구역                                   | 2021년<br>01월_계<br>_총인구<br>수 | 2021년<br>01월_계<br>_연령구<br>간인구<br>수 | 2021<br>년01<br>월_계<br>_0세 | 2021<br>년01<br>월_계<br>_1세 | 2021<br>년01<br>월_계<br>_2세 | 2021<br>년01<br>월_계<br>_3세 | 2021<br>년01<br>월_계<br>_4세 | 2021<br>년01<br>월_계<br>_5세 | 2021<br>년01<br>월_계<br>_6세 | ... |
|---|--|-----------------------------|------------------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|---------------------------|-----|
| 0 | 서울특별시<br>(1100000000)                  | 9657969                     | 9657969                            | 44249                     | 49910                     | 52741                     | 56732                     | 63560                     | 68419                     | 67870                     | ... |
| 1 | 서울특별시<br>종로구<br>(1111000000)           | 149125                      | 149125                             | 464                       | 588                       | 564                       | 663                       | 761                       | 841                       | 833                       | ... |
| 2 | 서울특별시<br>종로구 청운<br>효자동<br>(1111051500) | 12426                       | 12426                              | 47                        | 60                        | 65                        | 77                        | 71                        | 104                       | 91                        | ... |
| 3 | 서울특별시<br>종로구 사직<br>동<br>(1111053000)   | 9544                        | 9544                               | 27                        | 37                        | 33                        | 54                        | 69                        | 64                        | 82                        | ... |
| 4 | 서울특별시<br>종로구 삼청<br>동<br>(1111054000)   | 2613                        | 2613                               | 7                         | 7                         | 6                         | 16                        | 3                         | 17                        | 7                         | ... |

5 rows × 104 columns

## ■ 인구 공공데이터

### ● 서울특별시 관악구 신림동(1162069500) 정보 확인

```
df2 = df[ df['행정구역'] == '서울특별시 관악구 신림동(1162069500)' ]
print(df2)
```

| 행정구역                                 | 2021<br>년01<br>월_계<br>_총인<br>_구수 | 2021  | 2021                           | 2021                           | 2021                           | 2021                           | 2021                           | 2021                           | 2021                           | ...    |
|--------------------------------------|----------------------------------|---|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------|
|                                      |                                  | 2021<br>년01<br>월_계<br>_연령<br>_구간<br>_인구<br>_수 | 2021<br>년01<br>월_계<br>_0<br>_세 | 2021<br>년01<br>월_계<br>_1<br>_세 | 2021<br>년01<br>월_계<br>_2<br>_세 | 2021<br>년01<br>월_계<br>_3<br>_세 | 2021<br>년01<br>월_계<br>_4<br>_세 | 2021<br>년01<br>월_계<br>_5<br>_세 | 2021<br>년01<br>월_계<br>_6<br>_세 |        |
| 서울특별시<br>관악구 신림<br>동<br>(1162069500) | 355                              | 21831   | 21831                          | 42                             | 43                             | 29                             | 47                             | 39                             | 37                             | 40 ... |

1 rows × 104 columns

## ■ 인구 공공데이터

### ● 2021년 1월 연령별 인구 현황 그래프로 표현

- 불필요 컬럼 삭제

```
df2 = df2.drop(
    ['행정구역', '2021년01월_계_총인구수', '2021년01월_계_연령구간인구수'], axis=1)
df2.head()
```

| 2021<br>년01<br>월_0<br>계_0<br>세 | 2021<br>년01<br>월_1<br>계_1<br>세 | 2021<br>년01<br>월_2<br>계_2<br>세 | 2021<br>년01<br>월_3<br>계_3<br>세 | 2021<br>년01<br>월_4<br>계_4<br>세 | 2021<br>년01<br>월_5<br>계_5<br>세 | 2021<br>년01<br>월_6<br>계_6<br>세 | 2021<br>년01<br>월_7<br>계_7<br>세 | 2021<br>년01<br>월_8<br>계_8<br>세 | 2021<br>년01<br>월_9<br>계_9<br>세 | ... |
|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|-----|
|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|--------------------------------|-----|

---

|     |    |    |    |    |    |    |    |    |    |    |     |
|-----|----|----|----|----|----|----|----|----|----|----|-----|
| 355 | 42 | 43 | 29 | 47 | 39 | 37 | 40 | 21 | 29 | 32 | ... |
|-----|----|----|----|----|----|----|----|----|----|----|-----|

1 rows x 101 columns



## ■ 인구 공공데이터

### ● 2021년 1월 연령별 인구 현황 그래프로 표현

- 열의 정보를 행으로 정리

```
df3 = df2.melt()  
df3.drop('variable', axis=1, inplace=True)  
df3
```

|     | variable           | value |
|-----|--------------------|-------|
| 0   | 2021년01월_계_0세      | 42    |
| 1   | 2021년01월_계_1세      | 43    |
| 2   | 2021년01월_계_2세      | 29    |
| 3   | 2021년01월_계_3세      | 47    |
| 4   | 2021년01월_계_4세      | 39    |
| ... | ...                | ...   |
| 96  | 2021년01월_계_96세     | 1     |
| 97  | 2021년01월_계_97세     | 1     |
| 98  | 2021년01월_계_98세     | 1     |
| 99  | 2021년01월_계_99세     | 3     |
| 100 | 2021년01월_계_100세 이상 | 16    |

101 rows × 2 columns



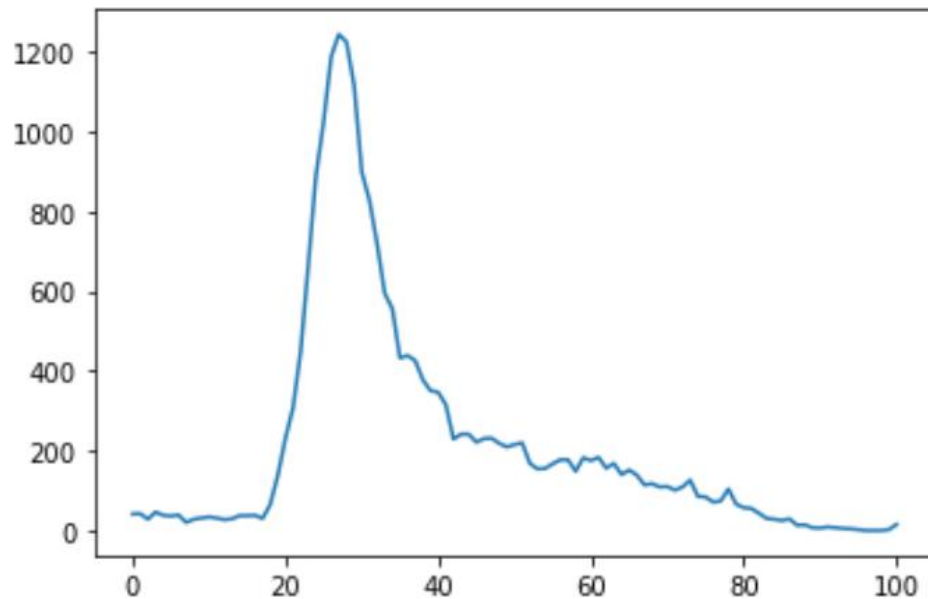
|     | value |
|-----|-------|
| 0   | 42    |
| 1   | 43    |
| 2   | 29    |
| 3   | 47    |
| 4   | 39    |
| ... | ...   |
| 96  | 1     |
| 97  | 1     |
| 98  | 1     |
| 99  | 3     |
| 100 | 16    |

101 rows × 1 columns

## ■ 인구 공공데이터

### ● 2021년 1월 연령별 인구 현황 그래프로 표현

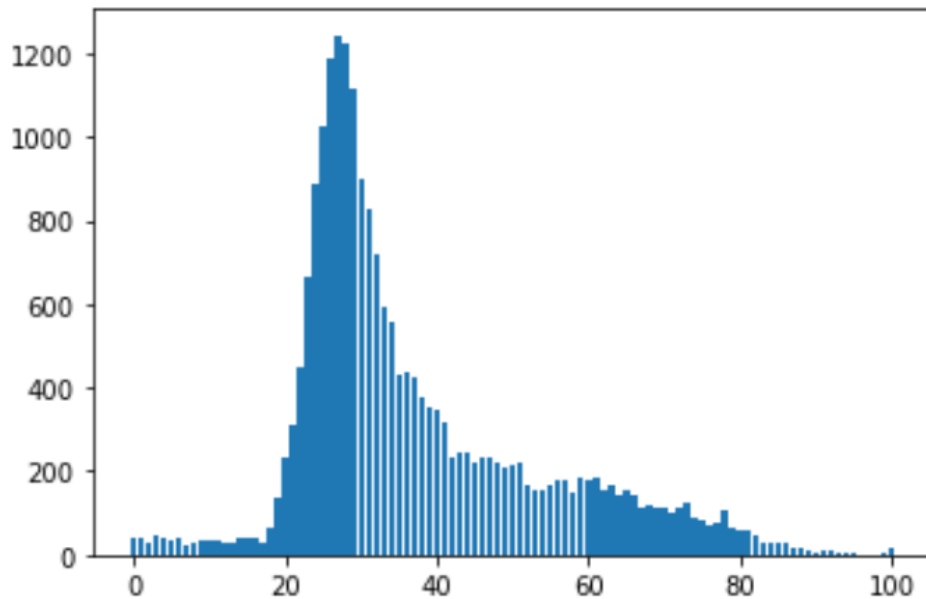
```
import matplotlib.pyplot as plt  
plt.plot(df3)  
plt.show()
```



## ■ 인구 공공데이터

- 2021년 1월 연령별 인구 현황 그래프로 표현

```
import matplotlib.pyplot as plt  
plt.bar(range(101), df3['value'])  
plt.show()
```



## ■ 인구 공공데이터

### ● 2021년 1월 성별 / 연령별 인구 현황 데이터 가져오기

```
df = pd.read_csv(  
    'data/people_gender_202101.csv', encoding='cp949',  
    thousands=',')  
df.head()
```

|   | 행정구역                                   | 2021년<br>01월_남<br>_총인구<br>수 | 2021년<br>01월_남<br>_연령구<br>간인구<br>수 | 2021년<br>01월_남<br>_0~9<br>세 | 2021년<br>01월_남<br>_10~19<br>세 | 2021년<br>01월_남<br>_20~29<br>세 | 2021년<br>01월_남<br>_30~39<br>세 | 2021년<br>01월_남<br>_40~49<br>세 | 2021년<br>01월_남<br>_50~59<br>세 | 2021년<br>01월_남<br>_60~69<br>세 | ... |
|---|--|-----------------------------|------------------------------------|-----------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-----|
| 0 | 서울특별시<br>(1100000000)                  | 4695999                     | 4695999                            | 317563                      | 399754                        | 704862                        | 735813                        | 755311                        | 748598                        | 590203                        | ... |
| 1 | 서울특별시<br>종로구<br>(1111000000)           | 72528                       | 72528                              | 3767                        | 5443                          | 11493                         | 9883                          | 10800                         | 12765                         | 9979                          | ... |
| 2 | 서울특별시<br>종로구 청운<br>효자동<br>(1111051500) | 5784                        | 5784                               | 430                         | 618                           | 795                           | 772                           | 969                           | 922                           | 620                           | ... |
| 3 | 서울특별시<br>종로구 사직<br>동<br>(1111053000)   | 4273                        | 4273                               | 290                         | 306                           | 568                           | 619                           | 668                           | 695                           | 567                           | ... |
| 4 | 서울특별시<br>종로구 삼청<br>동<br>(1111054000)   | 1251                        | 1251                               | 47                          | 108                           | 164                           | 170                           | 179                           | 202                           | 187                           | ... |

5 rows × 27 columns

## ■ 인구 공공데이터

### ● 서울특별시 관악구 신림동(1162069500) 정보 확인

```
df2 = df[ df[ '행정구역' ] == '서울특별시 관악구 신림동(1162069500)' ]
df2
```

| 행정구역                                 | 2021<br>년01<br>월_남<br>_총인<br>구수 | 2021<br>년01<br>월_남<br>_연령<br>구간<br>인구<br>수 | 2021<br>년01<br>월_남<br>_0~9<br>세 | 2021년<br>01월_남<br>_10~19<br>세 | 2021년<br>01월_남<br>_20~29<br>세 | 2021년<br>01월_남<br>_30~39<br>세 | 2021년<br>01월_남<br>_40~49<br>세 | 2021년<br>01월_남<br>_50~59<br>세 | 2021년<br>01월_남<br>_60~69<br>세 | ... |
|--------------------------------------|---------------------------------|--|---------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-----|
| 서울특별시<br>관악구 신림<br>동<br>(1162069500) | 11398                           | 11398                                      | 178                             | 231                           | 4032                          | 3247                          | 1425                          | 999                           | 695                           | ... |

1 rows × 27 columns

## ■ 인구 공공데이터

### ● 2021년 1월 성별 / 연령별 인구 현황 그래프로 표현

- 불필요 컬럼 삭제

```
df2.drop(['행정구역'], axis=1, inplace=True)
df2.drop(
    ['2021년01월_남_총인구수', '2021년01월_남_연령구간인구수'], axis=1, inplace=True)
df2.drop(
    ['2021년01월_여_총인구수', '2021년01월_여_연령구간인구수'], axis=1, inplace=True)
df2.columns
```

```
Index(['2021년01월_남_0~9세', '2021년01월_남_10~19세', '2021년01월_남_20~29세',
      '2021년01월_남_30~39세', '2021년01월_남_40~49세', '2021년01월_남_50~59세',
      '2021년01월_남_60~69세', '2021년01월_남_70~79세', '2021년01월_남_80~89세',
      '2021년01월_남_90~99세', '2021년01월_남_100세 이상', '2021년01월_여_0~9세',
      '2021년01월_여_10~19세', '2021년01월_여_20~29세', '2021년01월_여_30~39세',
      '2021년01월_여_40~49세', '2021년01월_여_50~59세', '2021년01월_여_60~69세',
      '2021년01월_여_70~79세', '2021년01월_여_80~89세', '2021년01월_여_90~99세',
      '2021년01월_여_100세 이상'],
      dtype='object')
```

## ■ 인구 공공데이터

### ● 2021년 1월 성별 / 연령별 인구 현황 그래프로 표현

#### - 남자 데이터 확인

```
male = df2.iloc[:, 0:11]
male
```

|     | 2021년01월_남_0~9세 | 2021년01월_남_10~19세 | 2021년01월_남_20~29세 | 2021년01월_남_30~39세 | 2021년01월_남_40~49세 | 2021년01월_남_50~59세 | 2021년01월_남_60~69세 | 2021년01월_남_70~79세 | 2021년01월_남_80~89세 | 2021년01월_남_90~99세 | 2021년01월_남_100세 이상 |
|-----|-----------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------------|
| 355 | 178             | 231               | 4032              | 3247              | 1425              | 999               | 695               | 443               | 138               | 9                 | 1                  |

#### - 여자 데이터 확인

```
female = df2.iloc[:, 11:22]
female
```

|     | 2021년01월_여_0~9세 | 2021년01월_여_10~19세 | 2021년01월_여_20~29세 | 2021년01월_여_30~39세 | 2021년01월_여_40~49세 | 2021년01월_여_50~59세 | 2021년01월_여_60~69세 | 2021년01월_여_70~79세 | 2021년01월_여_80~89세 | 2021년01월_여_90~99세 | 2021년01월_여_100세 이상 |
|-----|-----------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|-------------------|--------------------|
| 355 | 181             | 243               | 4304              | 2378              | 1066              | 773               | 767               | 497               | 173               | 36                | 15                 |

## ■ 인구 공공데이터

### ● 2021년 1월 성별 / 연령별 인구 현황 그래프로 표현

- 남자 데이터 정리

```
male2 = male.melt()
male2.drop('variable', axis=1, inplace=True)
male2['value'] = male2['value'].apply(lambda x : x * -1)
male2
```

|    | value |
|----|-------|
| 0  | -178  |
| 1  | -231  |
| 2  | -4032 |
| 3  | -3247 |
| 4  | -1425 |
| 5  | -999  |
| 6  | -695  |
| 7  | -443  |
| 8  | -138  |
| 9  | -9    |
| 10 | -1    |

수평 막대 그래프를 이용하여 좌 / 우로 인구 분포 표현



## ■ 인구 공공데이터

- 2021년 1월 성별 / 연령별 인구 현황 그래프로 표현
  - 남자 데이터 정리

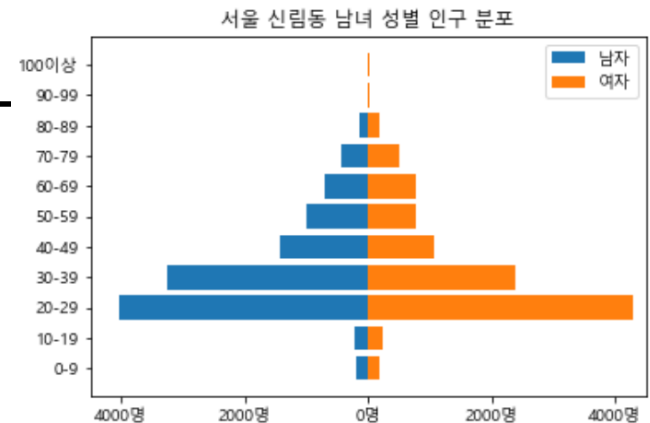
```
female2 = female.melt()  
female2.drop('variable', axis=1, inplace=True)  
female2
```

|    | value |
|----|-------|
| 0  | 181   |
| 1  | 243   |
| 2  | 4304  |
| 3  | 2378  |
| 4  | 1066  |
| 5  | 773   |
| 6  | 767   |
| 7  | 497   |
| 8  | 173   |
| 9  | 36    |
| 10 | 15    |

## ■ 인구 공공데이터

### ● 2021년 1월 성별 / 연령별 인구 현황 그래프로 표현

```
import matplotlib.pyplot as plt
plt.rc('font', family='Malgun Gothic')
plt.rcParams['axes.unicode_minus'] = False
plt.title('서울 신림동 남녀 성별 인구 분포')
plt.barh(range(11), male2['value'].tolist(), label='남자')
plt.barh(range(11), female2['value'].tolist(), label='여자')
plt.yticks(
    range(11),
    ['0-9', '10-19', '20-29', '30-39', '40-49', '50-59', '60-69',
     '70-79', '80-89', '90-99', '100이상'])
plt.xlim(-4500, 4500)
plt.xticks(
    range(-4000, 4001, 2000),
    ['4000명', '2000명', '0명', '2000명', '4000명'])
plt.legend()
plt.show()
```



## ■ 인구 공공데이터

### ● 제주도 성별 인구 비율 표현하기

```
import pandas as pd
import matplotlib.pyplot as plt

df = pd.read_csv(
    'data/people_gender_202101.csv', encoding='cp949', thousands=',')
df2 = df[ df[ '행정구역' ] == '제주특별자치도 (5000000000)' ]
df2
```

| 행정구역                        | 2021년<br>01월_남_총<br>인구수 | 2021년<br>01월_남_연<br>령구간<br>인구수 | 2021<br>년01<br>월_남<br>_0~9<br>세 | 2021년<br>01월_남<br>_10~19<br>세 | 2021년<br>01월_남<br>_20~29<br>세 | 2021년<br>01월_남<br>_30~39<br>세 | 2021년<br>01월_남<br>_40~49<br>세 | 2021년<br>01월_남<br>_50~59<br>세 | 2021년<br>01월_남<br>_60~69<br>세 | ... |
|-----------------------------|-------------------------|--------------------------------|---------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-------------------------------|-----|
| 제주특별자치<br>도<br>(5000000000) | 338553                  | 338553                         | 30833                           | 36894                         | 43655                         | 41539                         | 57902                         | 58463                         | 39783                         | ... |

1 rows × 27 columns

## ■ 인구 공공데이터

### ● 제주도 성별 인구 비율 표현하기

```
male = df2['2021년01월_남_총인구수']  
female = df2['2021년01월_여_총인구수']  
plt.rc('font', family='Malgun Gothic')  
color = ['magenta', 'cyan']  
plt.pie(  
    [male.sum(), female.sum()], labels=['남', '여'],  
    colors=color, startangle=90, autopct='%1f%%')  
plt.show()
```

