

## 경찰청 범죄 데이터 시각화

### 라이브러리 및 불러오기 및 한글 사용을 위한 준비

```
In [1]: ▶ import matplotlib as mpl          # 기본 설정 만지는 용도
import matplotlib.pyplot as plt          # 그래프 그리는 용도
import matplotlib.font_manager as fm     # 폰트 관련 용도
import numpy as np
```

```
In [2]: ▶ ### 나눔 고딕 설치
!apt-get update -qq # 설치를 업데이트 -qq : 로그를 최소한으로
!apt-get install fonts-nanum* -qq # 설치한다. fonts-nanum* => ttf-nanum,
```

```
W: Skipping acquire of configured file 'main/source/Sources' as repository
'https://r2u.stat.illinois.edu/ubuntu jammy InRelease' does not seem to pro
vide it (sources.list entry misspelt?)
Selecting previously unselected package fonts-nanum.
(Reading database ... 123594 files and directories currently installed.)
Preparing to unpack .../fonts-nanum_20200506-1_all.deb ...
Unpacking fonts-nanum (20200506-1) ...
Selecting previously unselected package fonts-nanum-coding.
Preparing to unpack .../fonts-nanum-coding_2.5-3_all.deb ...
Unpacking fonts-nanum-coding (2.5-3) ...
Selecting previously unselected package fonts-nanum-eco.
Preparing to unpack .../fonts-nanum-eco_1.000-7_all.deb ...
Unpacking fonts-nanum-eco (1.000-7) ...
Selecting previously unselected package fonts-nanum-extra.
Preparing to unpack .../fonts-nanum-extra_20200506-1_all.deb ...
Unpacking fonts-nanum-extra (20200506-1) ...
Setting up fonts-nanum-extra (20200506-1) ...
Setting up fonts-nanum (20200506-1) ...
Setting up fonts-nanum-coding (2.5-3) ...
Setting up fonts-nanum-eco (1.000-7) ...
Processing triggers for fontconfig (2.13.1-4.2ubuntu5) ...
```

```

In [3]: ▶ import matplotlib.pyplot as plt
import matplotlib.font_manager as fm
import matplotlib as mpl

# 폰트 파일 경로
path = '/usr/share/fonts/truetype/nanum/NanumGothicEco.ttf'

# 폰트 프로퍼티 생성
font_prop = fm.FontProperties(fname=path, size=10)
font_name = font_prop.get_name()
print(font_name) # NanumGothic Eco

# 폰트 매니저에 폰트 추가
fm.fontManager.addfont(path)

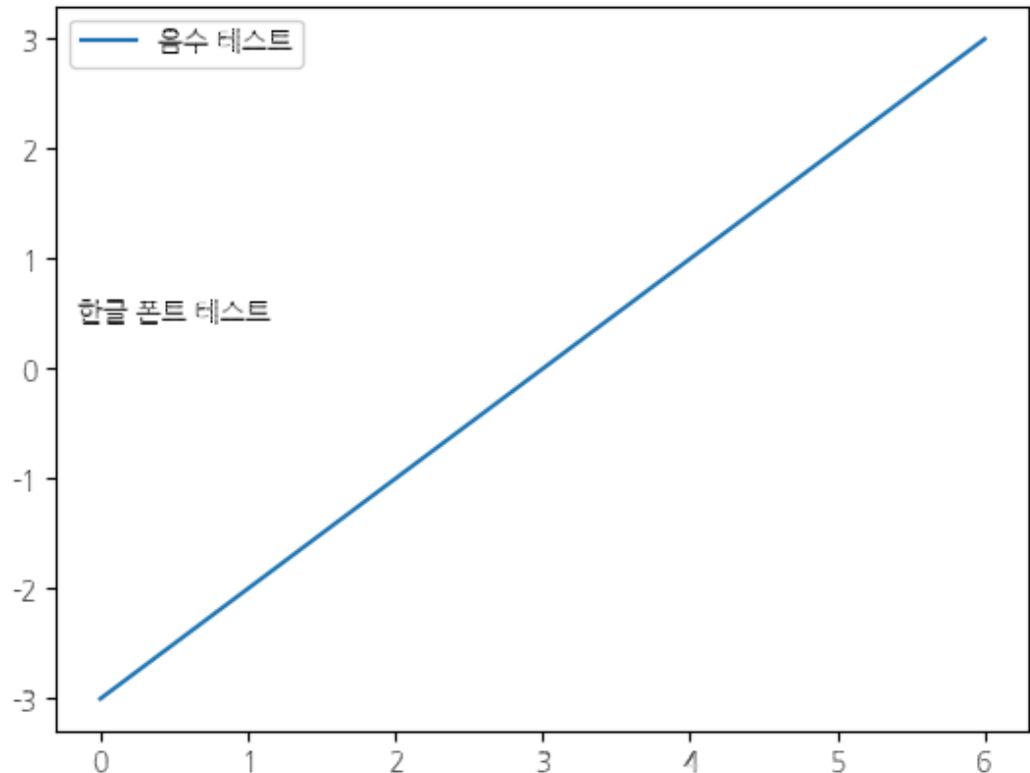
# matplotlib에 폰트 설정
plt.rc('font', family=font_name)

# 음수 표시되도록 설정
mpl.rcParams['axes.unicode_minus'] = False

# 예제 플롯
plt.figure()
plt.plot([-3, -2, -1, 0, 1, 2, 3], label='음수 테스트')
plt.text(0.5, 0.5, '한글 폰트 테스트', ha='center', va='center', fontproperties=font_prop)
plt.legend()
plt.show()

```

NanumGothic Eco



## 범죄대분류별 서울 지역 범죄 건수

```
In [4]: ▶ import pandas as pd
import matplotlib.pyplot as plt

# CSV 파일 경로 설정
file_path = '경찰청_범죄발생_data_20221231.csv'

# 데이터 불러오기 (euc-kr 인코딩 사용)
data = pd.read_csv(file_path, encoding='euc-kr')

# 범죄대분류별 서울 지역의 범죄 건수 합계 계산
crime_data_seoul = data.groupby('범죄대분류')['서울'].sum()

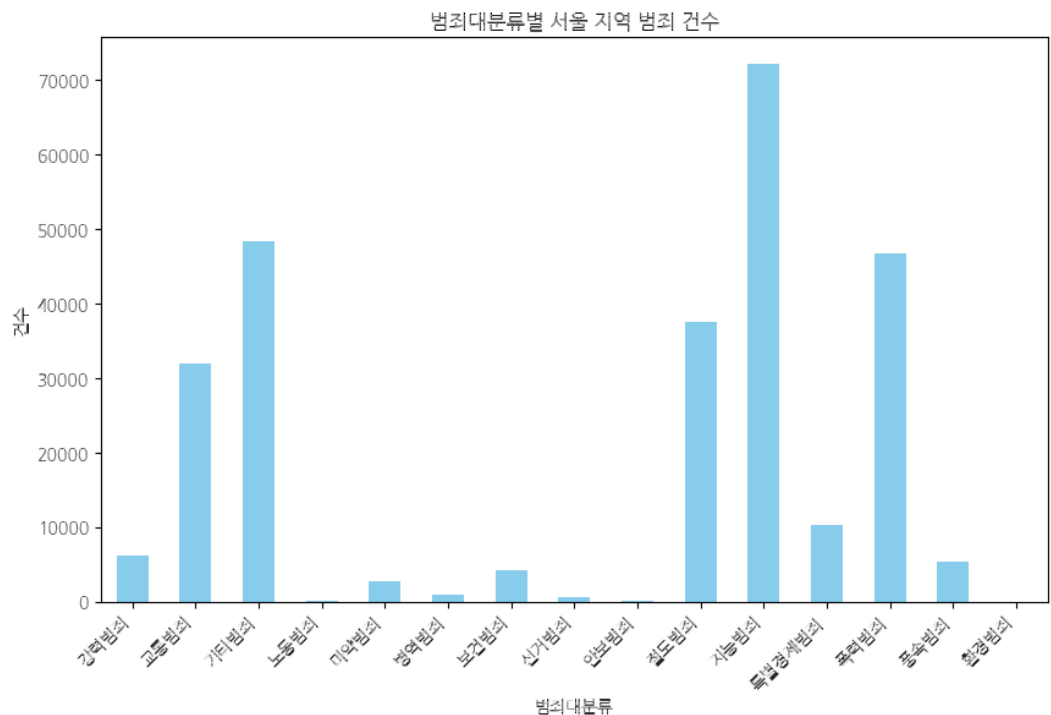
# 그래프 크기 설정
plt.figure(figsize=(10, 6))

# 막대그래프 생성
crime_data_seoul.plot(kind='bar', color='skyblue')

# 그래프 제목 및 레이블 설정
plt.title('범죄대분류별 서울 지역 범죄 건수')
plt.xlabel('범죄대분류')
plt.ylabel('건수')

# X축 레이블 회전 및 정렬 설정
plt.xticks(rotation=45, ha='right')

# 그래프 출력
plt.show()
```



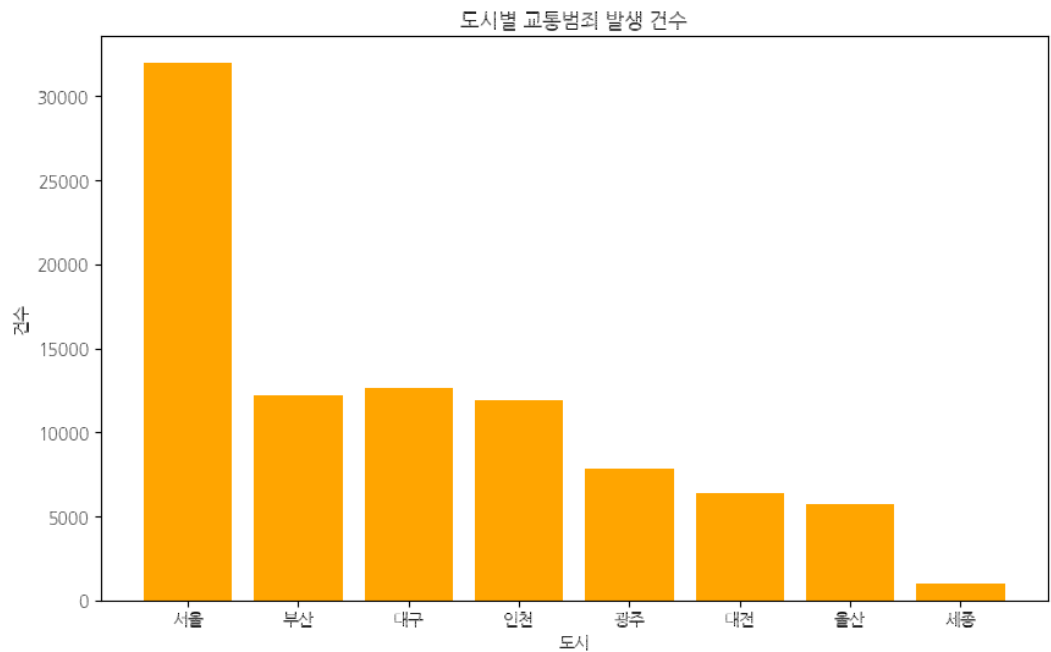
## 각 도시별 교통범죄 건수 발생

```
In [10]: ▶ # Step 1: Filter the data for '교통범죄'
traffic_crime_data = data[data['범죄대분류'] == '교통범죄']

# Step 2: Select cities to visualize
# Let's use the same cities as before for consistency
cities = ['서울', '부산', '대구', '인천', '광주', '대전', '울산', '세종']

# Aggregate traffic crime data by summing all traffic-related crimes for each city
traffic_crime_data = traffic_crime_data[cities].sum()

# Step 3: Visualize the data
plt.figure(figsize=(10, 6))
plt.bar(traffic_crime_data.index, traffic_crime_data.values, color='orange')
plt.title('도시별 교통범죄 발생 건수')
plt.xlabel('도시')
plt.ylabel('건수')
plt.show()
```



In [ ]: ▶

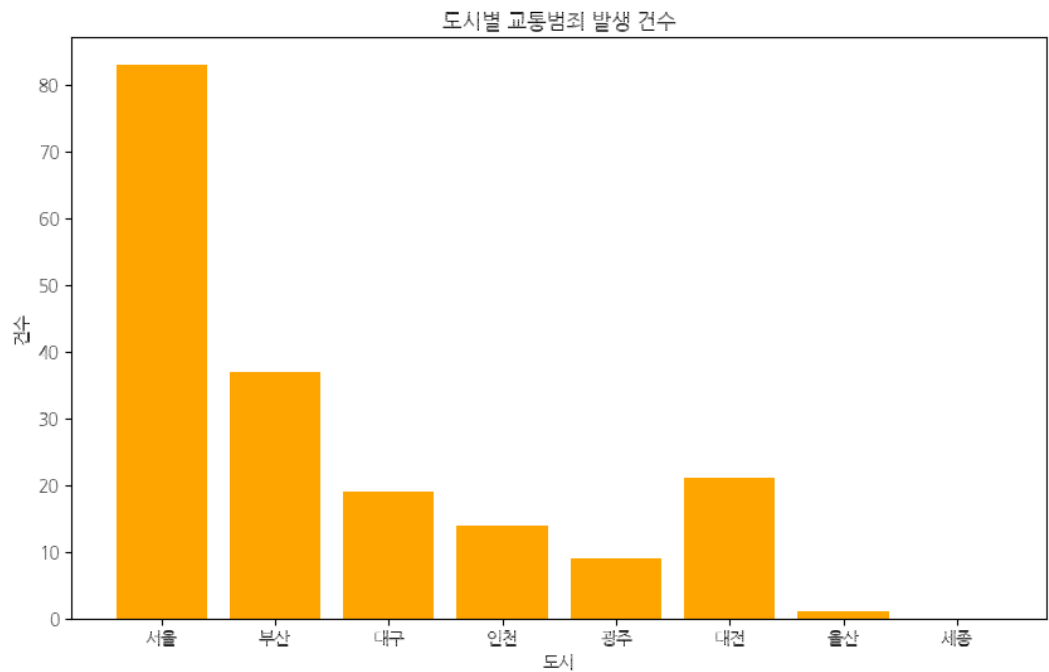
## 각 도시별 살인 건수 발생

```
In [11]: ▶ # Step 1: Filter the data for '교통범죄'
traffic_crime_data = data[data['범죄대분류'] == '노동범죄']

# Step 2: Select cities to visualize
# Let's use the same cities as before for consistency
cities = ['서울', '부산', '대구', '인천', '광주', '대전', '울산', '세종']

# Aggregate traffic crime data by summing all traffic-related crimes for each
traffic_crime_data = traffic_crime_data[cities].sum()

# Step 3: Visualize the data
plt.figure(figsize=(10, 6))
plt.bar(traffic_crime_data.index, traffic_crime_data.values, color='orange')
plt.title('도시별 노동범죄 발생 건수')
plt.xlabel('도시')
plt.ylabel('건수')
plt.show()
```



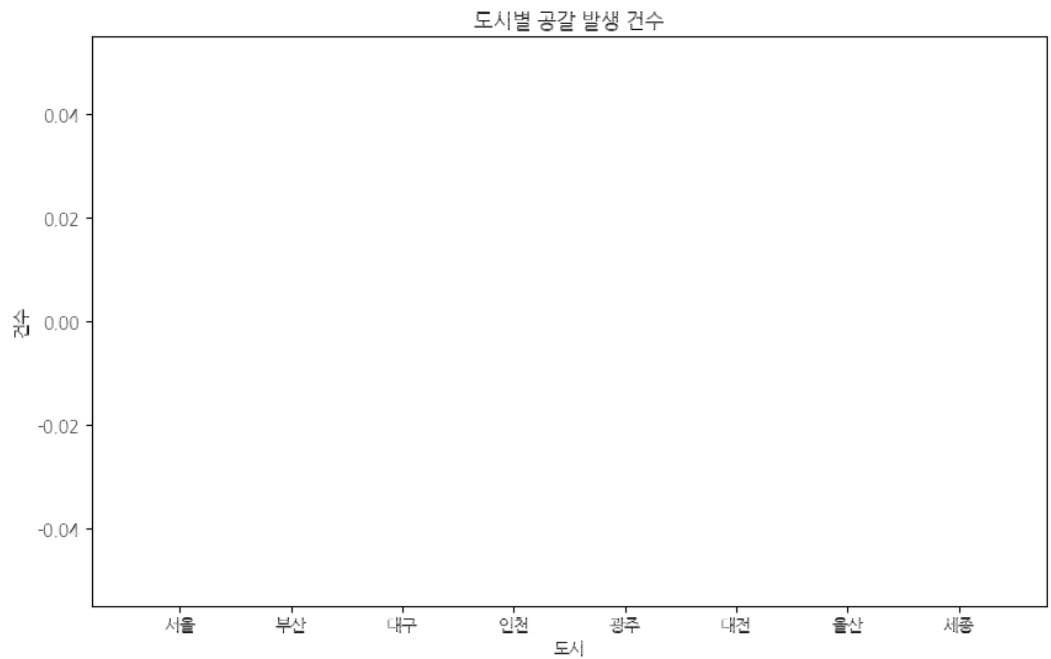
## 안된 케이스

```
In [12]: ▶ # Step 1: Filter the data for '교통범죄'
traffic_crime_data = data[data['범죄대분류'] == '공갈']

# Step 2: Select cities to visualize
# Let's use the same cities as before for consistency
cities = ['서울', '부산', '대구', '인천', '광주', '대전', '울산', '세종']

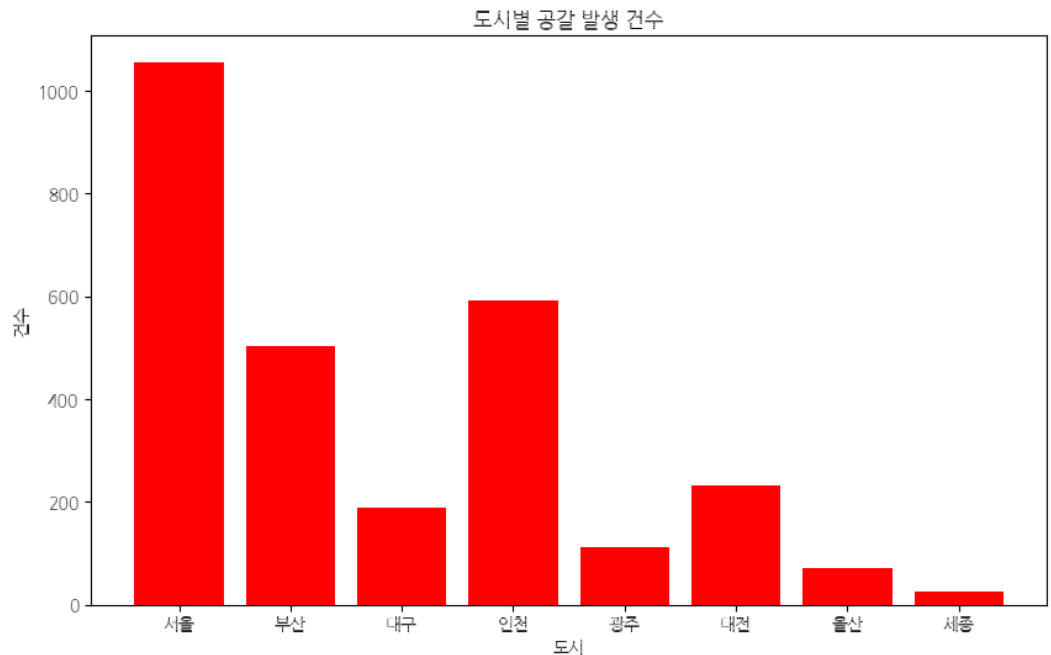
# Aggregate traffic crime data by summing all traffic-related crimes for each city
traffic_crime_data = traffic_crime_data[cities].sum()

# Step 3: Visualize the data
plt.figure(figsize=(10, 6))
plt.bar(traffic_crime_data.index, traffic_crime_data.values, color='orange')
plt.title('도시별 공갈 발생 건수')
plt.xlabel('도시')
plt.ylabel('건수')
plt.show()
```



```
In [15]: ▶ # 4. 공갈 발생 건수 시각화
extortion_data = data[data['범죄중분류'] == '공갈']
extortion_data = extortion_data[cities].T # 도시별로 전환

plt.figure(figsize=(10, 6))
plt.bar(extortion_data.index, extortion_data.iloc[:, 0], color='red')
plt.title('도시별 공갈 발생 건수')
plt.xlabel('도시')
plt.ylabel('건수')
plt.show()
```



## 도시별 범죄율 비교 분석 가능할까?

- 각 도시의 범죄 발생 건수를 도시 인구 수로 나누어 범죄율을 계산해야 합니다. 이는 범죄 발생 건수 / 도시 인구 수 \* 100,000으로 계산하여, 도시별로 비교 가능한 지표를 만들 수 있습니다. 하지만 현재 데이터에는 도시 인구 수 정보가 포함되어 있지 않기 때문에, 추가적인 인구 데이터가 필요합니다.

```

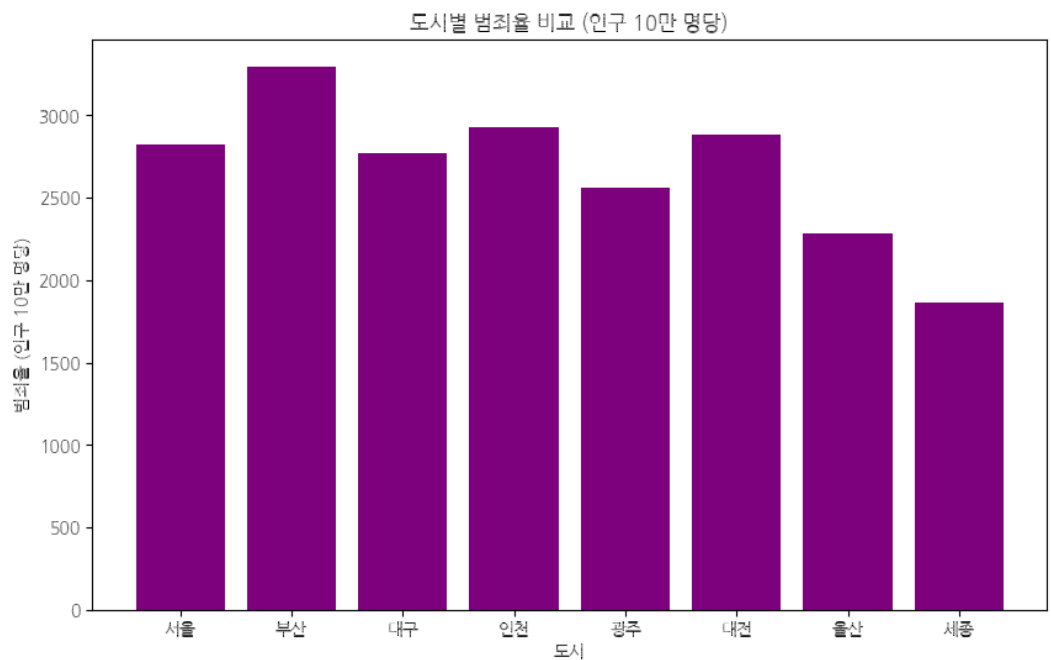
In [16]: ▶ # 1. 도시 인구 데이터 (예시)
population_data = {
    '서울': 9500000,
    '부산': 3400000,
    '대구': 2400000,
    '인천': 2900000,
    '광주': 1500000,
    '대전': 1500000,
    '울산': 1200000,
    '세종': 350000
}

# 2. 도시별 범죄 발생 총합 계산
total_crimes = data[cities].sum()

# 3. 도시별 범죄율 계산
crime_rate = (total_crimes / pd.Series(population_data)) * 100000

# 4. 범죄율 시각화
plt.figure(figsize=(10, 6))
plt.bar(crime_rate.index, crime_rate.values, color='purple')
plt.title('도시별 범죄율 비교 (인구 10만 명당)')
plt.xlabel('도시')
plt.ylabel('범죄율 (인구 10만 명당)')
plt.show()

```



In [ ]: ▶