

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

데이터프레임 시각화

데이터 프레임이 내부적으로 `pyplot` 객체를 초기화 하기 때문에 `plt.figure()` 를 호출할 필요가 없다.

#01. 작업준비

1) 패키지 참조

```
from pandas import read_excel
from matplotlib import pyplot as plt
```

2) 데이터 가져오기

년도별 교통사고 발생건수, 사망자수, 부상자수 데이터

데이터 출처: KOSIS 국가통계포털

```
df = read_excel("https://data.hossam.kr/D01/traffic_acc_year.xlsx", index_col=0)
```

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

	발생건수	사망자수	부상자수
년도			
2005	214171	6376	342233
2006	213745	6327	340229
2007	211662	6166	335906
2008	215822	5870	338962
2009	231990	5838	361875
2010	226878	5505	352458
2011	221711	5229	341391
2012	223656	5392	344565
2013	215354	5092	328711
2014	223552	4762	337497
2015	232035	4621	350400
2016	220917	4292	331720
2017	216335	4185	322829
2018	217148	3781	323037

```
df.describe()
```

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

	발생건수	사망자수	부상자수
count	14.000000	14.000000	14.000000
mean	220355.428571	5245.428571	339415.214286
std	6604.908929	829.784097	10981.076045
min	211662.000000	3781.000000	322829.000000
25%	215471.000000	4656.250000	332766.500000
50%	219032.500000	5310.500000	339595.500000
75%	223630.000000	5862.000000	343982.000000
max	232035.000000	6376.000000	361875.000000

3) 그래프 전역 설정

```
plt.rcParams['font.family'] = 'Malgun Gothic' # 윈도우 전용
#plt.rcParams['font.family'] = 'AppleGothic' # 맥 전용
plt.rcParams["font.size"] = 10
plt.rcParams["figure.figsize"] = (10, 5)
plt.rcParams['axes.unicode_minus'] = False
```

#02. Line Plot

1. 특정 컬럼에 대한 시각화

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#03. Bar Plot

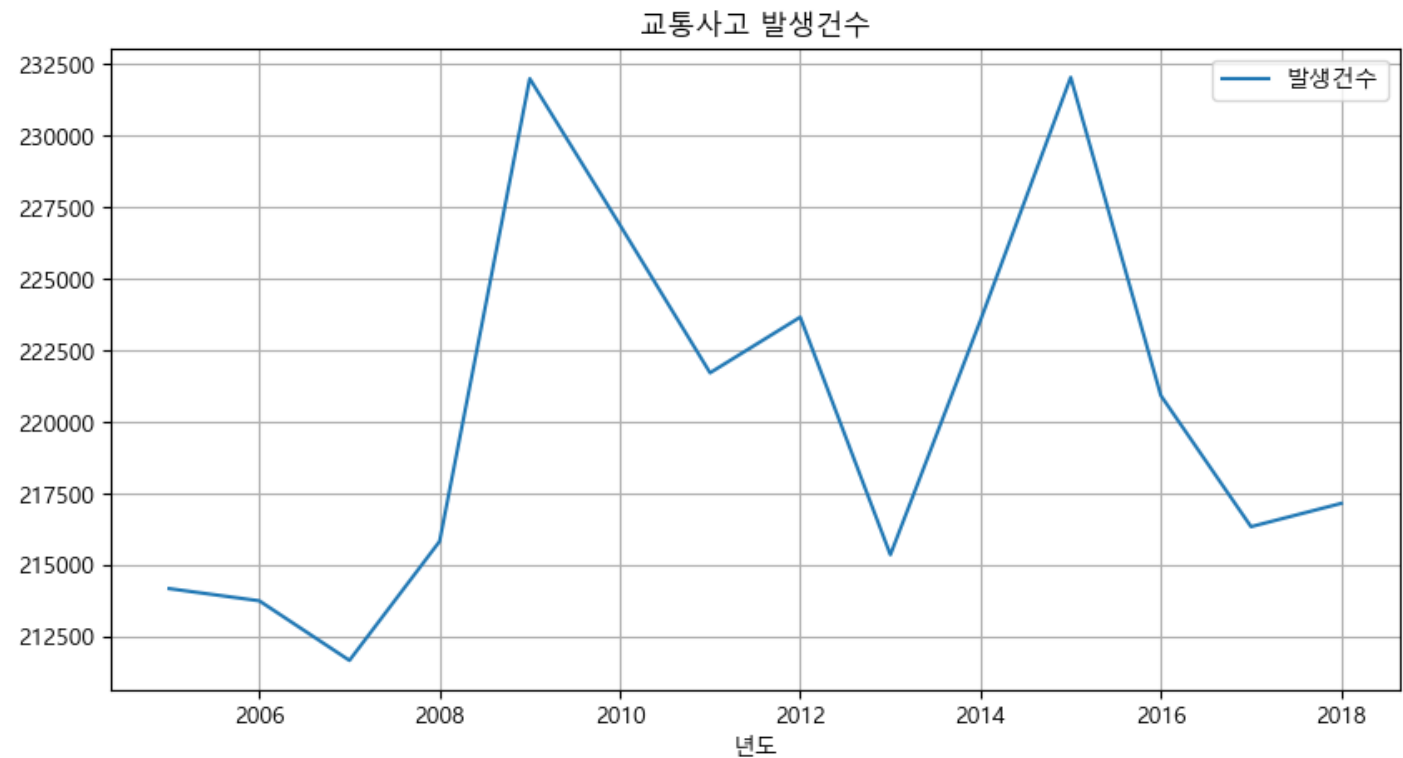
1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

기본적으로 DataFrame의 컬럼 이름이 label이 되지만 `plot()` 메서드에 label 파라미터를 설정하여 다른 이름으로 변경할 수 있다.

```
df['발생건수'].plot(label='교통사고')
plt.grid()
plt.legend()
plt.title("교통사고 발생건수")
plt.show()
plt.close()
```



2. 2개 이상 컬럼에 대한 시각화

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

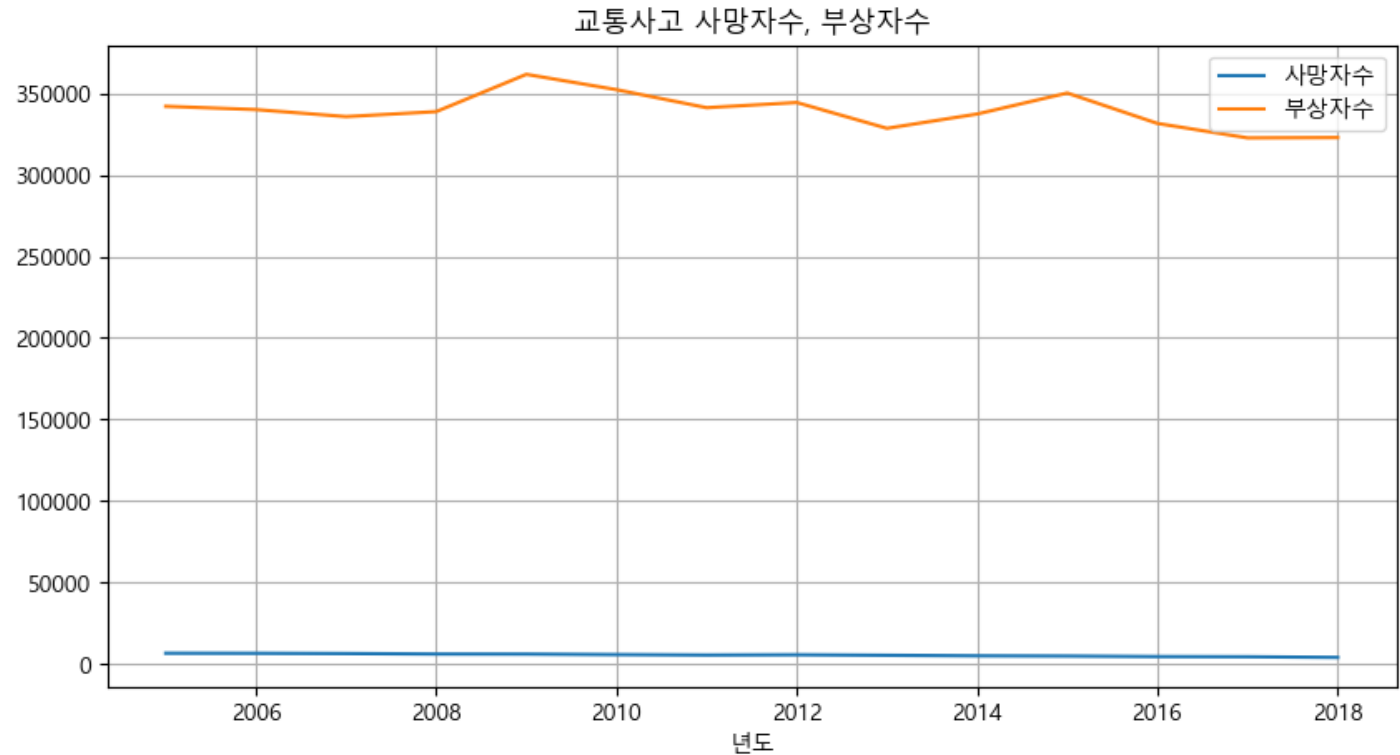
#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

```
df.filter(['사망자수', '부상자수']).plot()
plt.grid()
plt.legend()
plt.title("교통사고 사망자수, 부상자수")
plt.show()
plt.close()
```



3. 전체 컬럼에 대한 시각화

```
df.plot()
plt.grid()
```

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

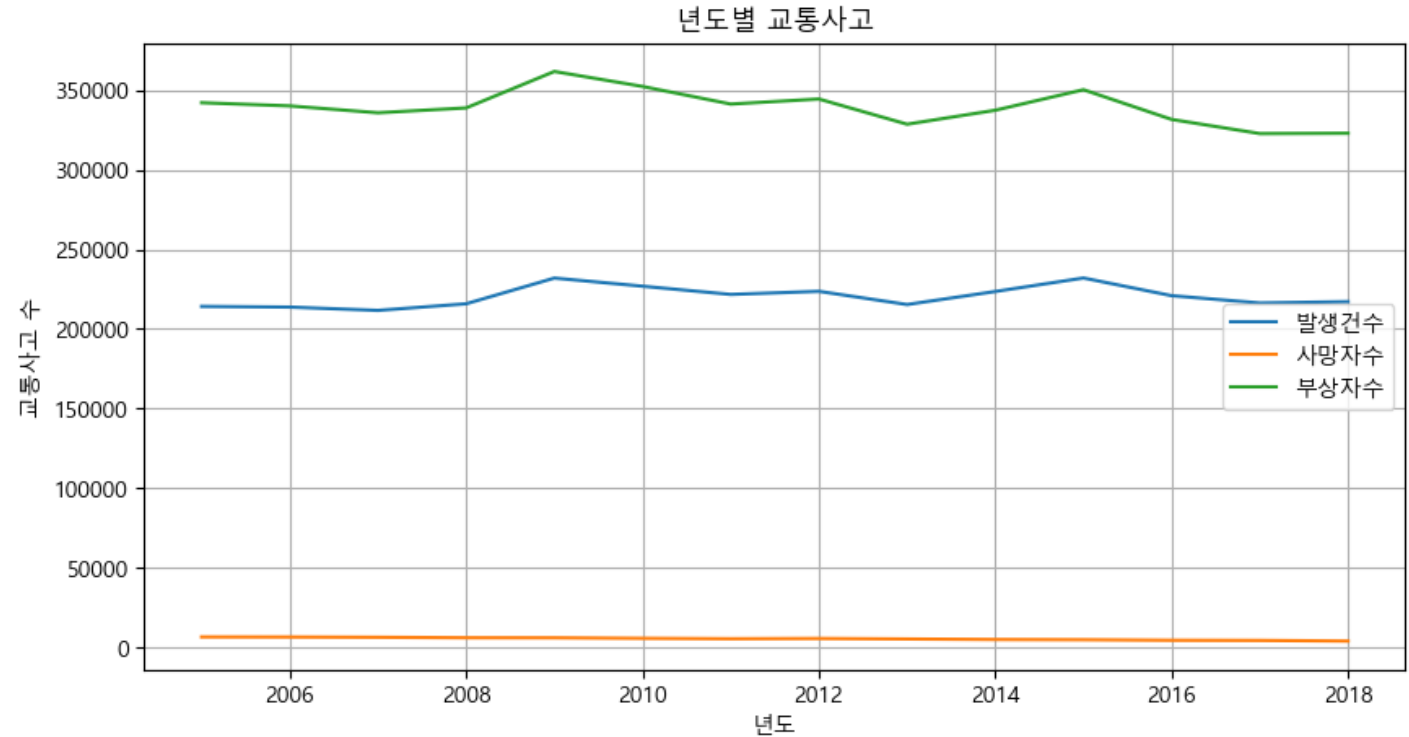
#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

```
plt.legend()
plt.title("년도별 교통사고")
plt.ylabel("교통사고 수")
plt.show()
plt.close()
```



#03. Bar Plot

1. 특정 컬럼에 대한 시각화

```
df['발생건수'].plot.bar(rot=45)
plt.grid()
```

데이터프레임 시각화

#01. 작업준비

1) 패키지 참조

2) 데이터 가져오기

3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화

2. 2개 이상 컬럼에 대한 시각화

3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화

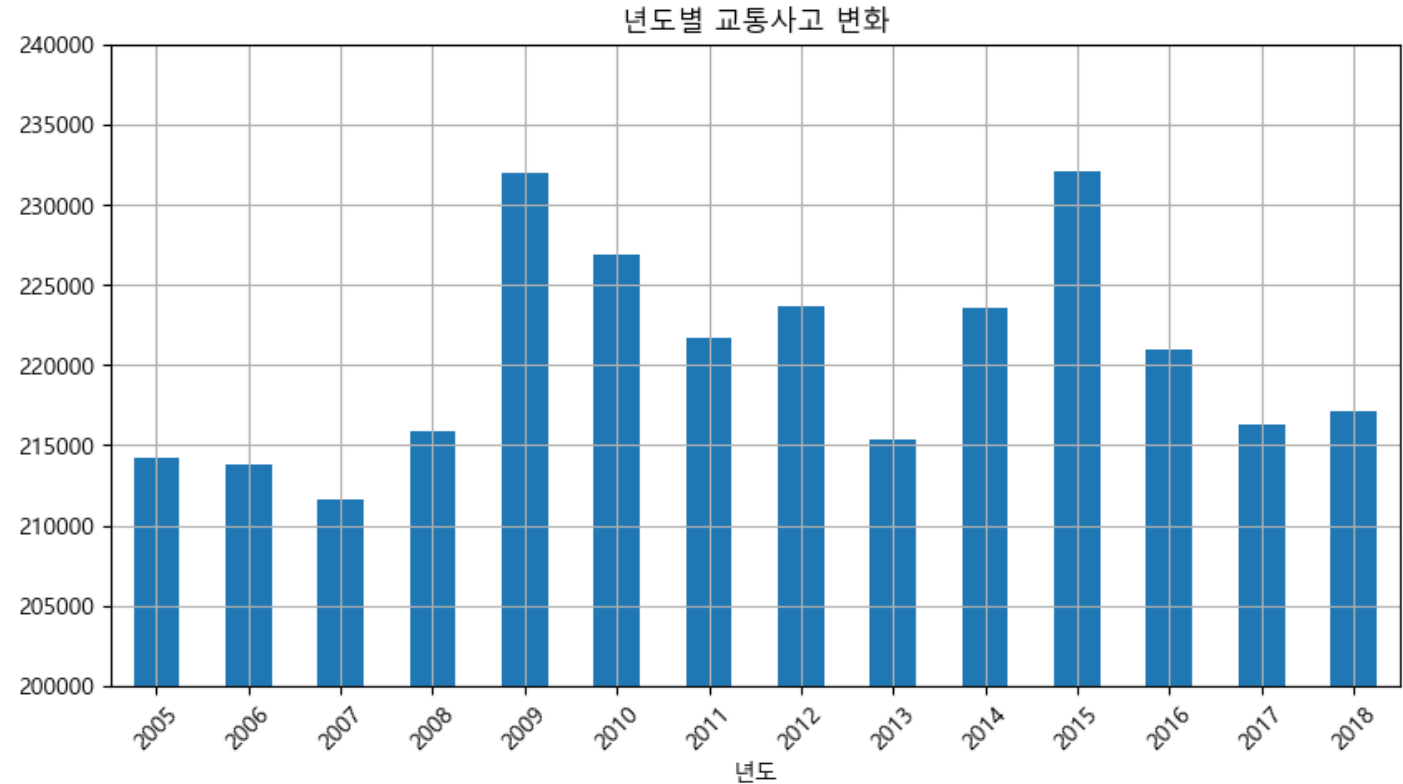
2. 2개 이상 컬럼에 대한 시각화

3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

```
plt.title("년도별 교통사고 변화")
plt.ylim(200000, 240000)
plt.show()
plt.close()
```



2. 2개 이상 컬럼에 대한 시각화

```
df.filter(['발생건수', '부상자수']).plot.bar(rot=45)
plt.grid()
plt.legend()
plt.title("년도별 교통사고 변화")
```

데이터프레임 시각화

#01. 작업준비

1) 패키지 참조

2) 데이터 가져오기

3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화

2. 2개 이상 컬럼에 대한 시각화

3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화

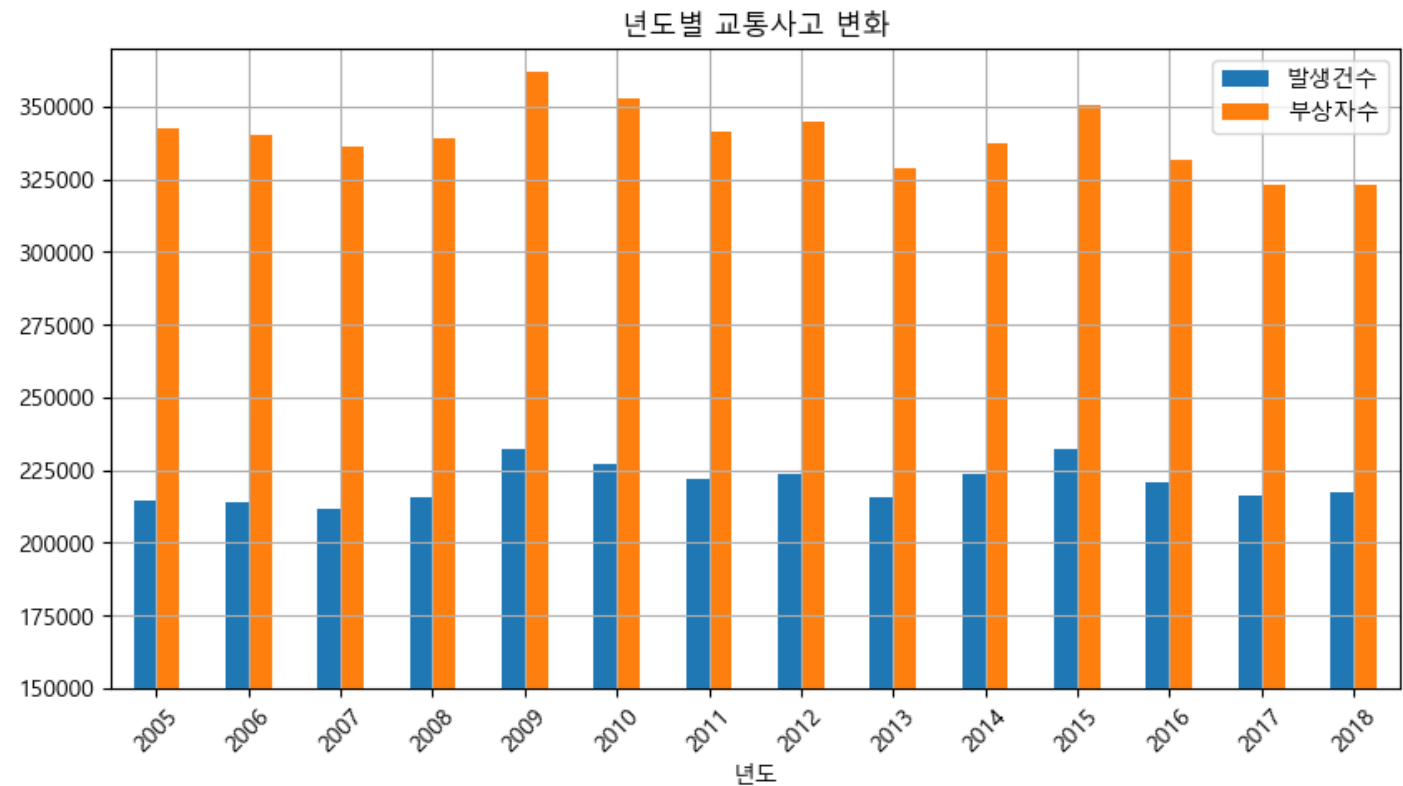
2. 2개 이상 컬럼에 대한 시각화

3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

```
plt.ylim(150000, 370000)
plt.show()
plt.close()
```



3. 전체 컬럼에 대한 시각화

```
df.plot.bar(rot=45)
plt.grid()
plt.legend()
plt.title("년도별 교통사고 변화")
```


데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

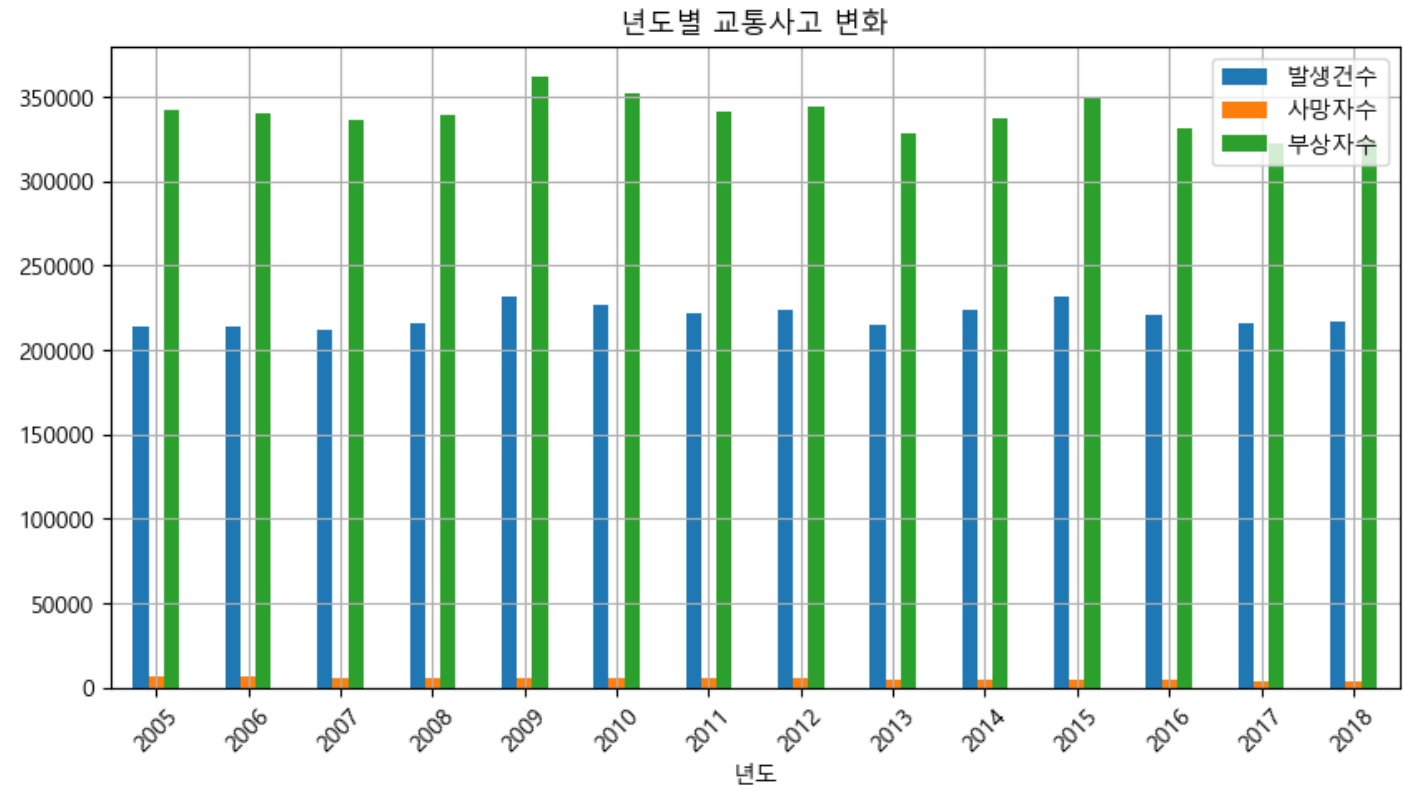
#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

```
plt.show()
plt.close()
```



#04. 산점도 그래프

```
df.plot.scatter(x='발생건수', y='부상자수', color="#0066ff")
plt.grid()
plt.show()
plt.close()
```

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

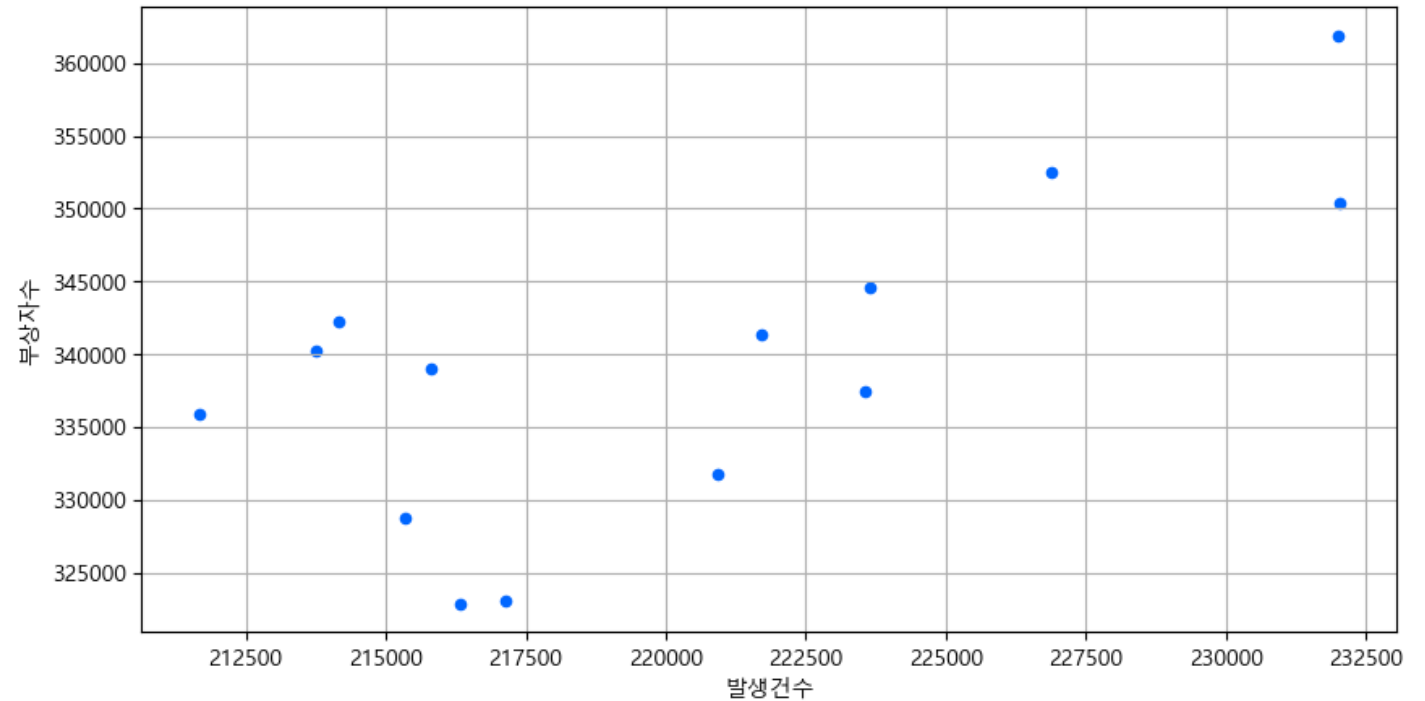
1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프



#05. 파이 그래프

dataframe의 index가 파이 그래프의 각 조각이 된다.

```
df['사망자수'].plot.pie(autopct='%0.1f%%')
plt.show()
plt.close()
```

데이터프레임 시각화

#01. 작업준비

- 1) 패키지 참조
- 2) 데이터 가져오기
- 3) 그래프 전역 설정

#02. Line Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#03. Bar Plot

1. 특정 컬럼에 대한 시각화
2. 2개 이상 컬럼에 대한 시각화
3. 전체 컬럼에 대한 시각화

#04. 산점도 그래프

#05. 파이 그래프

