

# 서울시 범죄현황 분석 part3 - 연도별 현황 by ham

Auto Mobile Robot

Exported on 02/23/2024

## Table of Contents

1	서울시 19년간 구별 범죄현황 데이터 .....	4
2	glob 모듈 .....	5
3	glob의 결과는 list형으로 나타남 .....	6
4	의도하지 않는 대상도 있다 .....	7
5	범위를 한정할 수 있다 .....	8
6	원하는 대상 파일만 리스트로 저장가능하다 .....	9
7	대상 파일을 list로 저장했으니 for 문에서 사용할 수 있다 .....	10
8	반복문 안에 들어갈 코드를 연습하자 .....	11
9	결과 .....	12
10	파일명에서 연도를 가져와야 한다 .....	13
11	split 명령을 사용하면 간편하다 .....	14
12	이렇게 파일경로+파일명에서 파일명만 가져올 수 있다 .....	15
13	이렇게 파일경로+파일명에서 파일명의 연도만 가져올 수 있다 .....	16
14	19개 데이터를 한 번에 읽자 .....	17
15	데이터 합치기 .....	18
16	pivot을 이용하면??? .....	19
17	이렇게 마법처럼 정리된다 .....	20
18	한글 폰트 설치 .....	21
19	heatmap으로 그려보자 .....	22
20	19년간 서울시 구별 살인사건의 변화 .....	23
21	살인사건 데이터를 그냥 그려보자 .....	24
22	흠 어떤 의미인지 모르겠다 .....	25
23	강남 서초 영등포의 살인 사건의 변화 .....	26
24	강남 서초 영등포의 살인 사건의 변화 .....	27
25	폭력사건만 몇몇 구에 대해 관찰해보기 .....	28
26	폭력사건만 몇몇 구에 대해 관찰해보기 .....	29
27	성범죄에 대해서 관찰해보기 .....	30

28 성범죄에 대해 연도별 상황 .....	31
29 첨부파일 .....	32

# 1 서울시 19년간 구별 범죄현황 데이터

내 드라이브

공유 문서함

최근 문서함

중요


















휴지통

백업

저장용량

100GB 중 30.4GB 사용


저장용량 업그레이드

	2000년.csv	나	2019. 10. 31. 나
	2001년.csv	나	2019. 10. 31. 나
	2002년.csv	나	2019. 10. 31. 나
	2003년.csv	나	2019. 10. 31. 나
	2004년.csv	나	2019. 10. 31. 나
	2005년.csv	나	2019. 10. 31. 나
	2006년.csv	나	2019. 10. 31. 나
	2007년.csv	나	2019. 10. 31. 나
	2008년.csv	나	2019. 10. 31. 나
	2009년.csv	나	2019. 10. 31. 나
	2010년.csv	나	2019. 10. 31. 나
	2011년.csv	나	2019. 10. 31. 나
	2012년.csv	나	2019. 10. 31. 나
	2013년.csv	나	2019. 10. 31. 나
	2014년.csv	나	2019. 10. 31. 나
	2015년.csv	나	2019. 10. 31. 나
	2016년.csv	나	2019. 10. 31. 나
	2017년.csv	나	2019. 10. 31. 나
	2018년.csv	나	2019. 10. 31. 나

## 2 glob 모듈

```
from glob import glob
```

```
glob('/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/*')
```



- glob 모듈은 파일의 경로에 접근해서 그 결과를 list형으로 반환한다
- 위 경로는 타이핑이 아니라 복사해서 사용하자

### 3 glob의 결과는 list형으로 나타남

```
[ '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2000년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2002년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2001년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2003년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2007년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2009년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2004년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2005년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2006년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2008년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2010년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2012년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2013년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2011년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2015년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2017년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2016년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2014년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2018년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/crime_station.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/seoul_crime_result.csv']
```

## 4 의도하지 않는 대상도 있다

```
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2013년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2011년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2015년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2017년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2016년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2014년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2018년.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/crime_station.csv',  
'/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/seoul_crime_result.csv']
```

## 5 범위를 한정할 수 있다

```
target_path = '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data'
file_list = glob(target_path+'20*.csv')
file_list
```



## 6 원하는 대상 파일만 리스트로 저장가능하다

```
[ '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2000년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2002년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2001년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2003년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2007년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2009년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2004년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2005년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2006년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2008년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2010년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2012년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2013년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2011년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2015년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2017년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2016년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2014년.csv',
  '/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2018년.csv' ]
```

## 7 대상 파일을 list로 저장했으니 for 문에서 사용할 수 있다

```
for each_file in file_list:  
    print(each_file)
```

## 8 반복문 안에 들어갈 코드를 연습하자

```
import pandas as pd
import numpy as np

crime_raw_data = pd.read_csv(file_list[0],
                             usecols=[0,1,2,3], encoding='euc-kr')

crime_station = crime_raw_data.pivot_table(
    crime_raw_data, index=["구분"], columns=["죄종", "발생검거"], aggfunc=[np.sum])

crime_station.columns = ['강간검거', '강간', '강도검거', '강도', '살인검거',
                        '살인', '절도검거', '절도', '폭력검거', '폭력']


crime_station['year'] = 2000
crime_station.head()
```

## 9 결과

	강간검거	강간	강도검거	강도	살인검거	살인	절도검거	절도	폭력검거	폭력	year
구분											
강남	70	78	72	146	4	3	523	2653	4345	4892	2000
강동	58	62	40	49	11	12	671	1838	4333	4716	2000
강북	48	47	48	47	8	9	458	859	3446	3624	2000
강서	64	75	55	75	7	6	509	1559	3988	4361	2000
관악	44	59	36	56	7	8	349	1764	2658	2921	2000

## 10 파일명에서 연도를 가져와야 한다

```
crime_station['year'] = 2000
crime_station.head()
```



	강간검거	강간	강도검거	강도	살인검거	살인	절도검거	절도	폭력검거	폭력	year
구분											
강남	70	78	72	146	4	3	523	2653	4345	4892	2000
강동	58	62	40	49	11	12	671	1838	4333	4716	2000
강북	48	47	48	47	8	9	458	859	3446	3624	2000
강서	64	75	55	75	7	6	509	1559	3988	4361	2000

```
/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2015년.csv
/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2017년.csv
/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2016년.csv
/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2014년.csv
/content/drive/My Drive/Colab Notebooks/데이터 사이언스 입문/Data/2018년.csv
```

## 11 split 명령을 사용하면 간편하다

```
for each_file in file_list:
    print(each_file.split('/'))
```

```
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2000년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2002년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2001년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2003년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2007년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2009년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2004년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2005년.csv']
[',', 'content', 'drive', 'My Drive', 'Colab Notebooks', '데이터 사이언스 입문', 'Data', '2006년.csv']
```

## 12 이렇게 파일경로+파일명에서 파일명만 가져올 수 있다

```
for each_file in file_list:  
    print(each_file.split('/')[-1])
```

```
2000년.csv  
2002년.csv  
2001년.csv  
2003년.csv  
2007년.csv  
2009년.csv  
2004년.csv  
2005년.csv  
2006년.csv  
2008년.csv  
2010년.csv  
2012년.csv  
2013년.csv  
2011년.csv  
2015년.csv  
2017년.csv  
2016년.csv  
2014년.csv  
2018년.csv
```

## 13 이렇게 파일경로+파일명에서 파일명의 연도만 가져올 수 있다

```
for each_file in file_list:  
    print(each_file.split('/')[-1][:4])
```

```
2000  
2002  
2001  
2003  
2007  
2009  
2004  
2005  
2006  
2008  
2010  
2012  
2013  
2011  
2015  
2017  
2016  
2014  
2018
```



## 14 19개 데이터를 한 번에 읽자

```

crime_station_t = []

for each_file in file_list:
    crime_raw_data = pd.read_csv(each_file, usecols=[0,1,2,3], encoding='euc-kr')

    crime_station = crime_raw_data.pivot_table(
        crime_raw_data, index=["구분"], columns=["죄종", "발생검거"], aggfunc=[np.sum]

    crime_station.columns = ['강간검거', '강간', '강도검거', '강도', '살인검거',
                             '살인', '절도검거', '절도', '폭력검거', '폭력']

    crime_station['year'] = each_file.split('/')[1][:-1][:-4]

    crime_station.reset_index(inplace=True)
    crime_station_t.append(crime_station)

```

## 15 데이터 합치기

```
crime_total = pd.concat(crime_station_t)
crime_total.head()
```

	구분	강간검거	강간	강도검거	강도	살인검거	살인	절도검거	절도	폭력검거	폭력	year
0	강남	70.0	78.0	72.0	146.0	4.0	3.0	523.0	2653.0	4345.0	4892.0	2000
1	강동	58.0	62.0	40.0	49.0	11.0	12.0	671.0	1838.0	4333.0	4716.0	2000
2	강북	48.0	47.0	48.0	47.0	8.0	9.0	458.0	859.0	3446.0	3624.0	2000
3	강서	64.0	75.0	55.0	75.0	7.0	6.0	509.0	1559.0	3988.0	4361.0	2000
4	관악	44.0	59.0	36.0	56.0	7.0	8.0	349.0	1764.0	2658.0	2921.0	2000

## 16 pivot을 이용하면???

```
1 crime_total_murder = crime_total.pivot(index='year', columns='구분', values='살인')
2 crime_total_murder
```

## 17 이렇게 마법처럼 정리된다

구분	강남	강동	강북	강서	관악	광진	구로	금천	남 대 문	노원	도봉	동대 문	동작	마포	방 배	서대 문	서 부	서초	성동	성 북	송파	수강
year																						
2000	3.0	12.0	9.0	6.0	8.0	2.0	10.0	10.0	3.0	3.0	3.0	5.0	13.0	3.0	1.0	3.0	6.0	2.0	10.0	3.0	2.0	3.0
2001	13.0	9.0	6.0	16.0	3.0	14.0	4.0	13.0	6.0	8.0	3.0	11.0	5.0	3.0	6.0	2.0	8.0	3.0	2.0	2.0	5.0	4.0
2002	3.0	6.0	7.0	8.0	8.0	12.0	4.0	16.0	2.0	7.0	2.0	8.0	10.0	4.0	4.0	1.0	5.0	6.0	4.0	6.0	6.0	7.0
2003	7.0	6.0	9.0	5.0	4.0	6.0	6.0	15.0	2.0	3.0	2.0	8.0	9.0	7.0	2.0	3.0	3.0	6.0	2.0	4.0	6.0	5.0
2004	8.0	7.0	15.0	8.0	0.0	10.0	13.0	16.0	1.0	5.0	4.0	12.0	18.0	24.0	2.0	5.0	7.0	2.0	5.0	6.0	6.0	6.0
2005	8.0	5.0	10.0	10.0	5.0	16.0	10.0	14.0	1.0	10.0	6.0	3.0	11.0	4.0	3.0	5.0	6.0	7.0	6.0	5.0	10.0	5.0
2006	5.0	4.0	5.0	16.0	5.0	7.0	16.0	5.0	2.0	6.0	10.0	10.0	7.0	4.0	1.0	7.0	3.0	2.0	5.0	5.0	8.0	4.0
2007	8.0	7.0	12.0	4.0	13.0	8.0	19.0	6.0	0.0	11.0	4.0	13.0	3.0	5.0	3.0	4.0	5.0	4.0	5.0	4.0	8.0	4.0
2008	10.0	8.0	7.0	18.0	10.0	3.0	22.0	3.0	1.0	3.0	9.0	11.0	9.0	8.0	0.0	8.0	3.0	1.0	1.0	5.0	10.0	3.0

## 18 한글 폰트 설치

```
import matplotlib as mpl
import matplotlib.pyplot as plt

%config InlineBackend.figure_format = 'retina'

!apt -qq -y install fonts-nanum

import matplotlib.font_manager as fm
fontpath = '/usr/share/fonts/truetype/nanum/NanumBarunGothic.ttf'
font = fm.FontProperties(fname=fontpath, size=9)

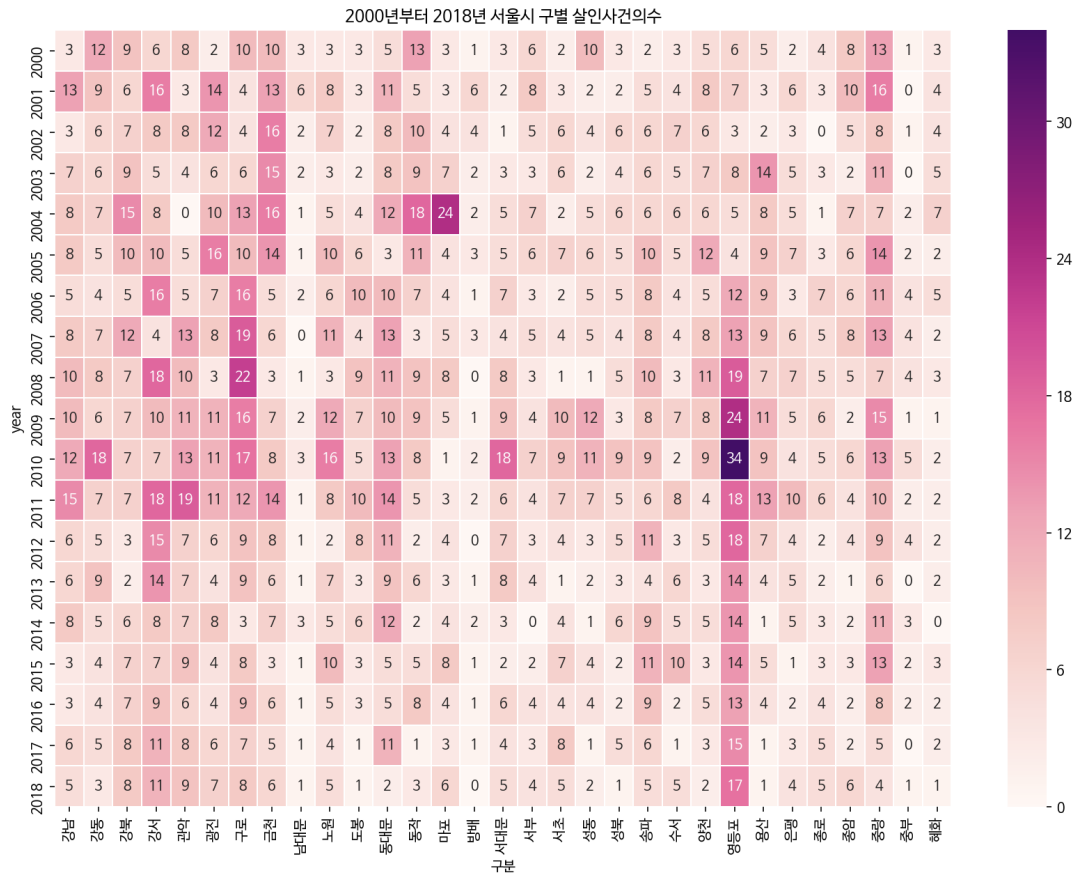
mpl.font_manager._rebuild()
plt.rc('font', family='NanumBarunGothic')
```

## 19 heatmap으로 그려보자

```
import seaborn as sns

plt.figure(figsize = (13,10))
sns.heatmap(crime_total_murder,
            annot=True, linewidths=.5, cmap='RdPu')
plt.title('2000년부터 2018년 서울시 구별 살인사건의수')
plt.autoscale()
plt.tight_layout(pad=5)
plt.show()
```

## 20 19년간 서울시 구별 살인사건의 변화

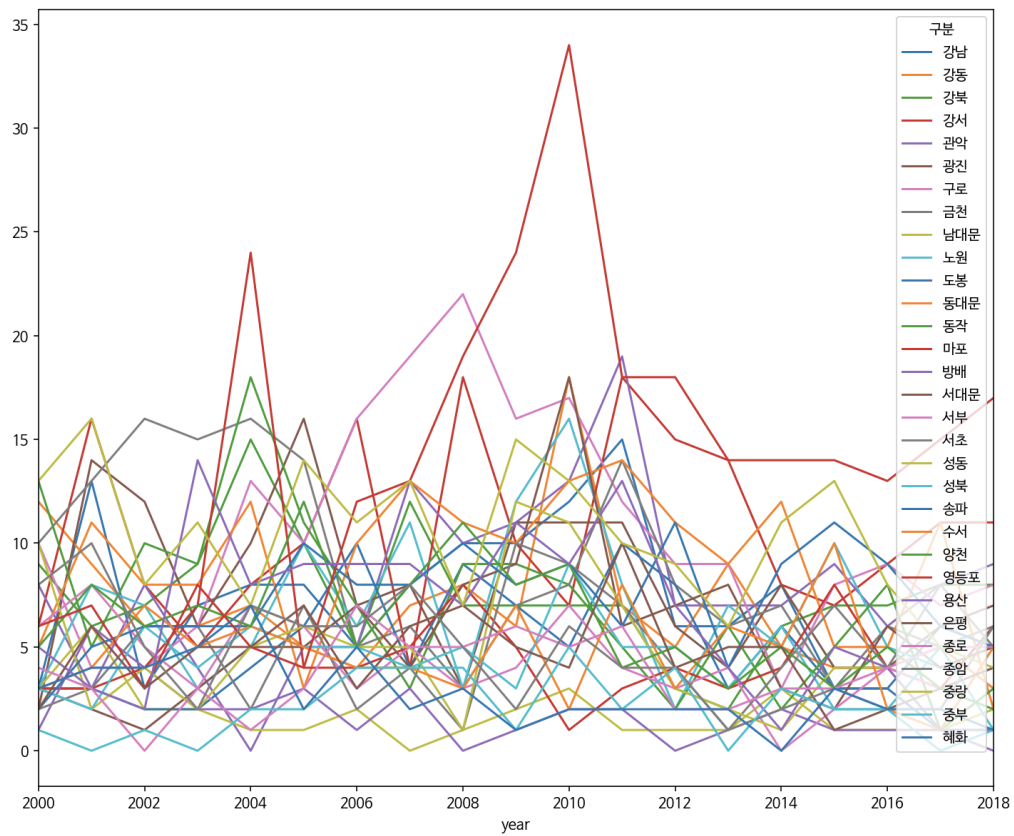


## 21 살인사건 데이터를 그냥 그려보자

```
crime_total_murder.plot(figsize=(12,10))  
plt.show()
```



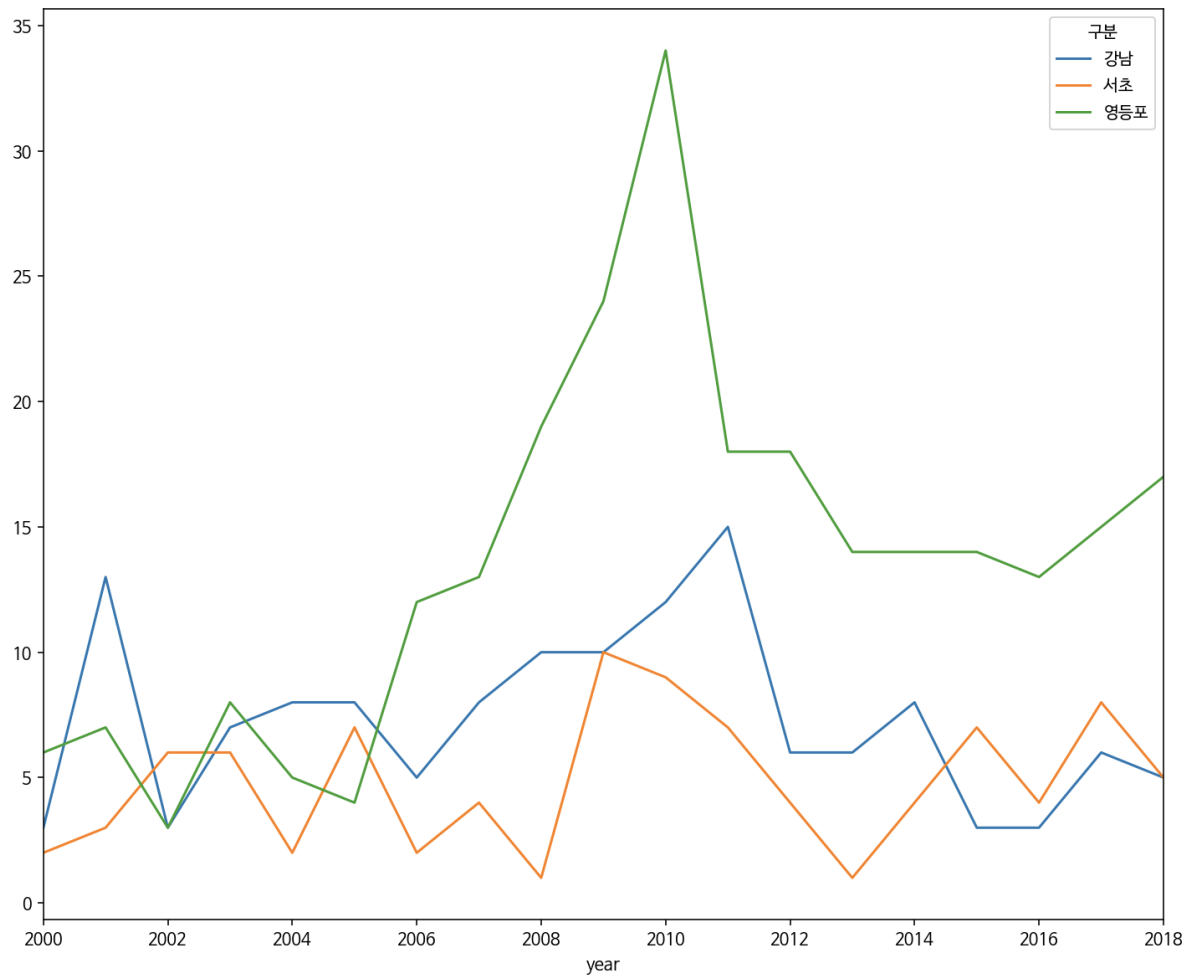
## 22 흠 어떤 의미인지 모르겠다



## 23 강남 서초 영등포의 살인 사건의 변화

```
crime_total_murder[['강남', '서초', '영등포']].plot(figsize=(12,10))  
plt.show()
```

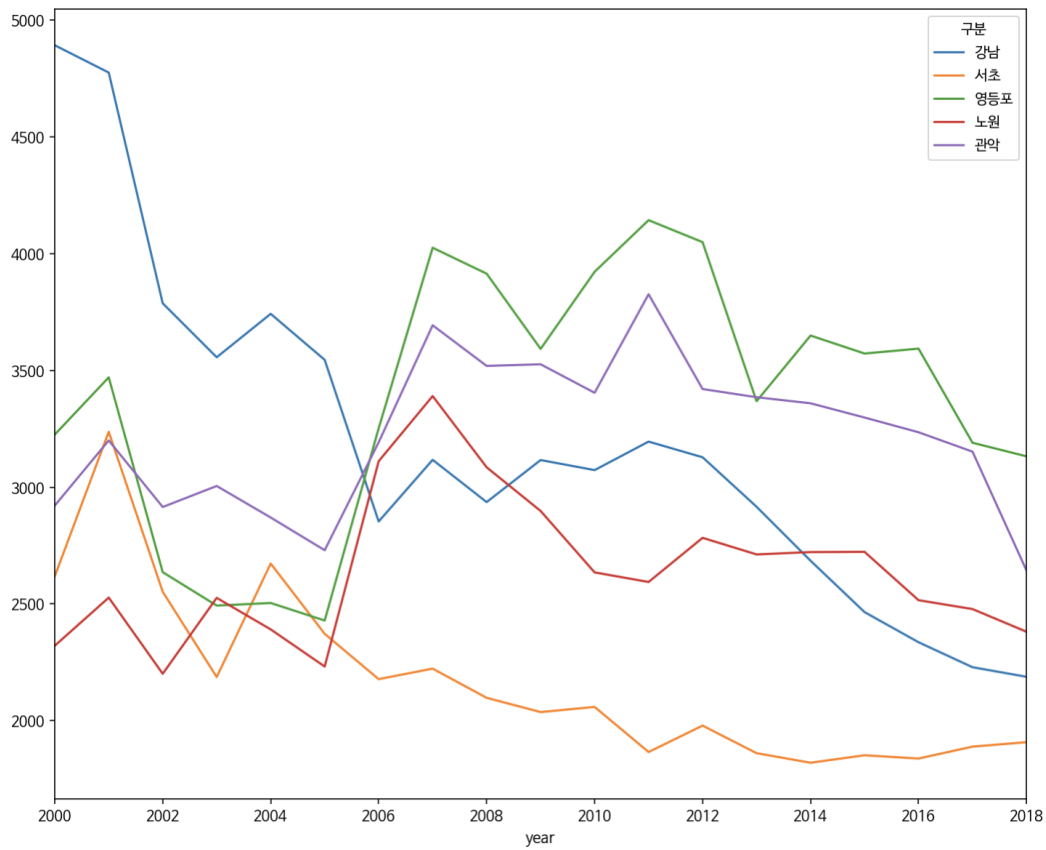
## 24 강남 서초 영등포의 살인 사건의 변화



## 25 폭력사건만 몇몇 구에 대해 관찰해보기

```
crime_total_violence = crime_total.pivot('year', '구분', '폭력')
crime_total_violence[['강남', '서초', '영등포', '노원', '관악']].plot(figsize=(12,10))
plt.show()
```

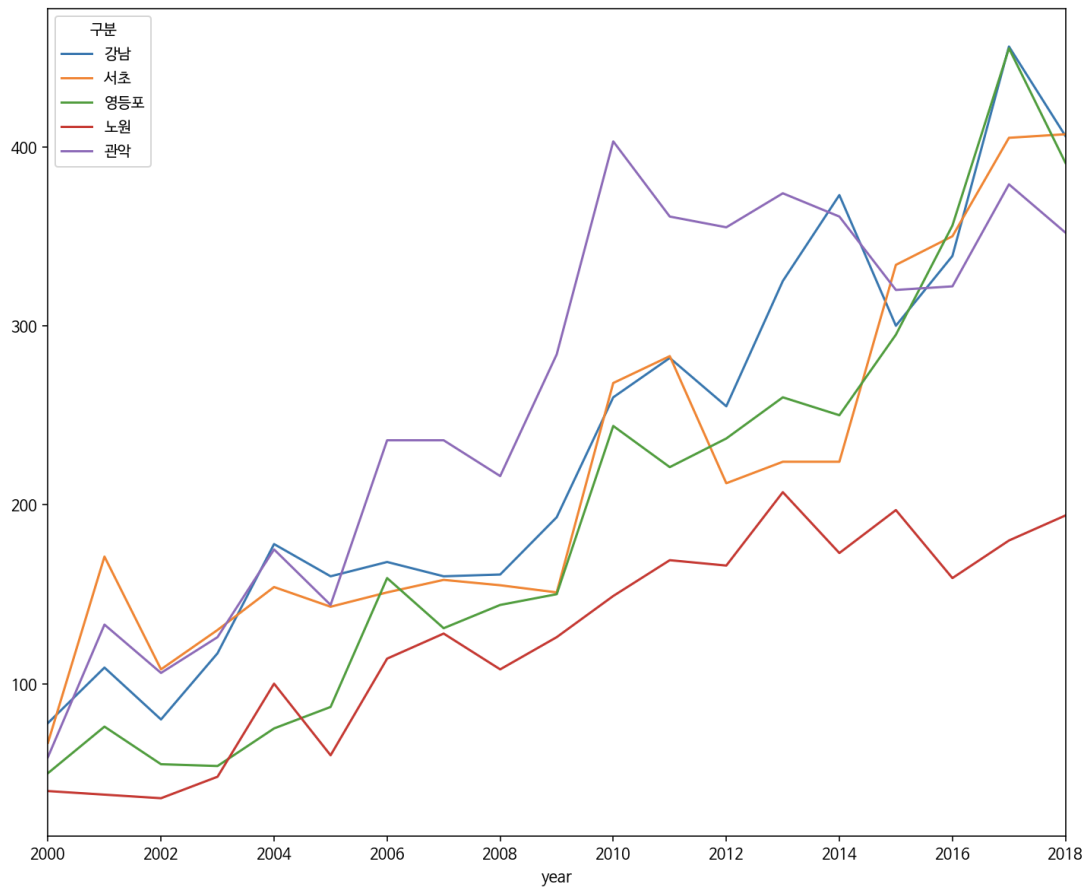
## 26 폭력사건만 몇몇 구에 대해 관찰해보기



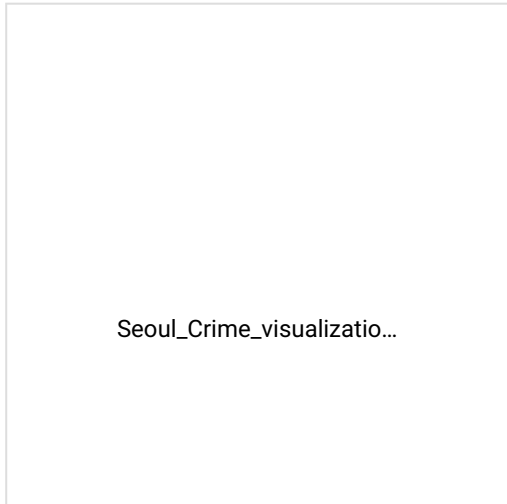
## 27 성범죄에 대해서 관찰해보기

```
crime_total_sexual = crime_total.pivot('year', '구분', '강간')
crime_total_sexual[['강남', '서초', '영등포', '노원', '관악']].plot(figsize=(12,10))
plt.show()
```

## 28 성범죄에 대해 연도별 상황



## 29 첨부파일



---

<sup>1</sup> [https://pinkwink.atlassian.net/wiki/spaces/AMR/pages/2254733689?preview=%2F2254733689%2F2254733789%2FSeoul\\_Crime\\_visualization\\_for\\_time\\_series\\_data.ipynb](https://pinkwink.atlassian.net/wiki/spaces/AMR/pages/2254733689?preview=%2F2254733689%2F2254733789%2FSeoul_Crime_visualization_for_time_series_data.ipynb)