따릉이 수요 예측 경진대회

Train 데이터EDA

```
In [8]: df = pd.read_csv("C:/Users/kjh1/Documents/드나/train.csv")
        df.head()
```

Out[8]:

	일시	광진구	동대문구	성동구	중랑구
0	20180101	0.592	0.368	0.580	0.162
1	20180102	0.840	0.614	1.034	0.260
2	20180103	0.828	0.576	0.952	0.288
3	20180104	0.792	0.542	0.914	0.292
4	20180105	0.818	0.602	0.994	0.308

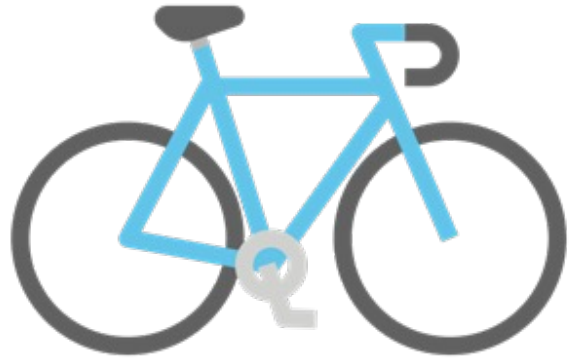
```
In [5]: print('데이터의 구조는:',df.shape)
        print('데이터의 타입은:',df.dtypes)
        print('데이터의 칼럼은:',df.columns)
```

```
데이터의 구조는: (1461, 5)
데이터의 타입은: 일시        int64
광진구      float64
동대문구     float64
성동구      float64
중랑구      float64
dtype: object
데이터의 칼럼은: Index(['일시', '광진구', '동대문구', '성동구', '중랑구'], dtype='object')
```

```
n [24]: df.isnull().sum()
```

```
Out[24]: 일시        0
광진구       0
동대문구      0
성동구       0
중랑구       0
year      0
month     0
day       0
dtype: int64
```

결측치 없음



파생변수 생성

```
df['일시'] = df['일시'].astype(str)
```

```
df['일시'] = pd.to_datetime(df['일시'])
```

```
df['year'] = df.일시.dt.year
df['month'] = df.일시.dt.month
df['day'] = df.일시.dt.day
df['weekday'] = df.일시.dt.weekday
```

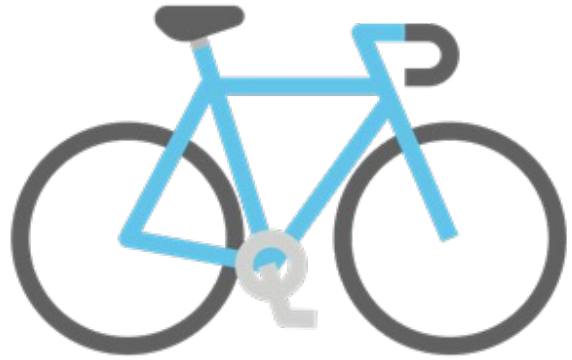
```
df.head()
```

	일시	광진구	동대문구	성동구	중랑구	year	month	day	weekday
0	2018-01-01	0.592	0.368	0.580	0.162	2018	1	1	0
1	2018-01-02	0.840	0.614	1.034	0.260	2018	1	2	1
2	2018-01-03	0.828	0.576	0.952	0.288	2018	1	3	2
3	2018-01-04	0.792	0.542	0.914	0.292	2018	1	4	3
4	2018-01-05	0.818	0.602	0.994	0.308	2018	1	5	4

Pd.to_datetime를 이용해

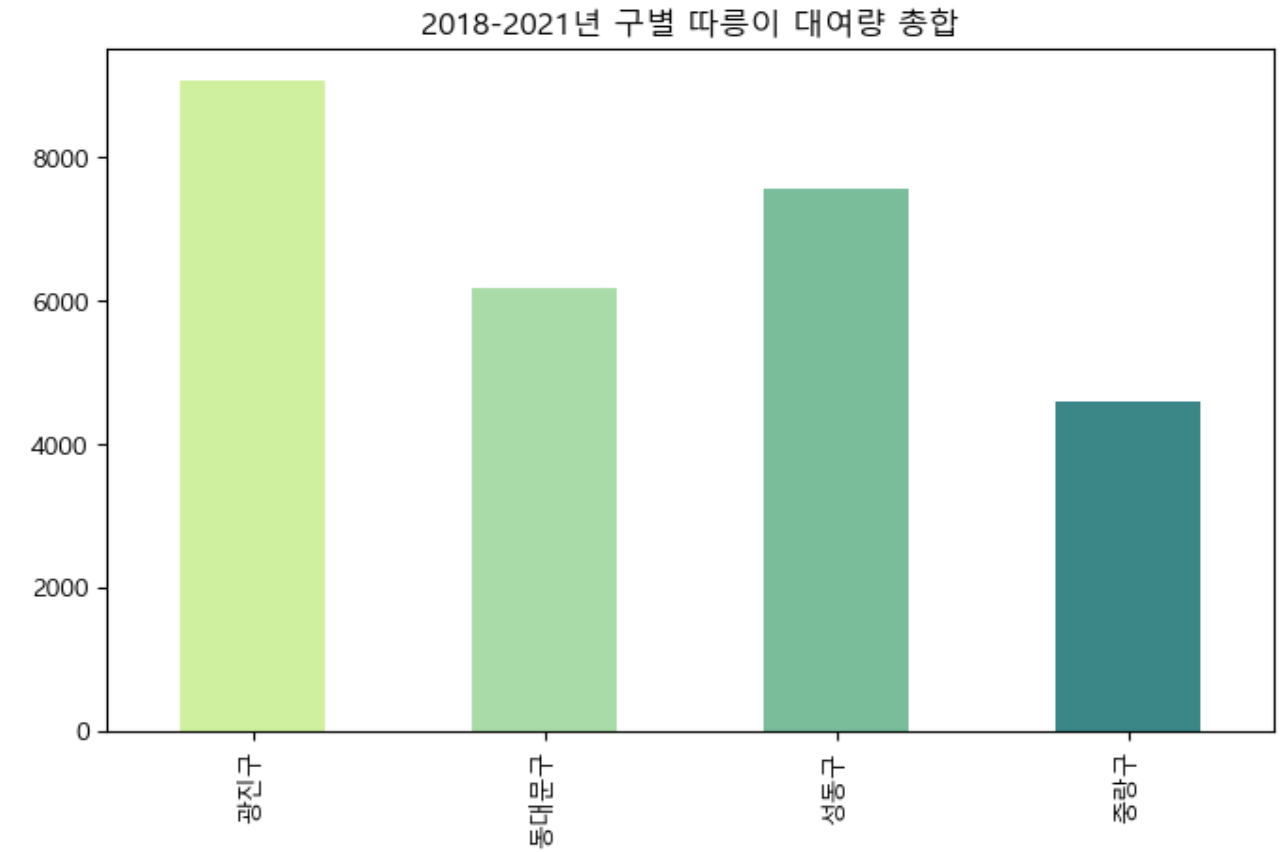
- 연/월/일/요일 분리

- Weekday: 0(월요일)~6(일요일)

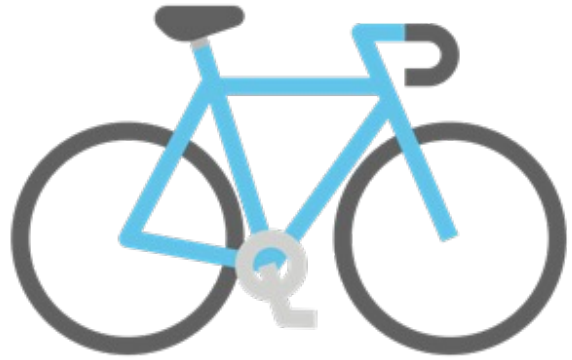


자치구별 해당기간 총 대여량

```
plt.figure(figsize=(8,5))
df.sum()[1:].plot(kind='bar',color=['#cff09e','#a8dba8','#79bd9a','#3b8686'])
plt.title('2018-2021년 구별 따릉이 대여량 총합')
plt.show()
# 광진구','동대문구','성동구','중랑구
# 광진구에서 가장 많은 대여량이 있음
```



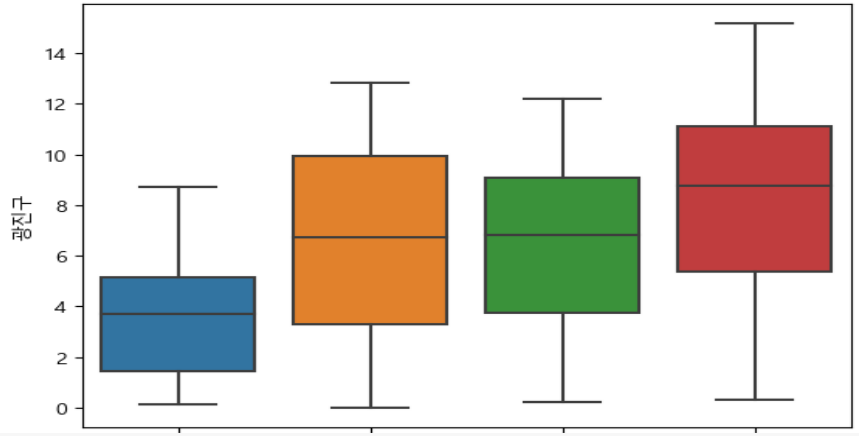
광진구가 가장 많고 중랑구가 가장 적음



자치구별 연간 수요그래프

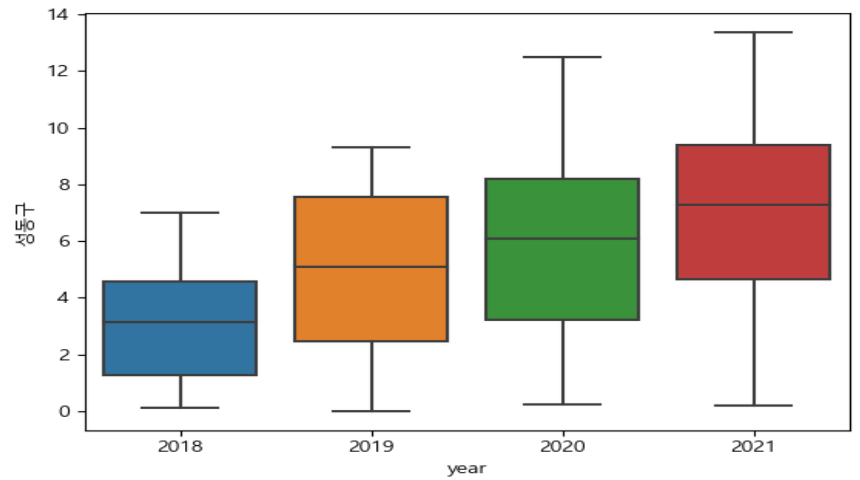
```
sns.boxplot(data = df, x = 'year', y = '광진구')
```

```
<AxesSubplot:xlabel='year', ylabel='광진구'>
```



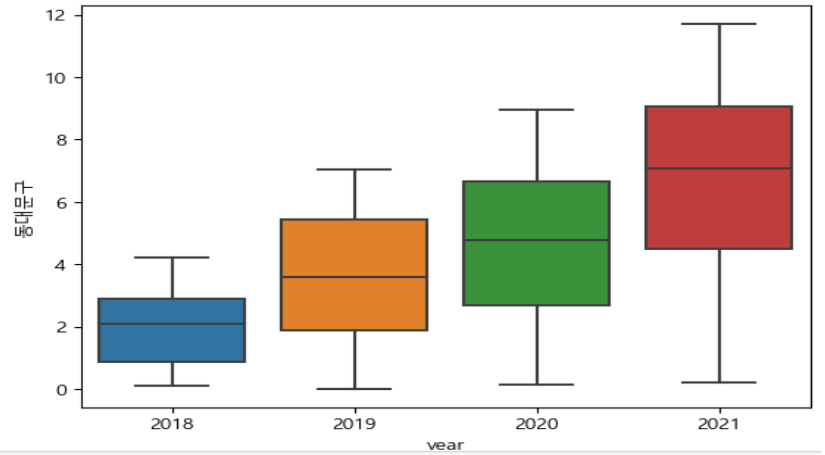

```
sns.boxplot(data = df, x = 'year', y = '성동구')
```

```
<AxesSubplot:xlabel='year', ylabel='성동구'>
```



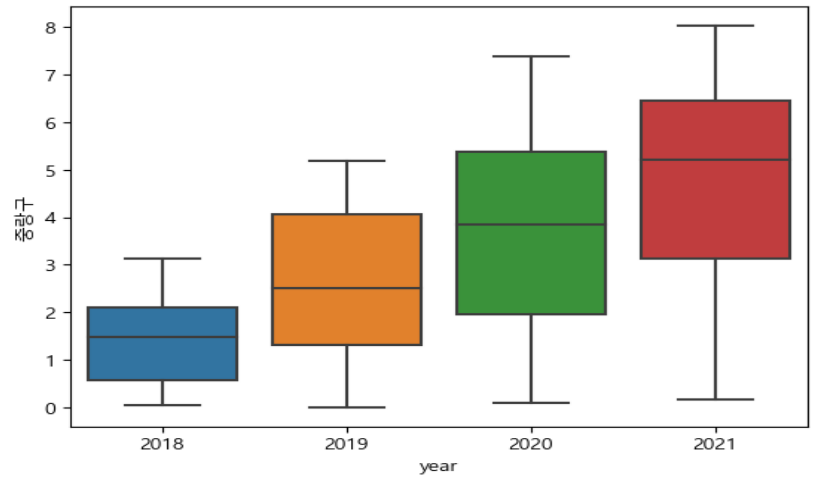

```
sns.boxplot(data = df, x = 'year', y = '동대문구')
```

```
<AxesSubplot:xlabel='year', ylabel='동대문구'>
```

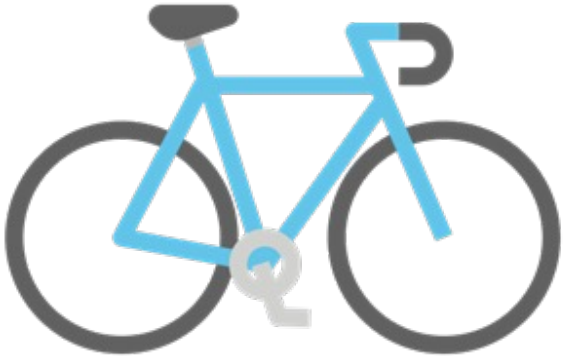


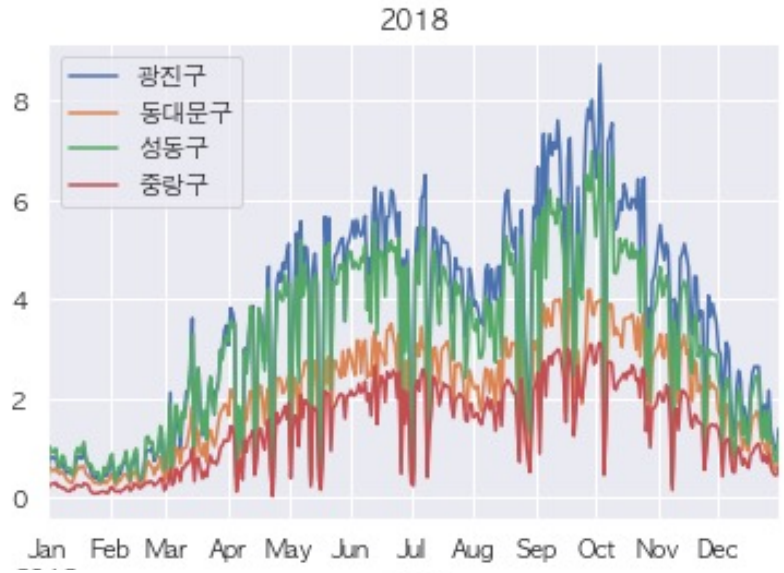

```
sns.boxplot(data = df, x = 'year', y = '중랑구')
```

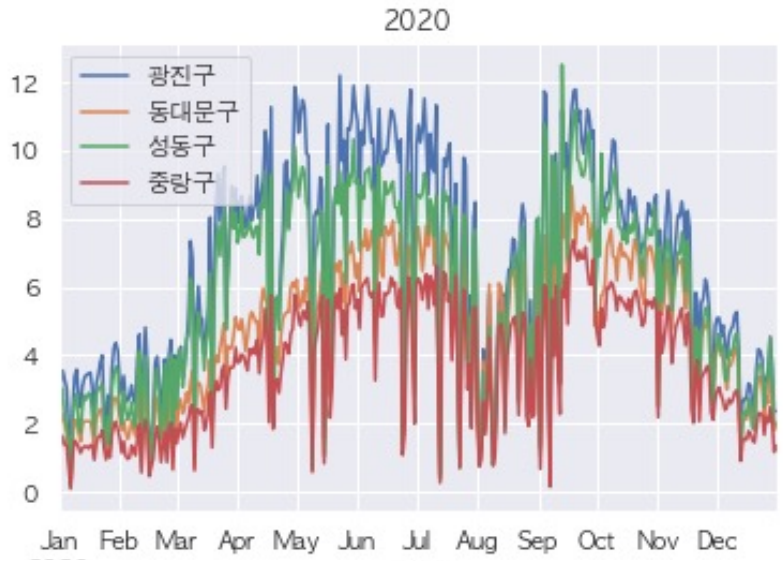
```
<AxesSubplot:xlabel='year', ylabel='중랑구'>
```

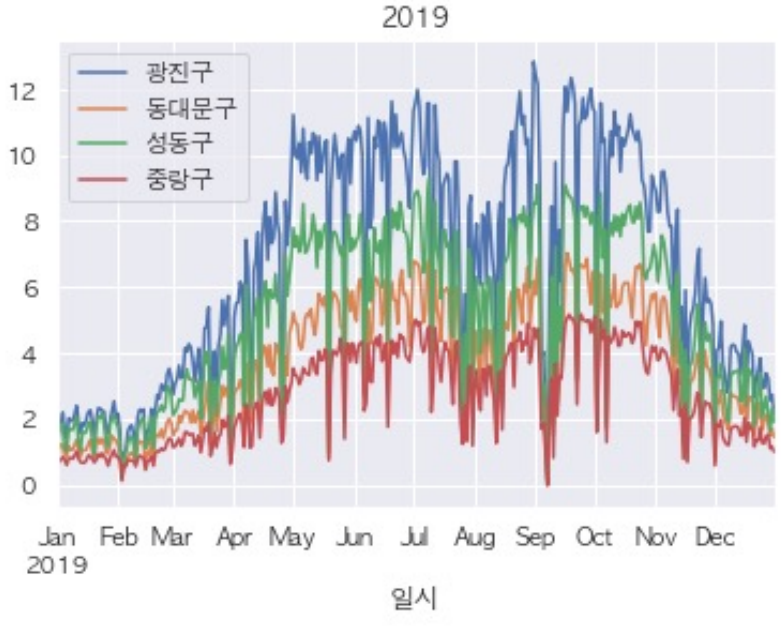


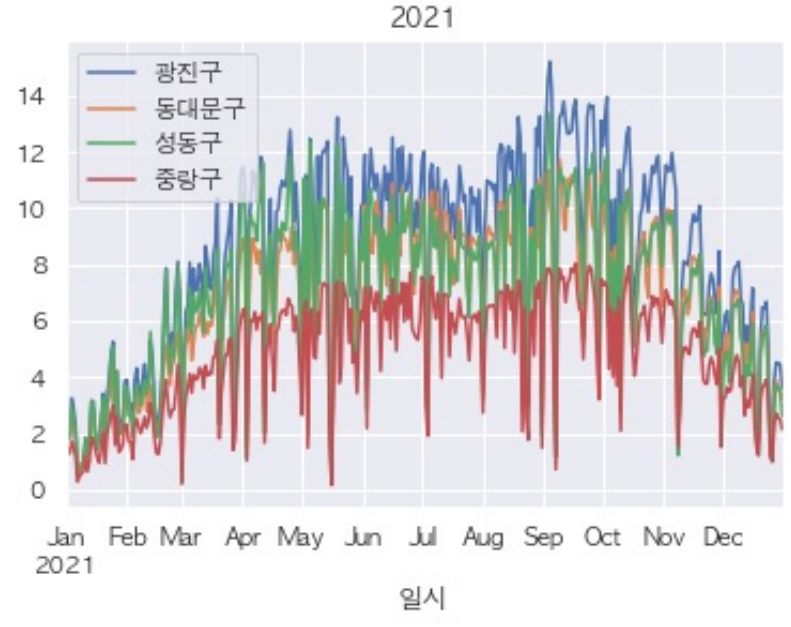
모든 구가 해가 지날수록 이용량이 증가함







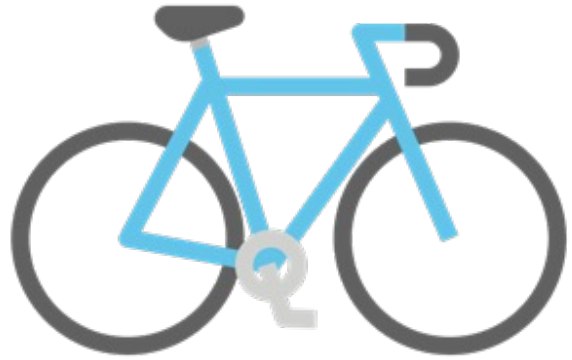




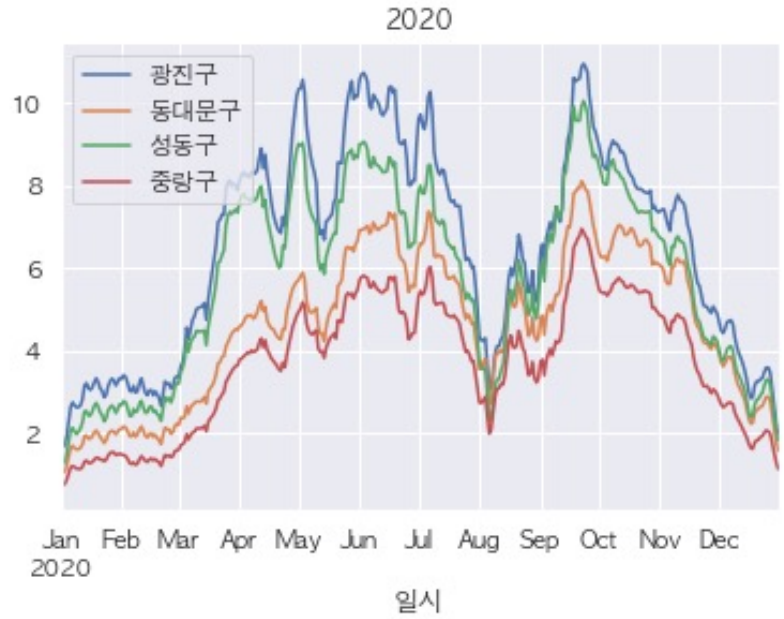
자치구별 월간 수요 그래프

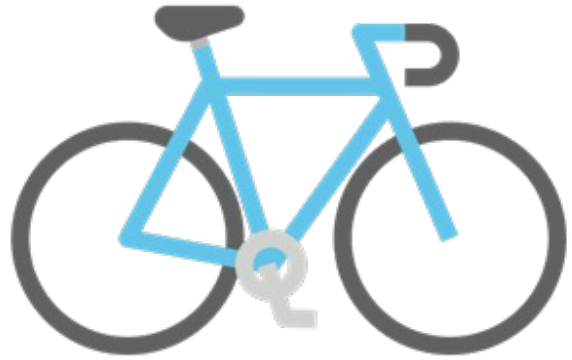
그래프의 노이즈가 너무 심해 평활기법을 사용해 시각적으로 보기 쉽게 변형

```python
# smoothed_data yearly plot
def smooth(y, box_pts):
    box = np.ones(box_pts)/box_pts
    y_smooth = np.convolve(y, box, mode='same')
    return y_smooth
```



자치구별 월간 수요 그래프





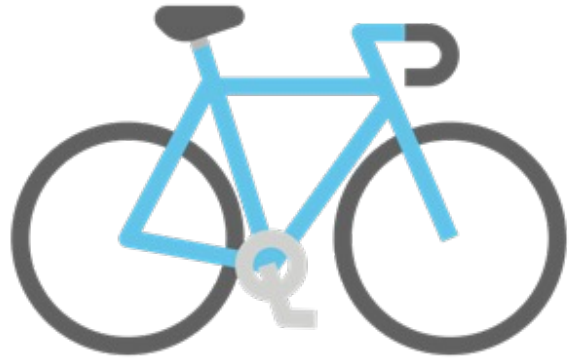

2018_동대문구
2019_동대문구
2020_동대문구
2021_동대문구


2018_성동구
2019_성동구
2020_성동구
2021_성동구


2018_중랑구
2019_중랑구
2020_중랑구
2021_중랑구


2018_광진구
2019_광진구
2020_광진구
2021_광진구

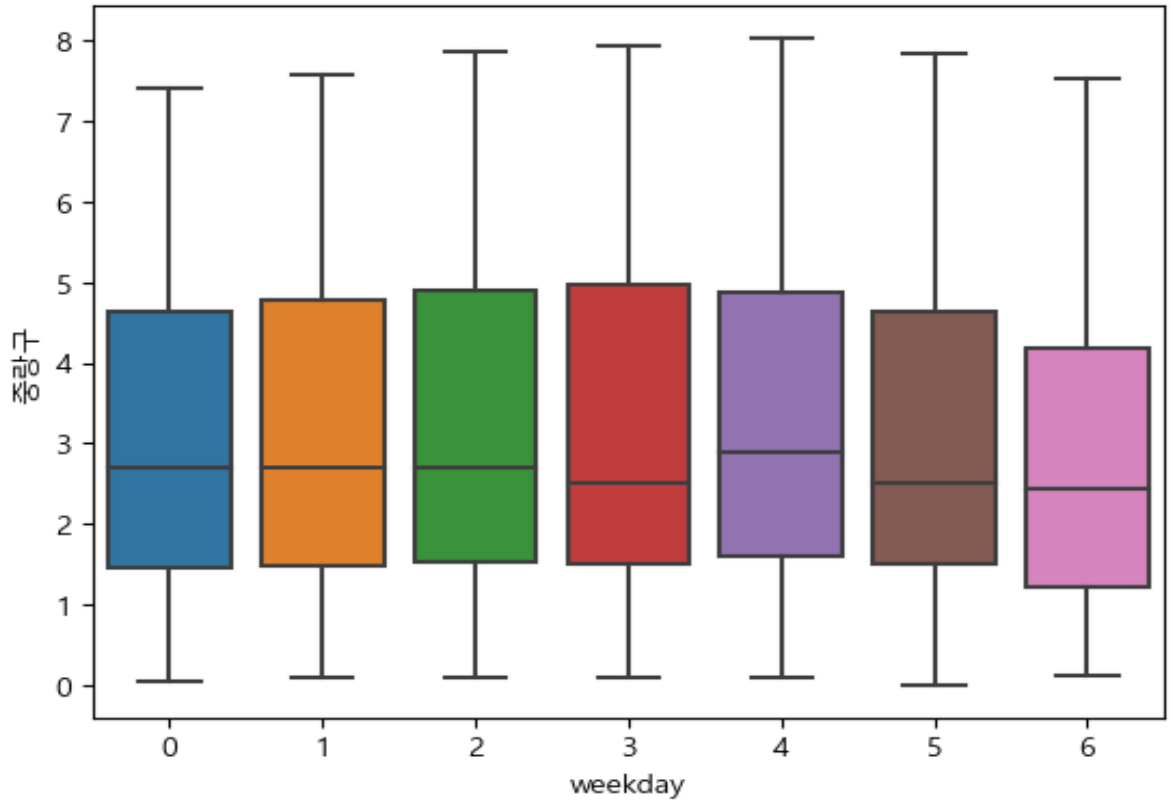
같은 구 별로 묶은 그래프



자치구별 일간 수요 그래프

```
sns.boxplot(data = df, x = 'weekday', y = '중랑구')
```

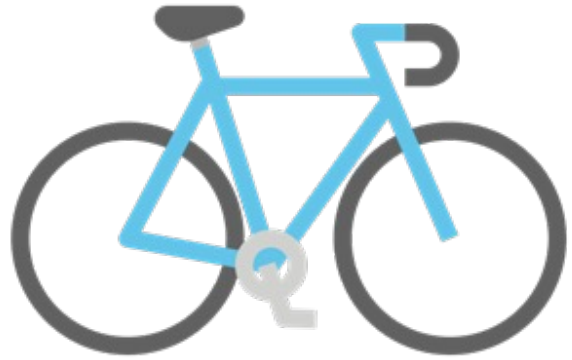
```
<AxesSubplot:xlabel='weekday', ylabel='중랑구'>
```



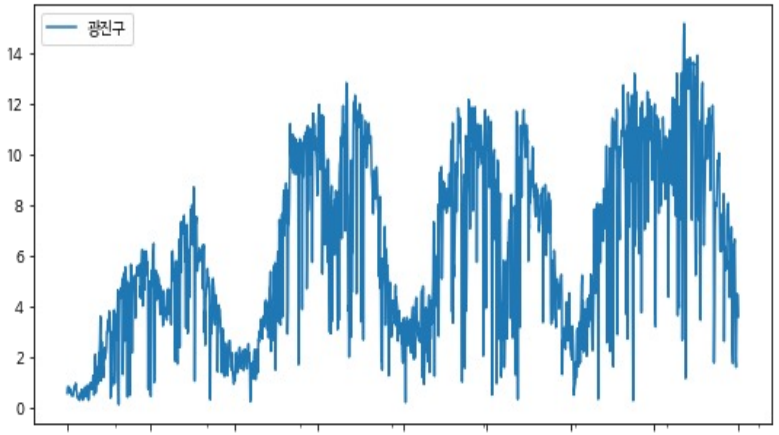
해가 지날수록 점차 수요량이 증가됨

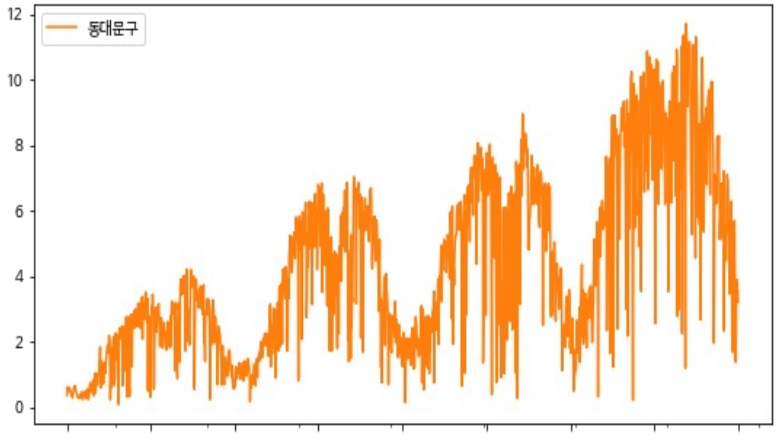
보편적으로 활동하기 좋은 봄, 가을에는 수요가 많고 여름과 겨울엔 수요가 낮아짐

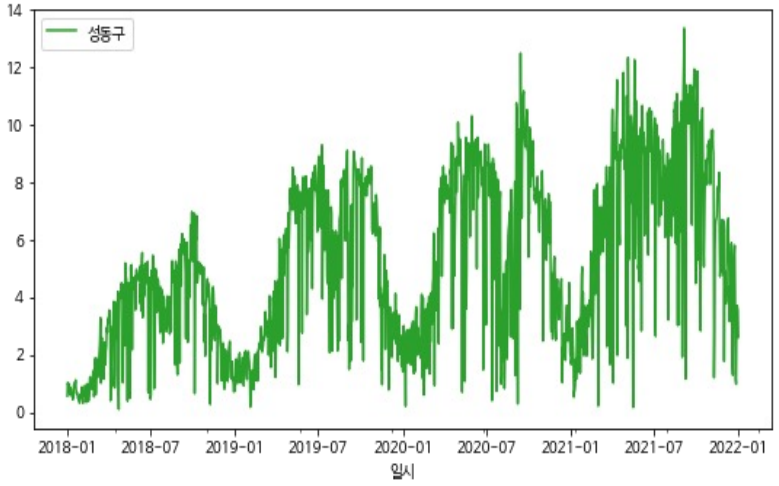
요일에 상관없이 수요량이 비슷함

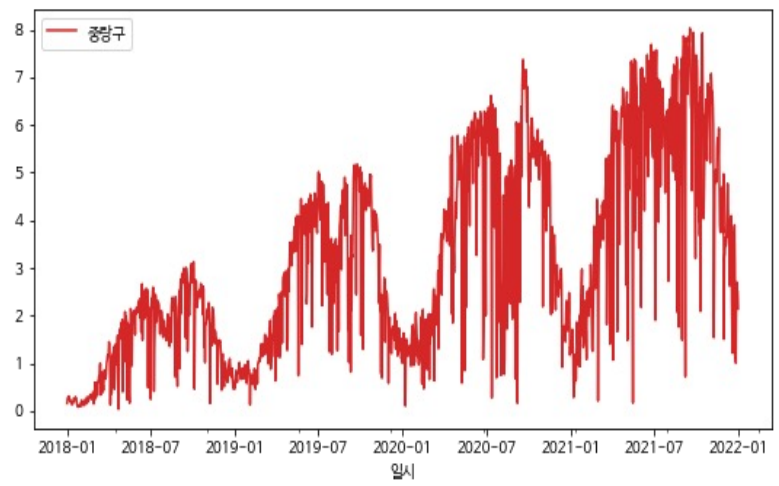


자치구별 수요 시계열 그래프

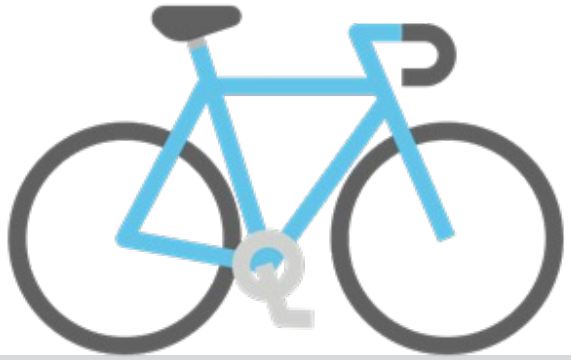








4개의 자치구 모두 비슷한 형태의 그래프를 가짐



외부 데이터탐색(강수량)

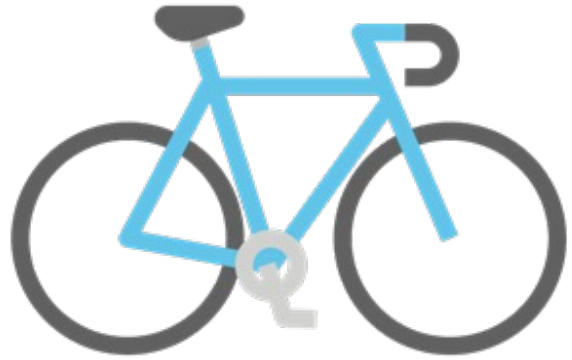
```
df3 = pd.read_csv('/content/2018 - 2021 강수량자료.csv',encoding='euc-kr')
df3
```

	강수분석	Unnamed: 1	Unnamed: 2
0	[검색조건]	NaN	NaN
1	자료구분 : 일	NaN	NaN
2	자료형태 : 기본	NaN	NaN
3	지역/지점 : 서울	NaN	NaN
4	기간 : 20180101~20211231	NaN	NaN
...	...	...	...
1463	2021-12-27	108	0
1464	2021-12-28	108	NaN
1465	2021-12-29	108	0.2
1466	2021-12-30	108	0
1467	2021-12-31	108	NaN

1468 rows × 3 columns

서울시 18년도~21년도 까지의 일간 강수량 데이터

날짜, 지역, 강수량으로 구성되어 있음



외부 데이터탐색(강수량)

```
df3.drop(df3.index[0:7], inplace = True)
df3 = df3.reset_index(drop=True); df3
```

	강수분석	Unnamed: 1	Unnamed: 2
0	2018-01-01	108	NaN
1	2018-01-02	108	NaN
2	2018-01-03	108	NaN
3	2018-01-04	108	NaN
4	2018-01-05	108	NaN
...	...	...	...
1456	2021-12-27	108	0
1457	2021-12-28	108	NaN
1458	2021-12-29	108	0.2
1459	2021-12-30	108	0
1460	2021-12-31	108	NaN

1461 rows × 3 columns

```
df3 = df3.rename(columns = {'Unnamed: 1': ' 지역','Unnamed: 2':'강수량(mm)','강수분석':'일시'})
df3
```

	일시	지역	강수량(mm)
0	2018-01-01	108	NaN
1	2018-01-02	108	NaN
2	2018-01-03	108	NaN
3	2018-01-04	108	NaN
4	2018-01-05	108	NaN
...	...	...	...
1456	2021-12-27	108	0
1457	2021-12-28	108	NaN
1458	2021-12-29	108	0.2
1459	2021-12-30	108	0
1460	2021-12-31	108	NaN

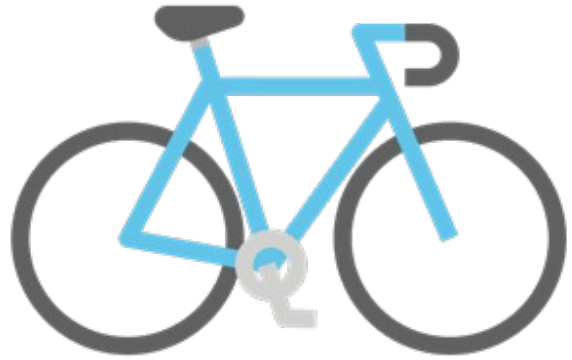
```
df3 = df3[['일시','강수량(mm)']]
df3['강수량(mm)'].fillna(0,inplace=True)
df3.columns=['일시','강수량(mm)']
df3
```

	일시	강수량(mm)
0	2018-01-01	0
1	2018-01-02	0
2	2018-01-03	0
3	2018-01-04	0
4	2018-01-05	0
...	...	...
1456	2021-12-27	0
1457	2021-12-28	0
1458	2021-12-29	0.2
1459	2021-12-30	0
1460	2021-12-31	0

불필요한 데이터가 들어간 행 제거 및 변수 이름 재설정

지역의 모든 값은 108(서울)을 의미하므로 삭제

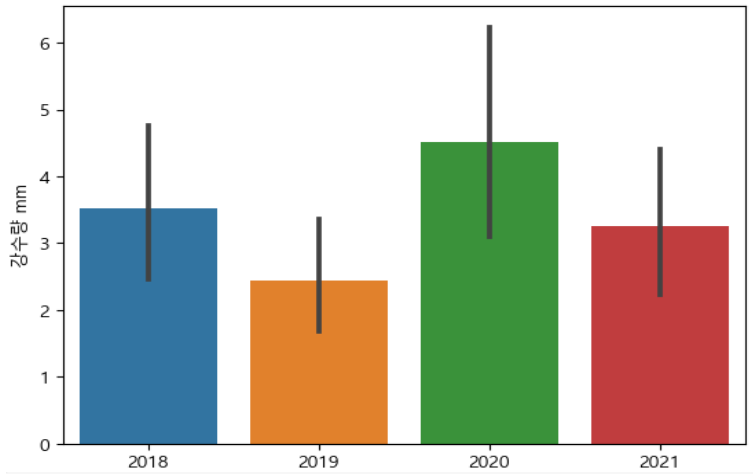
또한 강수량에 존재하는 결측값을 0으로 대체



외부 데이터탐색(강수량)

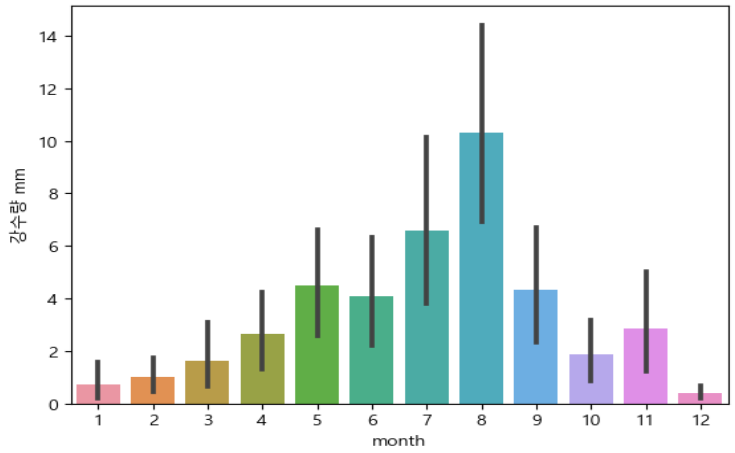
```
sns.barplot(data = df, x = 'year', y = '강수량 mm')
```

```
<AxesSubplot:xlabel='year', ylabel='강수량 mm'>
```



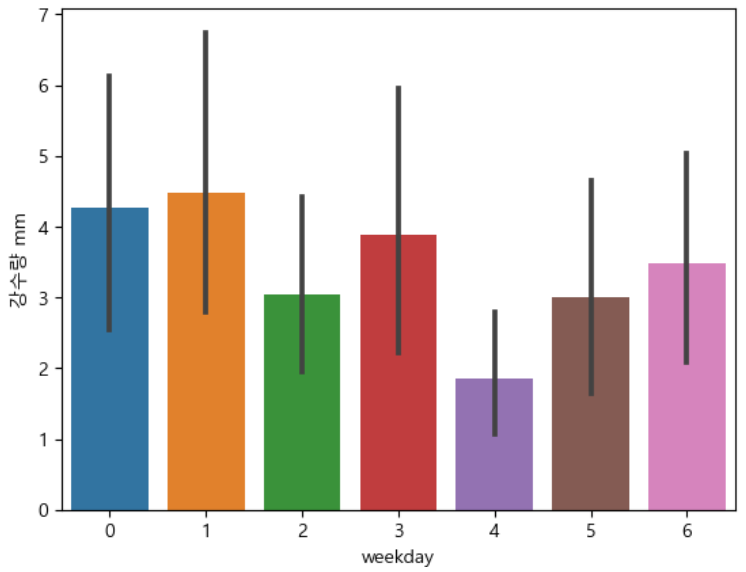

```
sns.barplot(data = df, x = 'month', y = '강수량 mm')
```

```
<AxesSubplot:xlabel='month', ylabel='강수량 mm'>
```



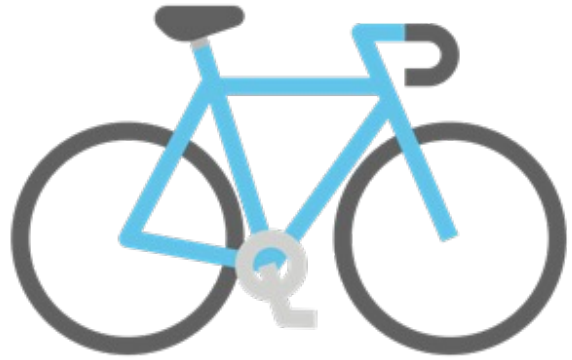

```
sns.barplot(data = df, x = 'weekday', y = '강수량 mm')
```

```
<AxesSubplot:xlabel='weekday', ylabel='강수량 mm'>
```



장마의 영향으로 7~8월에 강수량이 집중되있음

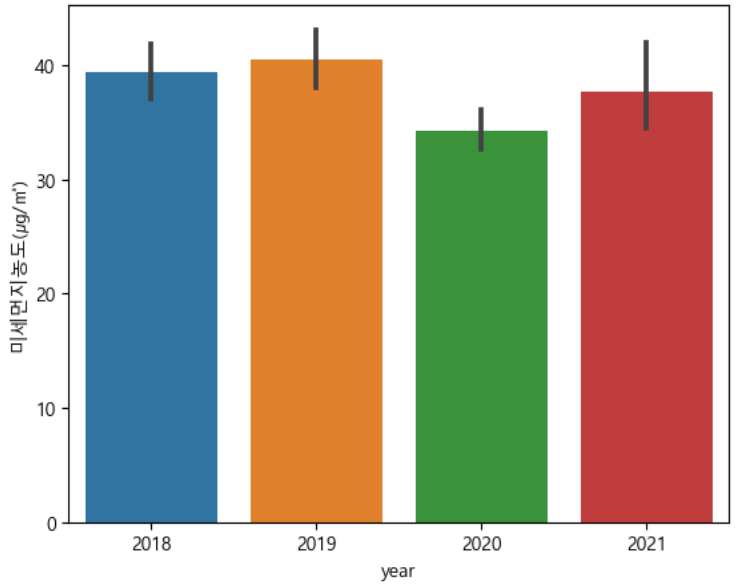
같은 방법으로 기온, 미세먼지에 대한 해당 작업 수행



외부 데이터탐색(미세먼지, 기온)

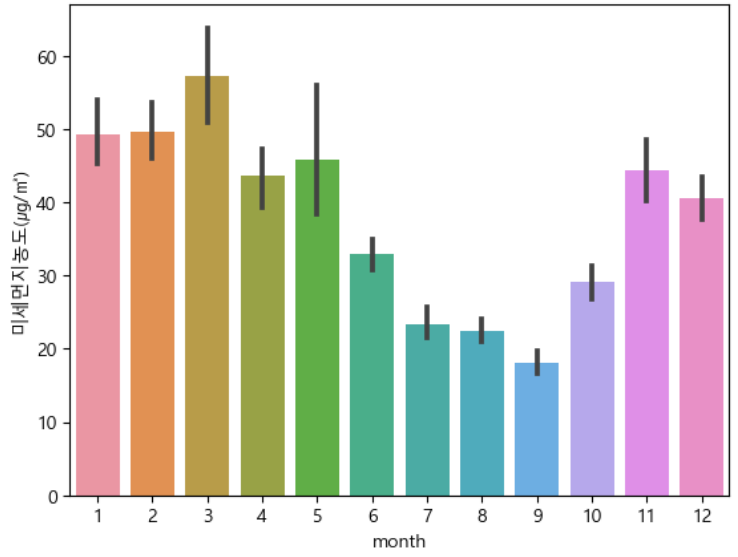
```
sns.barplot(data = df, x = 'year', y = '미세먼지농도(㎍/㎥)')
```

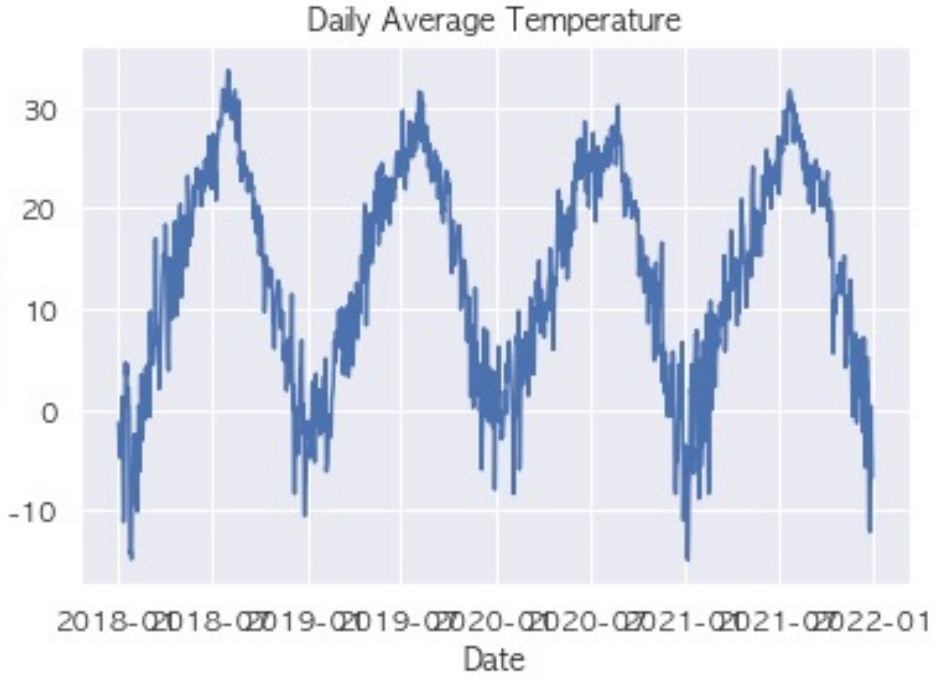
```
<AxesSubplot:xlabel='year', ylabel='미세먼지농도(㎍/㎥)'>
```




```
sns.barplot(data = df, x = 'month', y = '미세먼지농도(㎍/㎥)')
```

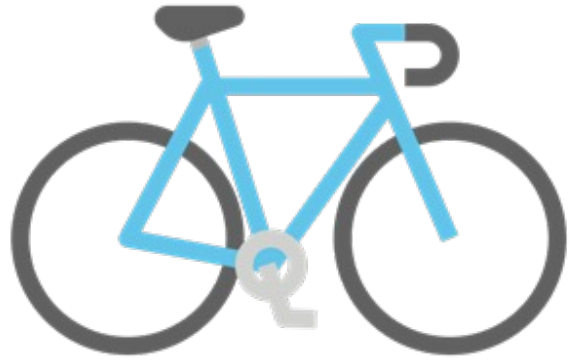
```
<AxesSubplot:xlabel='month', ylabel='미세먼지농도(㎍/㎥)'>
```





연간, 일간은 큰 특징이 보이진 않음
월간에선 6~10월에 낮아지는 데 장마, 태풍, 계절풍에 따른 영향이라고 판단됨

또한 기온은 예상한대로 계절성을 확실하게 가짐



외부 데이터탐색(미세먼지, 기온)

서울시 18~21년도 미세먼지, 초미세먼지 데이터

	측정일시	미세먼지농도(㎍/㎥)	초미세먼지농도(㎍/㎥)
1	2018-01-01	49.0	24.0
5	2018-01-01	42.0	22.0
10	2018-01-01	46.0	21.0
15	2018-01-01	48.0	23.0
24	2018-01-01	48.0	23.0
...	...	...	...
9101	2021-12-31	24.0	7.0
9105	2021-12-31	21.0	6.0
9110	2021-12-31	21.0	7.0
9115	2021-12-31	21.0	5.0
9124	2021-12-31	22.0	6.0

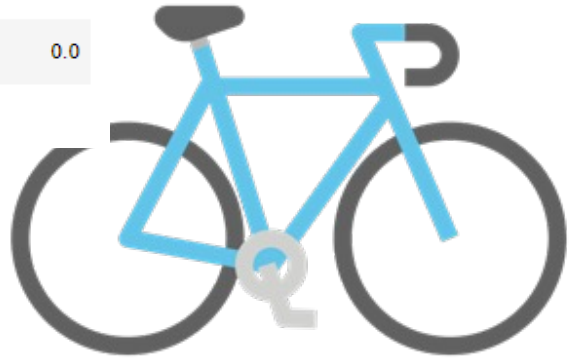
서울시 18~21년도평균 기온 데이터

	날짜	지점	평균기온(°C)	최저기온(°C)	최고기온(°C)
0	2018-01-01	108	-1.3	-5.1	3.8
1	2018-01-02	108	-1.8	-4.3	1.8
2	2018-01-03	108	-4.7	-7.1	-0.4
3	2018-01-04	108	-4.7	-8.7	-0.7
4	2018-01-05	108	-3.0	-5.6	1.6
...	...	...	...	...	...
1456	2021-12-27	108	-7.6	-12.9	-3.9
1457	2021-12-28	108	-4.1	-8.5	-0.9
1458	2021-12-29	108	0.4	-3.8	5.9
1459	2021-12-30	108	-3.9	-6.8	0.2
1460	2021-12-31	108	-6.7	-8.8	-3.9

최종데이터셋

	Unnamed: 0	일시	광진구	동대문구	성동구	중랑구	year	month	day	weekday	평균기온(℃)	미세먼지농도(㎍/㎥)	초미세먼지농도(㎍/㎥)	강수량 mm
0	0	2018-01-01	0.592	0.368	0.580	0.162	2018	1	1	0	-1.3	46.6	22.6	0.0
1	1	2018-01-02	0.840	0.614	1.034	0.260	2018	1	2	1	-1.8	43.8	23.2	0.0
2	2	2018-01-03	0.828	0.576	0.952	0.288	2018	1	3	2	-4.7	37.2	20.0	0.0
3	3	2018-01-04	0.792	0.542	0.914	0.292	2018	1	4	3	-4.7	49.2	26.0	0.0
4	4	2018-01-05	0.818	0.602	0.994	0.308	2018	1	5	4	-3.0	64.4	39.4	0.0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1456	1456	2021-12-27	3.830	3.416	2.908	2.350	2021	12	27	0	-7.6	29.6	16.8	0.0
1457	1457	2021-12-28	4.510	3.890	3.714	2.700	2021	12	28	1	-4.1	49.4	35.4	0.0
1458	1458	2021-12-29	4.490	3.524	3.660	2.524	2021	12	29	2	0.4	62.0	44.8	0.2
1459	1459	2021-12-30	4.444	3.574	3.530	2.506	2021	12	30	3	-3.9	28.2	14.6	0.0
1460	1460	2021-12-31	3.616	3.210	2.620	2.146	2021	12	31	4	-6.7	21.8	6.2	0.0

1461 rows × 14 columns

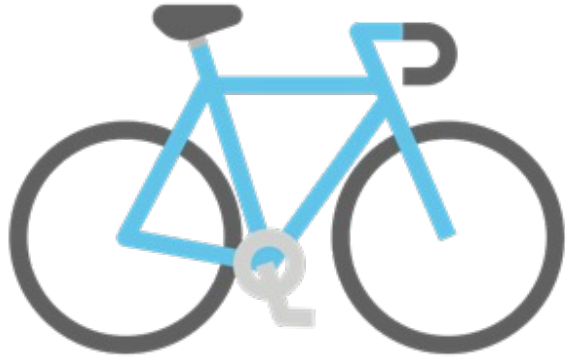


다음에 할 일

시계열 분해 - 가법 분해 예정 (additive)

ACF 및 PACF 결과 확인하기

차분 및 로그변환 예정



END.