Pandas 자료 병합

특정 컬럼명 변경 - rename

df.rename(columns = {'Column Name': 'New Name'})

```
df1 = data[data['구분'] == '연간(톤)']
df1 = df1.set_index('항목')
del df1['구분']
df1 = df1.T
df1 = df1.reset_index()

df1 = df1.rename(columns = {'index': '지역'})
df1
```

항목	지역	매립처리량	소각처리량	재활용처리량	음식물류발생량
0	중구	3034	13729	9082	5154
1	서구	2323	10446	21301	8779
2	동구	1653	8297	17968	6399
3	영도구	1219	8651	24616	9522
4	부산진구	17393	538	88453	27119
5	동래구	9195	17	56526	20128
6	남구	8042	158	57921	15548
7	북구	10498	97	63879	24201
8	해운대구	5251	31924	64667	40043
9	사하구	3089	32671	70923	21183
10	금정구	7240	222	56153	21273
11	강서구	9742	4788	25977	10810
12	연제구	1831	14521	37115	14122
13	수영구	2205	11987	28135	13289
14	사상구	9937	216	57110	24161
15	기장군	2290	13687	15170	16648

```
df2=  pd.read_excel('부산구청.xlsx')
df2 = df2.rename(columns = {'항목': '지역'})
df2
```

	지역	위도	경도
0	중구	35.106609	129.030064
1	서구	35.097924	129.022010
2	동구	35.129344	129.043314
3	영도구	35.091212	129.065700
4	부산진구	35.163087	129.051213
5	동래구	35.204841	129.081425
6	남구	35.136578	129.082053
7	북구	35.198325	128.987705
8	해운대구	35.163106	129.161390
9	사하구	35.104451	128.972650
10	금정구	35.242974	129.089958
11	강서구	35.212231	128.978379
12	연제구	35.176504	129.077611
13	수영구	35.145615	129.110889
14	사상구	35.152640	128.988623
15	기장군	35.244600	129.220059

자료 병합 - concat

- **concat**
  - 기준 열(key column)을 사용하지 않고 단순히 데이터를 연결(concatenate)
  - Index를 기준으로 합침

항목	지역	매립처리량	소각처리량	재활용처리량	음식물류발생량
0	중구	3034	13729	9082	5154
1	서구	2323	10446	21301	8779
2	동구	1653	8297	17968	6399
3	영도구	1219	8651	24616	9522
4	부산진구	17393	538	88453	27119
5	동래구	9195	17	56526	20128
6	남구	8042	158	57921	15548
7	북구	10498	97	63879	24201
8	해운대구	5251	31924	64667	40043
9	사하구	3089	32671	70923	21183
10	금정구	7240	222	56153	21273
11	강서구	9742	4788	25977	10810
12	연제구	1831	14521	37115	14122
13	수영구	2205	11987	28135	13289
14	사상구	9937	216	57110	24161
15	기장군	2290	13687	15170	16648

df1

	지역	위도	경도
0	중구	35.106609	129.030064
1	서구	35.097924	129.022010
2	동구	35.129344	129.043314
3	영도구	35.091212	129.065700
4	부산진구	35.163087	129.051213
5	동래구	35.204841	129.081425
6	남구	35.136578	129.082053
7	북구	35.198325	128.987705
8	해운대구	35.163106	129.161390
9	사하구	35.104451	128.972650
10	금정구	35.242974	129.089958
11	강서구	35.212231	128.978379
12	연제구	35.176504	129.077611
13	수영구	35.145615	129.110889
14	사상구	35.152640	128.988623
15	기장군	35.244600	129.220059

df2

```
#단순 합치기
m1 = pd.concat([df1, df2])
m1
```

	경도	매립처리량	소각처리량	위도	음식물류발생량	재활용처리량	지역
0	NaN	3034.0	13729.0	NaN	5154.0	9082.0	중구
1	NaN	2323.0	10446.0	NaN	8779.0	21301.0	서구
2	NaN	1653.0	8297.0	NaN	6399.0	17968.0	동구

Index로 열 방향으로 합치기

```
m2 = pd.concat([df1, df2], axis = 1, join='inner')
m2
```

	지역	매립처리량	소각처리량	재활용처리량	음식물류발생량	지역	위도	경도
0	중구	3034	13729	9082	5154	중구	35.106609	129.030064
1	서구	2323	10446	21301	8779	서구	35.097924	129.022010

Panda 병합

- **merge**
  - 두 데이터 프레임의 공통 열 혹은 인덱스를 기준으로 두 개의 테이블을 합침
  - 기준되는 열이나 행이 **키(key)**

	항목	지역	매립처리량	소각처리량	재활용처리량	음식물류발생량
0		중구	3034	13729	9082	5154
1		서구	2323	10446	21301	8779
2		동구	1653	8297	17968	6399
3		영도구	1219	8651	24616	9522
4		부산진구	17393	538	88453	27119
5		동래구	9195	17	56526	20128
6		남구	8042	158	57921	15548
7		북구	10498	97	63879	24201
8		해운대구	5251	31924	64667	40043
9		사하구	3089	32671	70923	21183
10		금정구	7240	222	56153	21273
11		강서구	9742	4788	25977	10810
12		연제구	1831	14521	37115	14122
13		수영구	2205	11987	28135	13289
14		사상구	9937	216	57110	24161
15		기장군	2290	13687	15170	16648

df1

	지역	위도	경도
0	중구	35.106609	129.030064
1	서구	35.097924	129.022010
2	동구	35.129344	129.043314
3	영도구	35.091212	129.065700
4	부산진구	35.163087	129.051213
5	동래구	35.204841	129.081425
6	남구	35.136578	129.082053
7	북구	35.198325	128.987705
8	해운대구	35.163106	129.161390
9	사하구	35.104451	128.972650
10	금정구	35.242974	129.089958
11	강서구	35.212231	128.978379
12	연제구	35.176504	129.077611
13	수영구	35.145615	129.110889
14	사상구	35.152640	128.988623
15	기장군	35.244600	129.220059

df2

```
m3 = pd.merge(df1, df2, on='지역', how='inner')
m3
```

	지역	매립처리량	소각처리량	재활용처리량	음식물류발생량	위도	경도
0	중구	3034	13729	9082	5154	35.106609	129.030064
1	서구	2323	10446	21301	8779	35.097924	129.022010
2	동구	1653	8297	17968	6399	35.129344	129.043314
3	영도구	1219	8651	24616	9522	35.091212	129.065700
4	부산진구	17393	538	88453	27119	35.163087	129.051213
5	동래구	9195	17	56526	20128	35.204841	129.081425
6	남구	8042	158	57921	15548	35.136578	129.082053
7	북구	10498	97	63879	24201	35.198325	128.987705
8	해운대구	5251	31924	64667	40043	35.163106	129.161390
9	사하구	3089	32671	70923	21183	35.104451	128.972650
10	금정구	7240	222	56153	21273	35.242974	129.089958
11	강서구	9742	4788	25977	10810	35.212231	128.978379
12	연제구	1831	14521	37115	14122	35.176504	129.077611
13	수영구	2205	11987	28135	13289	35.145615	129.110889
14	사상구	9937	216	57110	24161	35.152640	128.988623
15	기장군	2290	13687	15170	16648	35.244600	129.220059

응용문제

기상청 자료 개방 사이트에서 1968년 이후
기온자료를 받아서 시각화하시오.

기초자료

	A	B	C	D
1	지점	지점명	위도	경도
29	90	속초	38.2509	128.5647
30	92	양양공항	38.0667	128.6667
31	93	북춘천	37.9475	127.7547
32	93	북춘천	37.9474	127.7544
33	94	광덕산	38.1172	127.4333
34	95	철원	38.1479	127.3042
35	96	독도	37.2395	131.8698
36	98	동두천	37.9019	127.0607
37	99	파주	37.8859	126.7665
38	99	문산	37.8859	126.7665

	A	B	C	D	E
1	지점	일시	평균	최고	최저
2	90	1968-01-01	-0.7	11.4	-11.4
3	90	1968-02-01	-2.1	9.7	-9.9
4	90	1968-03-01	5.9	20.5	-3.2
5	90	1968-04-01	10.5	21.4	4
6	90	1968-05-01	14.6	24.5	9.3
7	90	1968-06-01	18.5	29.2	11.4
8	90	1968-07-01	22.8	35.8	17.1
9	90	1968-08-01	23.3	31.9	16.6
10	90	1968-09-01	19.9	27.1	11
11	90	1968-10-01	13.3	23.3	6.9
12	90	1968-11-01	9.5	20.8	-5.4
13	90	1968-12-01	5	16.7	-6.8

지점.csv
:중복자료 처리

기온.csv

합치기

기초자료

```
import pandas as pd

#외부 데이터 가져오기
df1 = pd.read_csv('지점정보.csv', engine='python')
df1.head()
```

	지점	지점명	위도	경도
0	3	선봉	42.3167	130.4000
1	5	삼지연	41.8167	128.3167
2	8	청진	41.7833	129.8167
3	14	중강	41.7833	126.8833
4	16	혜산	41.4000	128.1667

```
len(df1)
```

4272

```
df2 = pd.read_csv('기온.csv', engine='python')
df2.head()
```

	지점	일시	평균	최고	최저
0	90	1968-01-01	-0.7	11.4	-11.4
1	90	1968-02-01	-2.1	9.7	-9.9
2	90	1968-03-01	5.9	20.5	-3.2
3	90	1968-04-01	10.5	21.4	4.0
4	90	1968-05-01	14.6	24.5	9.3

```
len(df2)
```

45555

```
#기온 데이터에 포함
df_area_list = df2['지점'].drop_duplicates().tolist()
len(df_area_list)
```

102

기초자료

```
import pandas as pd

#외부 데이터 가져오기
df1 = pd.read_csv('지점정보.csv', engine='python')
df1.head()
```

	지점	지점명	위도	경도
0	3	선봉	42.3167	130.4000
1	5	삼지연	41.8167	128.3167
2	8	청진	41.7833	129.8167
3	14	중강	41.7833	126.8833
4	16	혜산	41.4000	128.1667

```
len(df1)
```

4272

```
#기온 데이터에 포함
df_area_list = df2['지점'].drop_duplicates().tolist()
len(df_area_list)
```

102

```
#기온 데이터에 포함된 지점정보만 추출
df_area = df1[df1['지점'].isin(df_area_list)]
df_area.head()
```

	지점	지점명	위도	경도
27	90	속초	38.2509	128.5647
29	93	북춘천	37.9475	127.7547
30	93	북춘천	37.9474	127.7544
32	95	철원	38.1479	127.3042
34	98	동두천	37.9019	127.0607

```
#중복데이터 포함한 자료 개수
len(df_area)
```

138

기초자료

```
#기온 데이터에 포함된 지점정보만 추출
df_area  =  df1[df1['지점'].isin(df_area_list)]
df_area.head()
```

	지점	지점명	위도	경도
27	90	속초	38.2509	128.5647
29	93	북춘천	37.9475	127.7547
30	93	북춘천	37.9474	127.7544
32	95	철원	38.1479	127.3042
34	98	동두천	37.9019	127.0607

```
#중복데이터 포함한 자료 개수
len(df_area)
```

138

```
#지점 정보의 중복데이터 제거
#중복해서 나오는 지점번호의 첫번째 자료만 취득
#삭제할 인덱스 리스트 작성
old = 0
droplst = []
for i in df_area.index :
    if df_area['지점'][i] == old :
        droplst.append(i)

    old = df_area['지점'][i]

#삭제할 인덱스리스트를 이용하여 자료 삭제
df_area = df_area.drop(droplst, 0)
df_area.head()
```

	지점	지점명	위도	경도
27	90	속초	38.2509	128.5647
29	93	북춘천	37.9475	127.7547
32	95	철원	38.1479	127.3042
34	98	동두천	37.9019	127.0607
35	99	파주	37.8859	126.7665

```
#중복된 자료 삭제한 후 자료 개수
len(df_area)
```

102

Panda 병합

- **merge**
  - 두 데이터 프레임의 공통 열 혹은 인덱스를 기준으로 두 개의 테이블을 합침
  - 기준되는 열이나 행이 **키(key)**

```
#병합
df = pd.merge(df2, df_area, on='지점', how='inner')
df.head()
```

	지점	일시	평균	최고	최저	지점명	위도	경도
0	90	1968-01-01	-0.7	11.4	-11.4	속초	38.2509	128.5647
1	90	1968-02-01	-2.1	9.7	-9.9	속초	38.2509	128.5647
2	90	1968-03-01	5.9	20.5	-3.2	속초	38.2509	128.5647
3	90	1968-04-01	10.5	21.4	4.0	속초	38.2509	128.5647
4	90	1968-05-01	14.6	24.5	9.3	속초	38.2509	128.5647

```
#병합 후 자료 개수 확인 : 병합전 자료 개수와 동일
len(df)
```

```
45555
```

Panda 병합

```
#지점 확인
#df_area  = df['지점명'].drop_duplicates().tolist()

check = df[['지점','지점명']].drop_duplicates()
check.head()
```

	지점	지점명
0	90	속초
619	93	북춘천
653	95	철원
1032	98	등두천
1291	99	파주

```
len(check)
```

102

```
#지점 확인
#df_area  = df['지점명'].drop_duplicates().tolist()

area = df['지점명'].drop_duplicates()
len(area)
```

101

지점명 중복되는 자료찾기

```
old = ''

for i in check.index :
    if (old == check['지점명'][i]) : print(i, check['지점명'][i])
    else : old = check['지점명'][i]
```

22561 성산

```
#중복 자료 확인
df.loc[df['지점명'] == '성산', ['지점', '지점명']].drop_duplicates()
```

	지점	지점명
22132	187	성산
22561	188	성산

```
#중복자료 지점명 변경
df.loc[df['지점']==187,'지점명'] = '성산1'

check = df['지점명'].drop_duplicates()
len(check)
```

102

Pandas 날짜 형식

```
#날짜 데이터
#열 타입 변경
df['일시'] = pd.to_datetime(df['일시'])

#년도 월 열 삽입
df['년도'] = df['일시'].dt.year
df['월'] = df['일시'].dt.month

df.head()
```

	지점	일시	평균	최고	최저	지점명	위도	경도	년도	월
0	90	1968-01-01	-0.7	11.4	-11.4	속초	38.2509	128.5647	1968	1
1	90	1968-02-01	-2.1	9.7	-9.9	속초	38.2509	128.5647	1968	2
2	90	1968-03-01	5.9	20.5	-3.2	속초	38.2509	128.5647	1968	3
3	90	1968-04-01	10.5	21.4	4.0	속초	38.2509	128.5647	1968	4
4	90	1968-05-01	14.6	24.5	9.3	속초	38.2509	128.5647	1968	5

Pandas 자료 추출

```
#주요 지역 추출
mainarea = ['서울', '부산', '광주','대전','제주','강릉']
dfMain = df[df['지점명'].isin(mainarea)]

mainyear = list(range(1970,2018))
dfMain = dfMain[dfMain['년도'].isin(mainyear)]

dfMain.head()
```

	지점	일시	평균	최고	최저	지점명	위도	경도	년도	월
3197	105	1970-01-01	-1.2	12.6	-14.7	강릉	37.7515	128.891	1970	1
3198	105	1970-02-01	2.2	17.1	-9.5	강릉	37.7515	128.891	1970	2
3199	105	1970-03-01	2.4	21.6	-7.9	강릉	37.7515	128.891	1970	3
3200	105	1970-04-01	11.4	25.7	-0.2	강릉	37.7515	128.891	1970	4
3201	105	1970-05-01	17.7	31.9	9.4	강릉	37.7515	128.891	1970	5

Pandas 그룹화 후 인덱스 변경

```
#지점별 년도별 평균기온의 평균
dfMain_year = dfMain.groupby(['지점명', '년도'])['평균'].mean()

dfMain_year
```

인덱스

```
지점명  년도
강릉    1970    12.083333
        1971    12.250000
        1972    12.583333
        1973    12.850000
        1974    11.608333
        1975    12.650000
```

```
#인덱스를 컬럼으로 변환
dfMain_year = dfMain_year.reset_index()
dfMain_year.head()
```

	지점명	년도	평균
0	강릉	1970	12.083333
1	강릉	1971	12.250000
2	강릉	1972	12.583333
3	강릉	1973	12.850000
4	강릉	1974	11.608333

- set_index :
  기존의 행 인덱스를 제거하고 데이터 열 중 하나를 인덱스로 설정
- reset_index :
  기존의 행 인덱스를 제거하고 인덱스를 마지막 데이터 열로 추가

Pandas 피벗테이블 생성

```
#피벗 테이블 만들기
df3 = dfMain_year.pivot('년도', '지점명','평균')
df3 = df3.reset_index()
df3.head()
```

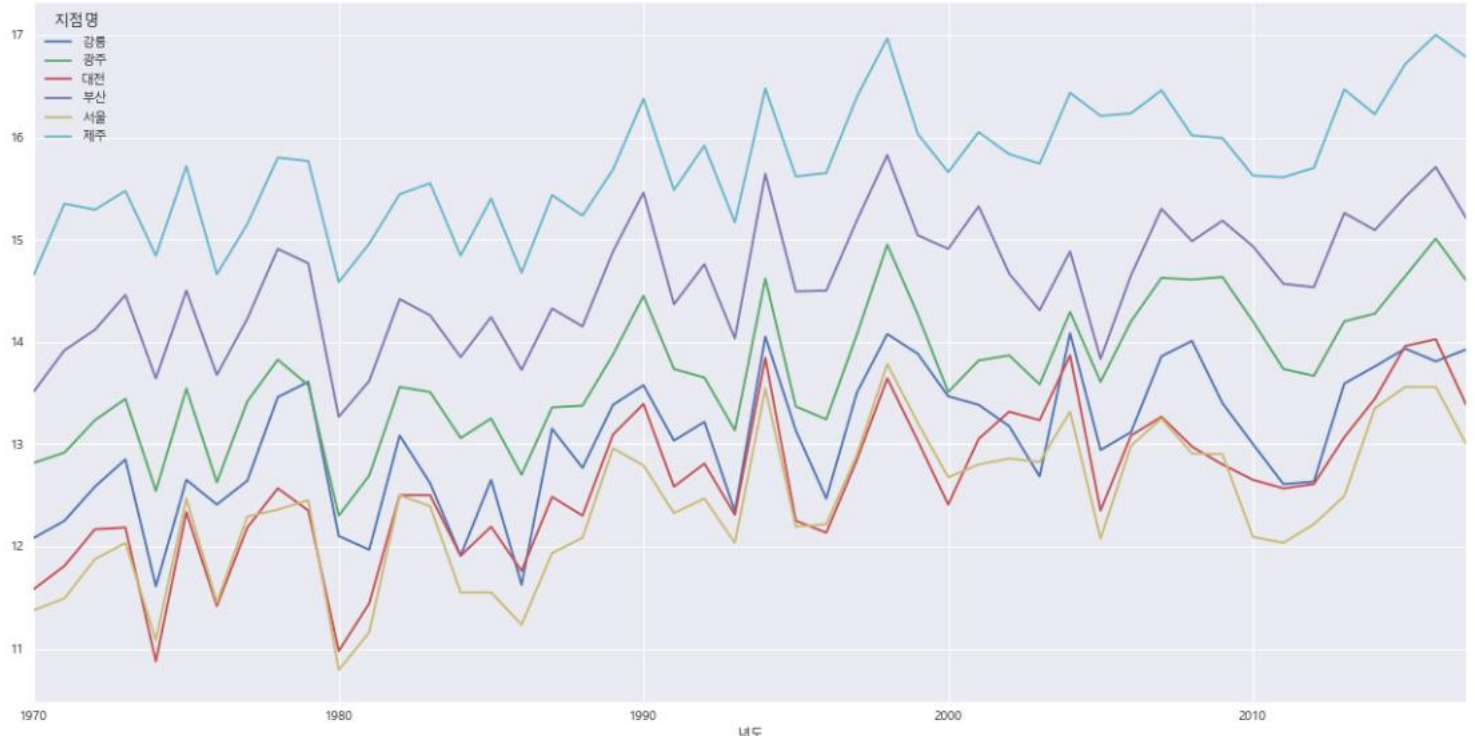
	지점명	년도	강릉	광주	대전	부산	서울	제주
0		1970	12.083333	12.816667	11.583333	13.516667	11.375000	14.658333
1		1971	12.250000	12.916667	11.808333	13.916667	11.491667	15.350000
2		1972	12.583333	13.233333	12.166667	14.116667	11.875000	15.291667
3		1973	12.850000	13.441667	12.183333	14.458333	12.033333	15.475000
4		1974	11.608333	12.541667	10.875000	13.641667	11.083333	14.841667

```
df3 = df3.set_index('년도')
df3.head()
```

지점명	강릉	광주	대전	부산	서울	제주
년도						
1970	12.083333	12.816667	11.583333	13.516667	11.375000	14.658333
1971	12.250000	12.916667	11.808333	13.916667	11.491667	15.350000
1972	12.583333	13.233333	12.166667	14.116667	11.875000	15.291667
1973	12.850000	13.441667	12.183333	14.458333	12.033333	15.475000
1974	11.608333	12.541667	10.875000	13.641667	11.083333	14.841667

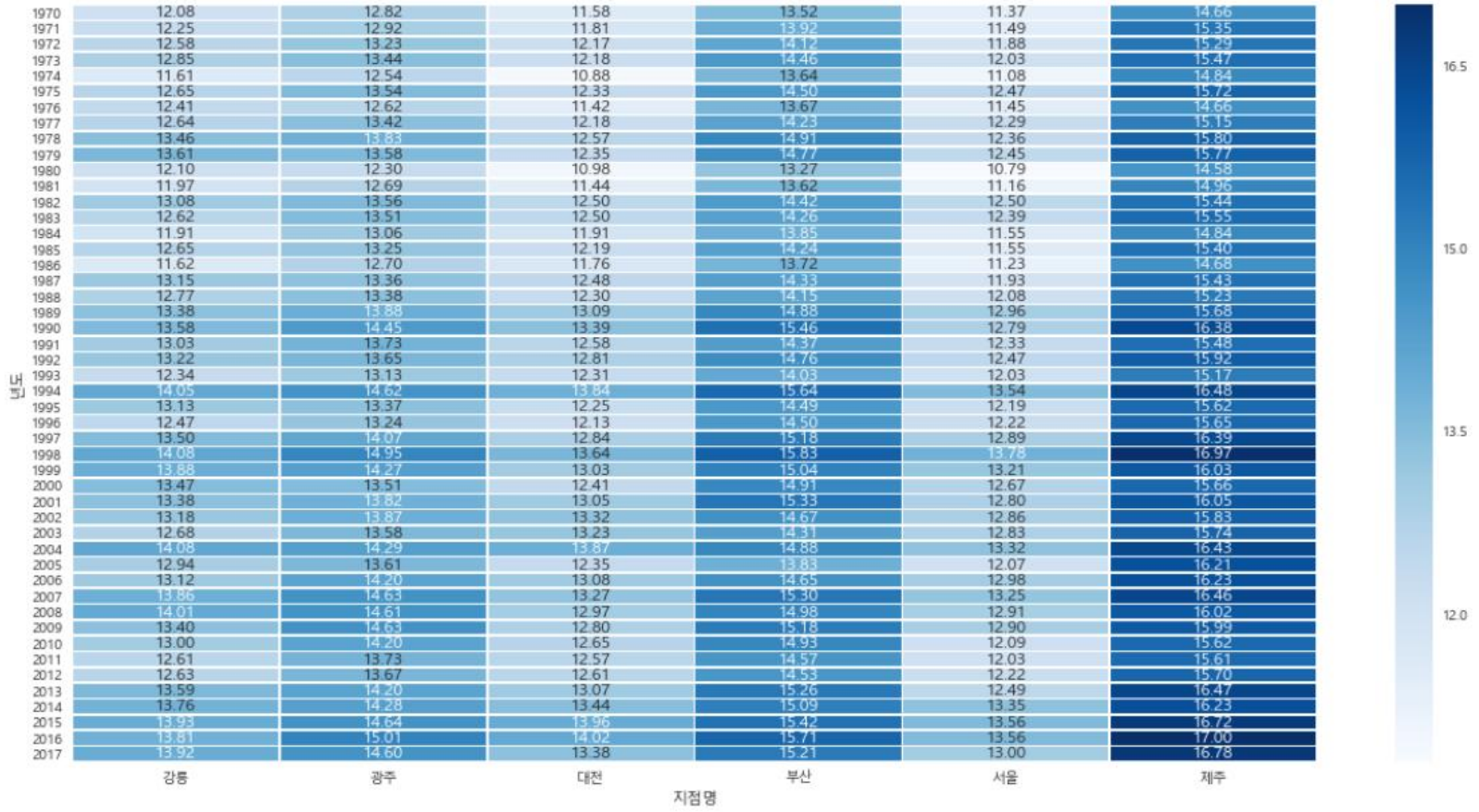
Pandas 시각화

```
# import matplotlib.pyplot as plt
#한글 폰트 사용
from matplotlib import font_manager, rc
font_name = font_manager.FontProperties(fname="c:/Windows/Fonts/malgun.ttf").get_name()
rc('font', family=font_name)

df3.plot(figsize=(20,10))
plt.show()
```



Pandas 시각화

```
import seaborn as sns

plt.figure(figsize=(20,10))
sns.heatmap(data = df3, annot=True, fmt = '.2f', linewidths=.5, cmap='Blues')
plt.show()
```



년도	강릉	광주	대전	부산	서울	제주
1970	12.08	12.82	11.58	13.52	11.37	14.66
1971	12.25	12.92	11.81	13.92	11.49	15.35
1972	12.58	13.23	12.17	14.12	11.88	15.29
1973	12.85	13.44	12.18	14.46	12.03	15.47
1974	11.61	12.54	10.88	13.64	11.08	14.84
1975	12.65	13.54	12.33	14.50	12.47	15.72
1976	12.41	12.62	11.42	13.67	11.45	14.66
1977	12.64	13.42	12.18	14.23	12.29	15.15
1978	13.46	13.83	12.57	14.91	12.36	15.80
1979	13.61	13.58	12.35	14.77	12.45	15.77
1980	12.10	12.30	10.98	13.27	10.79	14.58
1981	11.97	12.69	11.44	13.62	11.16	14.96
1982	13.08	13.56	12.50	14.42	12.50	15.44
1983	12.62	13.51	12.50	14.26	12.39	15.55
1984	11.91	13.06	11.91	13.85	11.55	14.84
1985	12.65	13.25	12.19	14.24	11.55	15.40
1986	11.62	12.70	11.76	13.72	11.23	14.68
1987	13.15	13.36	12.48	14.33	11.93	15.43
1988	12.77	13.38	12.30	14.15	12.08	15.23
1989	13.38	13.88	13.09	14.88	12.96	15.68
1990	13.58	14.45	13.39	15.46	12.79	16.38
1991	13.03	13.73	12.58	14.37	12.33	15.48
1992	13.22	13.65	12.81	14.76	12.47	15.92
1993	12.34	13.13	12.31	14.03	12.03	15.17
1994	14.05	14.62	13.84	15.64	13.54	16.48
1995	13.13	13.37	12.25	14.49	12.19	15.62
1996	12.47	13.24	12.13	14.50	12.22	15.65
1997	13.50	14.07	12.84	15.18	12.89	16.39
1998	14.08	14.95	13.64	15.83	13.78	16.97
1999	13.88	14.27	13.03	15.04	13.21	16.03
2000	13.47	13.51	12.41	14.91	12.67	15.66
2001	13.38	13.82	13.05	15.33	12.80	16.05
2002	13.18	13.87	13.32	14.67	12.86	15.83
2003	12.68	13.58	13.23	14.31	12.83	15.74
2004	14.08	14.29	13.87	14.88	13.32	16.43
2005	12.94	13.61	12.35	13.83	12.07	16.21
2006	13.12	14.20	13.08	14.65	12.98	16.23
2007	13.86	14.63	13.27	15.30	13.25	16.46
2008	14.01	14.61	12.97	14.98	12.91	16.02
2009	13.40	14.63	12.80	15.18	12.90	15.99
2010	13.00	14.20	12.65	14.93	12.09	15.62
2011	12.61	13.73	12.57	14.57	12.03	15.61
2012	12.63	13.67	12.61	14.53	12.22	15.70
2013	13.59	14.20	13.07	15.26	12.49	16.47
2014	13.76	14.28	13.44	15.09	13.35	16.23
2015	13.93	14.64	13.96	15.42	13.56	16.72
2016	13.81	15.01	14.02	15.71	13.56	17.00
2017	13.92	14.60	13.38	15.21	13.00	16.78