기상 데이터 전처리 및 저장

In [1]:

```
# 필요 패키지 로딩

import pandas as pd
import matplotlib.pyplot as plt
import numpy as np
import re
from scipy.stats.mstats import gmean
```

1. 데이터 확인

In [10]:

```
# 2016년 기상 데이터 확인

weather_2016 = pd.read_csv(f'../../lawdata/weather/종기상관측_ASOS_2016.csv', encoding = 'cp949')

print(weather_2016.info())
print(weather_2016.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34496 entries, 0 to 34495
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   지점              34496 non-null  int64
 1   지점명             34496 non-null  object
 2   일시              34496 non-null  object
 3   평균기온(°C)        34470 non-null  float64
 4   최저기온(°C)        34493 non-null  float64
 5   최고기온(°C)        34495 non-null  float64
 6   강수 계속시간(hr)     3267 non-null   float64
 7   일강수량(mm)        13323 non-null  float64
 8   평균 풍속(m/s)      34493 non-null  float64
 9   최다풍향(16방위)      34448 non-null  float64
 10  평균 현지기압(hPa)    34465 non-null  float64
 11  일 최심신적설(cm)     213 non-null    float64
dtypes: float64(9), int64(1), object(2)
memory usage: 3.2+ MB
None
   지점 지점명          일시  평균기온(°C)  최저기온(°C)  최고기온(°C)  강수 계속시간(hr)  일강수량(mm)  \
0  90  속초  2016-01-01       3.6       -2.3        7.6           NaN        NaN
1  90  속초  2016-01-02       8.4        5.0       11.5           NaN        NaN
2  90  속초  2016-01-03       6.7        2.4        9.8           NaN        NaN
3  90  속초  2016-01-04       5.4        0.8        9.0           NaN        NaN
4  90  속초  2016-01-05       0.6       -2.6        4.4           NaN        NaN

   평균 풍속(m/s)  최다풍향(16방위)  평균 현지기압(hPa)  일 최심신적설(cm)
0         2.3        290.0         1024.2           NaN
1         3.1        270.0         1017.0           NaN
2         1.9        290.0         1015.6           NaN
3         2.7        290.0         1017.2           NaN
4         2.5        290.0         1021.4           NaN
```

In [11]:

```
# 2017년 기상 데이터 확인

weather_2017 = pd.read_csv(f'../../lawdata/weather/종기상관측_ASOS_2017.csv', encoding = 'cp949')

print(weather_2017.info())
print(weather_2017.head())
```

```
<class 'pandas.core.frame.DataFrame'>
```

```
RangeIndex: 34670 entries, 0 to 34669
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   지점              34670 non-null  int64
 1   지점명             34670 non-null  object
 2   일시              34670 non-null  object
 3   평균기온(°C)        34615 non-null  float64
 4   최저기온(°C)        34665 non-null  float64
 5   최고기온(°C)        34662 non-null  float64
 6   강수 계속시간(hr)     3101 non-null   float64
 7   일강수량(mm)        12519 non-null  float64
 8   평균 풍속(m/s)      34659 non-null  float64
 9   최다풍향(16방위)      34608 non-null  float64
 10  평균 현지기압(hPa)    34611 non-null  float64
 11  일 최심신적설(cm)     247 non-null    float64
dtypes: float64(9), int64(1), object(2)
memory usage: 3.2+ MB
None
```

	지점	지점명	일시	평균기온(°C)	최저기온(°C)	최고기온(°C)	강수 계속시간(hr)	일강수량(mm)	\
0	90	속초	2017-01-01	6.0	1.5	9.8	NaN	NaN	
1	90	속초	2017-01-02	8.2	4.7	10.5	NaN	NaN	
2	90	속초	2017-01-03	6.2	1.7	11.3	NaN	NaN	
3	90	속초	2017-01-04	5.6	1.7	9.8	NaN	NaN	
4	90	속초	2017-01-05	2.8	1.3	4.0	NaN	15.5	

	평균 풍속(m/s)	최다풍향(16방위)	평균 현지기압(hPa)	일 최심신적설(cm)
0	1.7	250.0	1022.1	NaN
1	2.5	270.0	1016.6	NaN
2	2.0	290.0	1017.9	NaN
3	2.2	290.0	1020.5	NaN
4	1.5	290.0	1028.1	NaN

In [12]:

```
# 2018년 기상 데이터 확인

weather_2018 = pd.read_csv(f'../../lawdata/weather/종기상관측_ASOS_2018.csv', encoding = 'cp949')

print(weather_2018.info())
print(weather_2018.head())
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 34674 entries, 0 to 34673
Data columns (total 12 columns):
 #   Column          Non-Null Count  Dtype
---  ------          --------------  -----
 0   지점              34674 non-null  int64
 1   지점명             34674 non-null  object
 2   일시              34674 non-null  object
 3   평균기온(°C)        34630 non-null  float64
 4   최저기온(°C)        34672 non-null  float64
 5   최고기온(°C)        34673 non-null  float64
 6   강수 계속시간(hr)     3071 non-null   float64
 7   일강수량(mm)        12166 non-null  float64
 8   평균 풍속(m/s)      34645 non-null  float64
 9   최다풍향(16방위)      34417 non-null  float64
 10  평균 현지기압(hPa)    34624 non-null  float64
 11  일 최심신적설(cm)     268 non-null    float64
dtypes: float64(9), int64(1), object(2)
memory usage: 3.2+ MB
None
```

	지점	지점명	일시	평균기온(°C)	최저기온(°C)	최고기온(°C)	강수 계속시간(hr)	일강수량(mm)	\
0	90	속초	2018-01-01	1.0	-3.2	4.2	NaN	NaN	
1	90	속초	2018-01-02	1.5	-2.1	5.6	NaN	NaN	
2	90	속초	2018-01-03	-1.6	-5.5	3.3	NaN	NaN	
3	90	속초	2018-01-04	-1.0	-5.7	2.2	NaN	NaN	
4	90	속초	2018-01-05	1.5	-1.4	7.2	NaN	NaN	

	평균 풍속(m/s)	최다풍향(16방위)	평균 현지기압(hPa)	일 최심신적설(cm)
0	2.6	270.0	1021.8	NaN
1	2.9	290.0	1022.6	NaN
2	1.6	290.0	1025.0	NaN
3	1.5	290.0	1023.4	NaN
4	1.2	290.0	1014.9	NaN

2. 각 연도별 데이터 필요 변수 추출 및 저장

In [13]:

```
# 시도코드 컬럼 생성 함수

def sido_func(vector):
    for i in sido_list:
        if vector in sido_list[i] :
            return i
            pass
        pass
    pass

def sido_code_func(vector2):
    for i in sido_code:
        if vector2 == sido_code[i]:
            return str(i)
            pass
        pass
    pass

# 시도코드 리스트

sido_list = {
'강원' : ['속초', '북춘천', '철원', '대관령', '춘천', '북강릉', '강릉', '동해', '원주', '영월', '인제', '홍천
, '태백', '정선군'],
'경기' : ['동두천', '파주', '수원', '양평', '이천'],
'충북' : ['충주', '추풍령', '제천', '보은', '청주'],
'충남' : ['서산', '홍성', '보령', '부여', '금산', '천안'],
'대전' : ['대전'],
'경북' : ['울릉도', '울진', '안동', '상주', '포항', '봉화', '영주', '문경', '청송군', '영덕', '의성', '구미',
'영천', '경주시'],
'경남': ['창원', '통영', '진주', '김해시', '북창원', '양산시', '의령군', '함양군', '밀양', '산청', '거제', '남
해', '합천', '거창'],
'전북' : ['군산', '전주', '고창', '부안', '임실', '정읍', '남원', '장수', '고창군', '순창군', '장흥'],
'전남' : ['목포', '여수', '흑산도', '완도', '순천', '진도', '영광군', '보성군', '강진군', '해남', '고흥', '광
양시', '진도군'],
'서울' : ['서울'],
'인천' : ['백령도', '인천', '강화'],
'대구' : ['대구'],
'울산' : ['울산'],
'광주' : ['광주'],
'부산' : ['부산'],
'제주' : ['제주', '고산', '성산', '서귀포']}

sido_code = {
     '42' : '강원',
     '41' : '경기',
     '43' : '충북',
     '44' : '충남',
     '30' : '대전',
     '47' : '경북',
     '48' : '경남',
     '45' : '전북',
     '46' : '전남',
     '11' : '서울',
     '28' : '인천',
     '27' : '대구',
     '31' : '울산',
     '29' : '광주',
     '26' : '부산',
     '49' : '제주',
     '36' : '세종'
}
```

In [14]:

```
## 각 연도별 데이터 필요 변수 추출 및 저장

for i in [2016,2017,2018]:
    df = pd.read_csv(f'../../lawdata/weather/종기상관측_ASOS_{i}.csv', encoding = 'cp949')
```

```
    # 대량의 결측치 포함하는 컬럼 결측치 처리(눈, 비 측정 X = 0으로 대체)
    df[['일 최심신적설(cm)','일강수량(mm)','강수 계속시간(hr)']] = df[['일 최심신적설(cm)','일강수량(mm)','
강수 계속시간(hr)']].fillna(value = 0)

    # 컬럼명 통합을 위해 지점명을 시도로 구분하는 함수 적용
    df['시도'] = df['지점명'].apply(sido_func)

    # 세종시 데이터 생성을 위해 대전, 청주, 공주 데이터 추출 후 추가
    df_mk_sejong = df[df['지점명'].str.contains('대전|청주|공주', regex=True)]
    df_mk_sejong['시도'] = df_mk_sejong['지점명'].apply(lambda x :'세종')
    df_concat = pd.concat([df,df_mk_sejong])

    # 구분된 시도별로 함수를 이용하여 시도코드 컬럼 생성
    df_concat['시도코드'] = df_concat['시도'].apply(sido_code_func)

    # 시도코드, 일시를 통합하여 키를 생성하기 위해 그룹바이 설정.
    df_group = df_concat.groupby(['시도코드','일시']).mean().reset_index()

    # 불필요한 변수 삭제
    df_group_1 = df_group.drop(['지점','최다풍향(16방위)'],axis=1)

    # 최다풍향 빈도수와 평균 풍속을 이용하여 비교하는 컬럼 생성
    df_temp = df_concat.groupby(['시도코드','일시','최다풍향(16방위)'])['평균 풍속(m/s)'].agg(['count','
sum']).sort_values(['시도코드','일시','count','sum'],ascending=False)

    # 임시 리스트 변수 생성
    temp_sido = []
    temp_day = []
    temp_wind = []

    # 임시 리스트에 데이터 넣기
    for j in range(0,len(df_temp.index)):
        temp_sido.append(df_temp.index[j][0])
        temp_day.append(df_temp.index[j][1])
        temp_wind.append(df_temp.index[j][2])
        pass

    # 리스트 결합
    df_wind = pd.DataFrame({'시도코드' : temp_sido ,'일시' : temp_day, '최다풍향(16방위)' : temp_wind})

    # 일시 별 최다풍향 데이터프레임 생성
    df_wind = df_wind.groupby(['시도코드','일시']).head(1).sort_values(['시도코드','일시']).reset_index
()

    # 새로 생성한 최다풍향 컬럼과 기존 데이터프레임 조인
    df_group_1 = pd.merge(df_group_1,df_wind, on = ['시도코드','일시']).drop('index',axis=1)

    # 데이터 저장
    df_group_1.to_csv(f'../../lawdata/weather/기상관측_{i}_Fixed.csv', encoding = 'cp949', index = F
alse)
    pass
```

```
c:\app\python37\lib\site-packages\ipykernel_launcher.py:14: SettingWithCopyWarning:
A value is trying to be set on a copy of a slice from a DataFrame.
Try using .loc[row_indexer,col_indexer] = value instead

See the caveats in the documentation: https://pandas.pydata.org/pandas-
docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy
```

3. 시도코드, 일시를 이용하여 기상번호 키를 생성

In [15]:

```
# 데이터 타입 변환 함수
def com(vec):
    result = re.compile('(\d{4})[-](\d{2})[-](\d{2})').sub('\g<1>\g<2>\g<3>', vec)
    return result
```

In [16]:

```
# 데이터 로딩
df_weather_2016 = pd.read_csv(f'../../lawdata/weather/기상관측_2016_Fixed.csv', encoding = 'cp949')
```

```
df_weather_2017 = pd.read_csv(f'../../lawdata/weather/기상관측_2017_Fixed.csv', encoding = 'cp949')
df_weather_2018 = pd.read_csv(f'../../lawdata/weather/기상관측_2018_Fixed.csv', encoding = 'cp949')

# 데이터 정보 확인
df_weather_2016.info()
df_weather_2017.info()
df_weather_2018.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6222 entries, 0 to 6221
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   시도코드            6222 non-null   int64  
 1   일시              6222 non-null   object 
 2   평균기온(°C)        6222 non-null   float64
 3   최저기온(°C)        6222 non-null   float64
 4   최고기온(°C)        6222 non-null   float64
 5   강수 계속시간(hr)     6222 non-null   float64
 6   일강수량(mm)        6222 non-null   float64
 7   평균 풍속(m/s)      6222 non-null   float64
 8   평균 현지기압(hPa)    6221 non-null   float64
 9   일 최심신적설(cm)     6222 non-null   float64
 10  최다풍향(16방위)      6222 non-null   float64
dtypes: float64(9), int64(1), object(1)
memory usage: 534.8+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6202 entries, 0 to 6201
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   시도코드            6202 non-null   int64  
 1   일시              6202 non-null   object 
 2   평균기온(°C)        6201 non-null   float64
 3   최저기온(°C)        6202 non-null   float64
 4   최고기온(°C)        6201 non-null   float64
 5   강수 계속시간(hr)     6202 non-null   float64
 6   일강수량(mm)        6202 non-null   float64
 7   평균 풍속(m/s)      6202 non-null   float64
 8   평균 현지기압(hPa)    6202 non-null   float64
 9   일 최심신적설(cm)     6202 non-null   float64
 10  최다풍향(16방위)      6202 non-null   float64
dtypes: float64(9), int64(1), object(1)
memory usage: 533.1+ KB
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 6205 entries, 0 to 6204
Data columns (total 11 columns):
 #   Column          Non-Null Count  Dtype  
---  ------          --------------  -----  
 0   시도코드            6205 non-null   int64  
 1   일시              6205 non-null   object 
 2   평균기온(°C)        6205 non-null   float64
 3   최저기온(°C)        6205 non-null   float64
 4   최고기온(°C)        6205 non-null   float64
 5   강수 계속시간(hr)     6205 non-null   float64
 6   일강수량(mm)        6205 non-null   float64
 7   평균 풍속(m/s)      6205 non-null   float64
 8   평균 현지기압(hPa)    6205 non-null   float64
 9   일 최심신적설(cm)     6205 non-null   float64
 10  최다풍향(16방위)      6205 non-null   float64
dtypes: float64(9), int64(1), object(1)
memory usage: 533.4+ KB
```

In [17]:

```
# 함수를 사용하여 일시, 시도코드 타입 변환

df_weather_2016['일시'] = df_weather_2016.일시.apply(com).astype('int64')
df_weather_2016['시도코드']=df_weather_2016['시도코드'].astype('int64')

df_weather_2017['일시'] = df_weather_2017.일시.apply(com).astype('int64')
df_weather_2017['시도코드']=df_weather_2017['시도코드'].astype('int64')

df_weather_2018['일시'] = df_weather_2018.일시.apply(com).astype('int64')
df_weather_2018['시도코드']=df_weather_2018['시도코드'].astype('int64')
```

In [18]:

```
# 변환된 일시, 시도코드를 이용하여 기상번호 컬럼 생성

df_weather_2016['기상번호'] = df_weather_2016['시도코드'].astype('str') + df_weather_2016['일
시'].astype('str')
df_weather_2017['기상번호'] = df_weather_2017['시도코드'].astype('str') + df_weather_2017['일
시'].astype('str')
df_weather_2018['기상번호'] = df_weather_2018['시도코드'].astype('str') + df_weather_2018['일
시'].astype('str')

# 기상번호를 인덱스(고유키)로 설정
df_weather_2016.index = df_weather_2016['기상번호']
df_weather_2017.index = df_weather_2017['기상번호']
df_weather_2018.index = df_weather_2018['기상번호']

# 기상번호 컬럼 삭제

df_weather_2016 = df_weather_2016.drop(['기상번호'],axis=1)
df_weather_2017 = df_weather_2017.drop(['기상번호'],axis=1)
df_weather_2018 = df_weather_2018.drop(['기상번호'],axis=1)
```

In [19]:

```
# 데이터 확인
df_weather_2016
df_weather_2017
df_weather_2018
```

Out[19]:

기상번호	시도코드	일시	평균기온(°C)	최저기온(°C)	최고기온(°C)	강수 계속시간(hr)	일강수량(mm)	평균 풍속(m/s)	평균 현지기압(hPa)	일 최심신적설(cm)	최다풍향(16방위)
1120180101	11	20180101	-1.300	-5.100	3.800	0.0000	0.000	1.400	1016.800	0.0	290.0
1120180102	11	20180102	-1.800	-4.300	1.800	0.0000	0.000	1.800	1018.100	0.0	290.0
1120180103	11	20180103	-4.700	-7.100	-0.400	0.0000	0.000	2.200	1019.900	0.0	290.0
1120180104	11	20180104	-4.700	-8.700	-0.700	0.0000	0.000	1.400	1016.500	0.0	290.0
1120180105	11	20180105	-3.000	-5.600	1.600	0.0000	0.000	1.700	1010.300	0.0	290.0
...	...	...	...	...	...	...	...	...	...	...	...
4920181227	49	20181227	3.575	1.350	7.475	0.0000	0.000	5.925	1021.000	0.0	360.0
4920181228	49	20181228	1.550	0.100	3.300	6.6375	0.925	6.200	1025.100	0.0	20.0
4920181229	49	20181229	2.800	0.825	4.975	1.2375	0.075	5.050	1028.650	0.0	20.0
4920181230	49	20181230	3.300	1.500	4.800	1.3250	0.025	4.350	1029.850	0.0	50.0
4920181231	49	20181231	4.750	3.050	6.625	0.0000	0.000	3.775	1029.625	0.0	360.0

6205 rows × 11 columns

4. 데이터 저장

In [20]:

```
# 2016~2018년 기상데이터 저장

df_weather_2016.to_csv(f'../../lawdata/weather/기상관측2016real_Fixed.csv', encoding = 'cp949')
df_weather_2017.to_csv(f'../../lawdata/weather/기상관측2017real_Fixed.csv', encoding = 'cp949')
df_weather_2018.to_csv(f'../../lawdata/weather/기상관측2018real_Fixed.csv', encoding = 'cp949')
```

In [ ]:

```

```

In [ ]:

```
```