

```
In [25]: import numpy as np
import pandas as pd
import xarray as xr
from matplotlib import pyplot as plt
%matplotlib inline
```

```
In [26]: # 2. Air temperature in Shenzhen during the past 25 years
df2=pd.read_csv("Baoan_Weather_1998_2022.csv")
```

C:\Users\60918\AppData\Local\Temp\ipykernel_8044\2092050105.py:2: DtypeWarning: Columns (4, 8, 9, 10, 11, 14, 15, 24, 25, 27, 29, 31, 34, 37, 38, 40, 41, 45, 49, 50) have mixed types. Specify dtype option on import or set low_memory=False.

```
df2=pd.read_csv("Baoan_Weather_1998_2022.csv")
```

```
In [27]: # 2.1 Select the valid datas
# Create a new column "cTMP_1" to extract the temperature information(e.g. if TMP="+0186,1", cTMP_1= +0186)
df2["cTMP_1"]=df2["TMP"].str[0:5].astype(int)
# Create another new column "cTMP_2" to extract the last character, which can detect the data validity.
df2["cTMP_2"]=df2["TMP"].str[-1].astype(int)
# Locate the valid data(cTMP_2= 1 or 5, means passed all quality control checks) within the temperature range(-932 to +618)
df2_new=df2.loc[((df2["cTMP_2"]==1) | (df2["cTMP_2"]==5)) & ((df2["cTMP_1"]>=-932)&(df2["cTMP_1"]<=618))]
df2_new
```

Out[27]:

| | STATION | DATE | SOURCE | REPORT_TYPE | CALL_SIGN | QUALITY_CONTROL | AA1 | AA2 | AA3 | AG1 | ... | SI |
|--------|-------------|---------------------|--------|-------------|-----------|-----------------|-------------|-------------|-------------|-------|-----|-------|
| 0 | 59493099999 | 1998-01-01T00:00:00 | 4 | SY-MT | ZGSZ | V020 | 06,0000,9,1 | NaN | NaN | 0,000 | ... | 10184 |
| 1 | 59493099999 | 1998-01-01T01:00:00 | 4 | FM-15 | ZGSZ | V020 | NaN | NaN | NaN | 0,999 | ... | 99999 |
| 2 | 59493099999 | 1998-01-01T02:00:00 | 4 | FM-15 | ZGSZ | V020 | NaN | NaN | NaN | 0,999 | ... | 99999 |
| 3 | 59493099999 | 1998-01-01T03:00:00 | 4 | SY-MT | ZGSZ | V020 | NaN | NaN | NaN | 0,000 | ... | 10185 |
| 4 | 59493099999 | 1998-01-01T04:00:00 | 4 | FM-15 | ZGSZ | V020 | NaN | NaN | NaN | 0,999 | ... | 99999 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 235669 | 59493099999 | 2022-10-10T20:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | 99999 |
| 235670 | 59493099999 | 2022-10-10T21:00:00 | 4 | FM-12 | 99999 | V020 | 06,0000,9,1 | 12,0000,9,1 | 24,0000,9,1 | NaN | ... | 10177 |
| 235671 | 59493099999 | 2022-10-10T21:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | 99999 |
| 235672 | 59493099999 | 2022-10-10T22:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | 99999 |
| 235673 | 59493099999 | 2022-10-10T23:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | 99999 |

234864 rows × 56 columns

```
In [28]: # 2.2 Create two columns "Month" and "Year"
df2_new["Month"]=pd.to_datetime(df2_new["DATE"]).dt.month
df2_new["Year"]=pd.to_datetime(df2_new["DATE"]).dt.year
df2_new
```

C:\Users\60918\AppData\Local\Temp\ipykernel_8044\2609756307.py:2: 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 (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
df2_new["Month"]=pd.to_datetime(df2_new["DATE"]).dt.month
C:\Users\60918\AppData\Local\Temp\ipykernel_8044\2609756307.py:3: 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 (https://pandas.pydata.org/pandas-docs/stable/user_guide/indexing.html#returning-a-view-versus-a-copy)

```
df2_new["Year"]=pd.to_datetime(df2_new["DATE"]).dt.year
```

Out[28]:

| | STATION | DATE | SOURCE | REPORT_TYPE | CALL_SIGN | QUALITY_CONTROL | AA1 | AA2 | AA3 | AG1 | ... | TM |
|--------|-------------|---------------------|--------|-------------|-----------|-----------------|-------------|-------------|-------------|-------|-----|-------|
| 0 | 59493099999 | 1998-01-01T00:00:00 | 4 | SY-MT | ZGSZ | V020 | 06,0000,9,1 | NaN | NaN | 0,000 | ... | +0186 |
| 1 | 59493099999 | 1998-01-01T01:00:00 | 4 | FM-15 | ZGSZ | V020 | NaN | NaN | NaN | 0,999 | ... | +0220 |
| 2 | 59493099999 | 1998-01-01T02:00:00 | 4 | FM-15 | ZGSZ | V020 | NaN | NaN | NaN | 0,999 | ... | +0240 |
| 3 | 59493099999 | 1998-01-01T03:00:00 | 4 | SY-MT | ZGSZ | V020 | NaN | NaN | NaN | 0,000 | ... | +0221 |
| 4 | 59493099999 | 1998-01-01T04:00:00 | 4 | FM-15 | ZGSZ | V020 | NaN | NaN | NaN | 0,999 | ... | +0240 |
| ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| 235669 | 59493099999 | 2022-10-10T20:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | +0210 |
| 235670 | 59493099999 | 2022-10-10T21:00:00 | 4 | FM-12 | 99999 | V020 | 06,0000,9,1 | 12,0000,9,1 | 24,0000,9,1 | NaN | ... | +0201 |

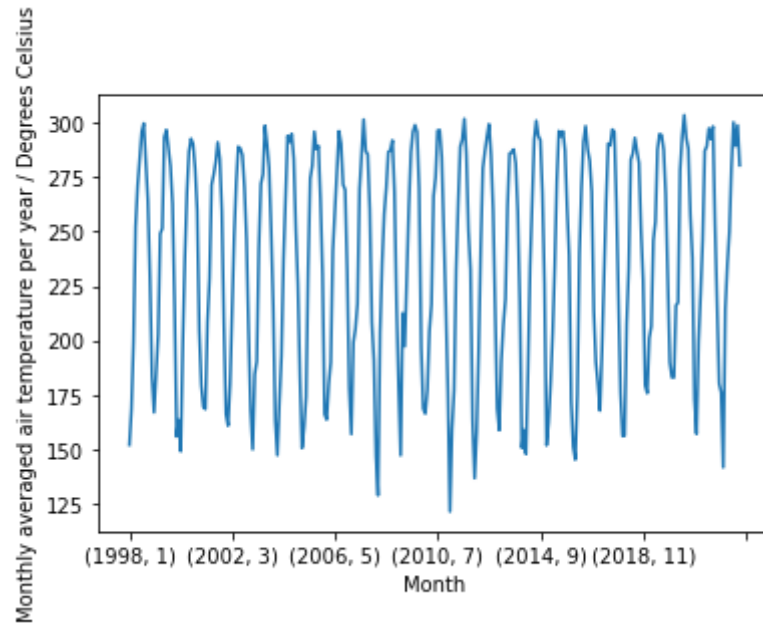
| | STATION | DATE | SOURCE | REPORT_TYPE | CALL_SIGN | QUALITY_CONTROL | AA1 | AA2 | AA3 | AG1 | ... | TA |
|---------------|-------------|---------------------|--------|-------------|-----------|-----------------|-----|-----|-----|-----|-----|-------|
| 235671 | 59493099999 | 2022-10-10T21:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | +0200 |
| 235672 | 59493099999 | 2022-10-10T22:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | +0200 |
| 235673 | 59493099999 | 2022-10-10T23:00:00 | 4 | FM-15 | 99999 | V020 | NaN | NaN | NaN | NaN | ... | +0200 |

234864 rows × 58 columns



```
In [29]: # 2.3.1 Considering the "Year" factor, plot the monthly averaged air temperature groupby "Year" & "Month"
Year_Monthly_TMP=df2_new.groupby(["Year", "Month"])["cTMP_1"].mean()
Year_Monthly_TMP.plot()
plt.xlabel("Month")
plt.ylabel("Monthly averaged air temperature per year / Degrees Celsius")
# From the output below, monthly averaged air temperature seemed to keep fluctuating along with year.
```

```
Out[29]: Text(0, 0.5, 'Monthly averaged air temperature per year / Degrees Celsius')
```



```
In [30]: # 2.3.2 Without considering the "Year" factor, plot the monthly averaged air temperature groupby "Month" for 25 years
Monthly_TMP=df2_new.groupby(["Month"])["cTMP_1"].mean()
Monthly_TMP.plot()
plt.xlabel("Month")
plt.ylabel("Monthly averaged air temperature / Degrees Celsius")
# From the output below, monthly averaged air temperature tended to increase from Jan to July, and decrease from July to Dec.
```

```
Out[30]: Text(0, 0.5, 'Monthly averaged air temperature / Degrees Celsius')
```

