

```
In [ ]: import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
import seaborn as sns  
from importlib import reload  
plt=reload=plt
```

```
In [ ]: data=pd.read_csv("/content/weather.csv")
         data.head()
```

	Formatted Date	Summary	Precip Type	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wind Bearing (degrees)	Visibility (km)
0	2006-04-01 00:00:00.000 +0200	Partly Cloudy	rain	9.472222	7.388889	0.89	14.1197	251.0	15.826
1	2006-04-01 01:00:00.000 +0200	Partly Cloudy	rain	9.355556	7.227778	0.86	14.2646	259.0	15.826
2	2006-04-01 02:00:00.000 +0200	Mostly Cloudy	rain	9.377778	9.377778	0.89	3.9284	204.0	14.956
3	2006-04-01 03:00:00.000 +0200	Partly Cloudy	rain	8.288889	5.944444	0.83	14.1036	269.0	15.826
4	2006-04-01 04:00:00.000 +0200	Mostly Cloudy	rain	8.755556	6.977778	0.83	11.0446	259.0	15.826

```
In [ ]: data.shape
```

```
Out[1]: (89010, 12)
```

```
In [ ]: data.columns
```

```
Out[ ]: Index(['Formatted Date', 'Summary', 'Precip Type', 'Temperature (C)',  
              'Apparent Temperature (C)', 'Humidity', 'Wind Speed (km/h)',  
              'Wind Bearing (degrees)', 'Visibility (km)', 'Loud Cover',  
              'Pressure (millibars)', 'Daily Summary'],  
              dtype='object')
```

```
In [ ]: data.dtypes
```

```
Out[ ]: Formatted Date      object
          Summary           object
          Precip Type        object
```

```
Temperature (C)          float64
Apparent Temperature (C) float64
Humidity                float64
Wind Speed (km/h)       float64
Wind Bearing (degrees)  float64
Visibility (km)          float64
Loud Cover               float64
Pressure (millibars)    float64
Daily Summary            object
dtype: object
```

```
In [ ]: data.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 89010 entries, 0 to 89009
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Formatted Date   89010 non-null   object  
 1   Summary          89009 non-null   object  
 2   Precip Type      88866 non-null   object  
 3   Temperature (C)  89009 non-null   float64 
 4   Apparent Temperature (C) 89009 non-null   float64 
 5   Humidity          89009 non-null   float64 
 6   Wind Speed (km/h) 89009 non-null   float64 
 7   Wind Bearing (degrees) 89009 non-null   float64 
 8   Visibility (km)   89009 non-null   float64 
 9   Loud Cover        89009 non-null   float64 
 10  Pressure (millibars) 89009 non-null   float64 
 11  Daily Summary    89009 non-null   object  
dtypes: float64(8), object(4)
memory usage: 8.1+ MB
```

```
In [ ]: data.describe()
```

	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wind Bearing (degrees)	Visibility (km)	Loud Cover
count	89009.000000	89009.000000	89009.000000	89009.000000	89009.000000	89009.000000	89009.0
mean	12.002990	10.943777	0.731399	10.824283	187.963622	10.305464	0.0
std	9.584068	10.720161	0.196718	6.986107	108.131207	4.152685	0.0
min	-21.822222	-27.716667	0.000000	0.000000	0.000000	0.000000	0.0
25%	4.838889	2.444444	0.600000	5.683300	112.000000	8.420300	0.0
50%	12.127778	12.127778	0.780000	9.949800	181.000000	10.046400	0.0
75%	18.866667	18.866667	0.890000	14.168000	290.000000	14.490000	0.0
max	39.905556	39.344444	1.000000	63.852600	359.000000	16.100000	0.0

```
In [ ]: data[["Date-Time","TZ"]]=data[["Formatted Date"]].str.split("+",expand=True)
df1=data.drop(columns="Formatted Date")
df1.head()
```

```
Out[ ]:  Summary  Precip Type  Temperature (C)  Apparent Temperature (C)  Humidity  Wind Speed (km/h)  Wind Bearing (degrees)  Visibility (km)  Loud Cover  F
          0          1          2          3          4          5          6          7          8          9          10         11
```

0	Partly Cloudy	rain	9.472222	7.388889	0.89	14.1197	251.0	15.8263	0.0
1	Partly Cloudy	rain	9.355556	7.227778	0.86	14.2646	259.0	15.8263	0.0
2	Mostly Cloudy	rain	9.377778	9.377778	0.89	3.9284	204.0	14.9569	0.0
3	Partly Cloudy	rain	8.288889	5.944444	0.83	14.1036	269.0	15.8263	0.0
4	Mostly Cloudy	rain	8.755556	6.977778	0.83	11.0446	259.0	15.8263	0.0

In []:

```
columns_order=["Date-Time","TZ","Summary","Precip Type","Temperature (C)","Apparent Temperature (C)", "Humidity", "Wind Speed (km/h)", "Wind Bearing (degrees)", "Visibility (km)", "Pressure (millibars)", "Daily Summary"]
df2=df1.reindex(columns=columns_order)
df3=df2.drop(columns="TZ")
df3.head()
```

Out[]:

	Date-Time	Summary	Precip Type	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wind Bearing (degrees)	Visibility (km)
0	2006-04-01 00:00:00.000	Partly Cloudy	rain	9.472222	7.388889	0.89	14.1197	251.0	15.826
1	2006-04-01 01:00:00.000	Partly Cloudy	rain	9.355556	7.227778	0.86	14.2646	259.0	15.826
2	2006-04-01 02:00:00.000	Mostly Cloudy	rain	9.377778	9.377778	0.89	3.9284	204.0	14.956
3	2006-04-01 03:00:00.000	Partly Cloudy	rain	8.288889	5.944444	0.83	14.1036	269.0	15.826
4	2006-04-01 04:00:00.000	Mostly Cloudy	rain	8.755556	6.977778	0.83	11.0446	259.0	15.826

```
In [ ]: df3["Date-Time"] = pd.to_datetime(df3["Date-Time"])
df3.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 89010 entries, 0 to 89009
Data columns (total 12 columns):
 #   Column           Non-Null Count  Dtype  
--- 
 0   Date-Time        89010 non-null   datetime64[ns]
 1   Summary          89009 non-null   object  
 2   Precip Type      88866 non-null   object  
 3   Temperature (C)  89009 non-null   float64 
 4   Apparent Temperature (C) 89009 non-null   float64 
 5   Humidity         89009 non-null   float64 
 6   Wind Speed (km/h) 89009 non-null   float64 
 7   Wind Bearing (degrees) 89009 non-null   float64 
 8   Visibility (km)  89009 non-null   float64 
 9   Loud Cover       89009 non-null   float64 
 10  Pressure (millibars) 89009 non-null   float64 
 11  Daily Summary    89009 non-null   object  
dtypes: datetime64[ns](1), float64(8), object(3)
memory usage: 8.1+ MB
```

```
In [ ]: df3["Year"] = pd.DatetimeIndex(df3["Date-Time"]).year
df3["Month"] = df3["Date-Time"].dt.month_name()
df3["day"] = df3["Date-Time"].dt.day
df3.head()
```

```
Out[ ]:
```

	Date-Time	Summary	Precip Type	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wind Bearing (degrees)	Visibility (km)	
0	2006-04-01 00:00:00	Partly Cloudy	rain	9.472222	7.388889	0.89	14.1197	251.0	15.8263	
1	2006-04-01 01:00:00	Partly Cloudy	rain	9.355556	7.227778	0.86	14.2646	259.0	15.8263	
2	2006-04-01 02:00:00	Mostly Cloudy	rain	9.377778	9.377778	0.89	3.9284	204.0	14.9569	
3	2006-04-01 03:00:00	Partly Cloudy	rain	8.288889	5.944444	0.83	14.1036	269.0	15.8263	
4	2006-04-01 04:00:00	Mostly Cloudy	rain	8.755556	6.977778	0.83	11.0446	259.0	15.8263	

```
In [ ]: df3["Wind Speed (km/h)"].describe()
```

```
Out[ ]: count    89009.000000
         mean     10.824283
         std      6.986107
         min      0.000000
         25%     5.683300
         50%     9.949800
         75%    14.168000
         max     63.852600
Name: Wind Speed (km/h), dtype: float64
```

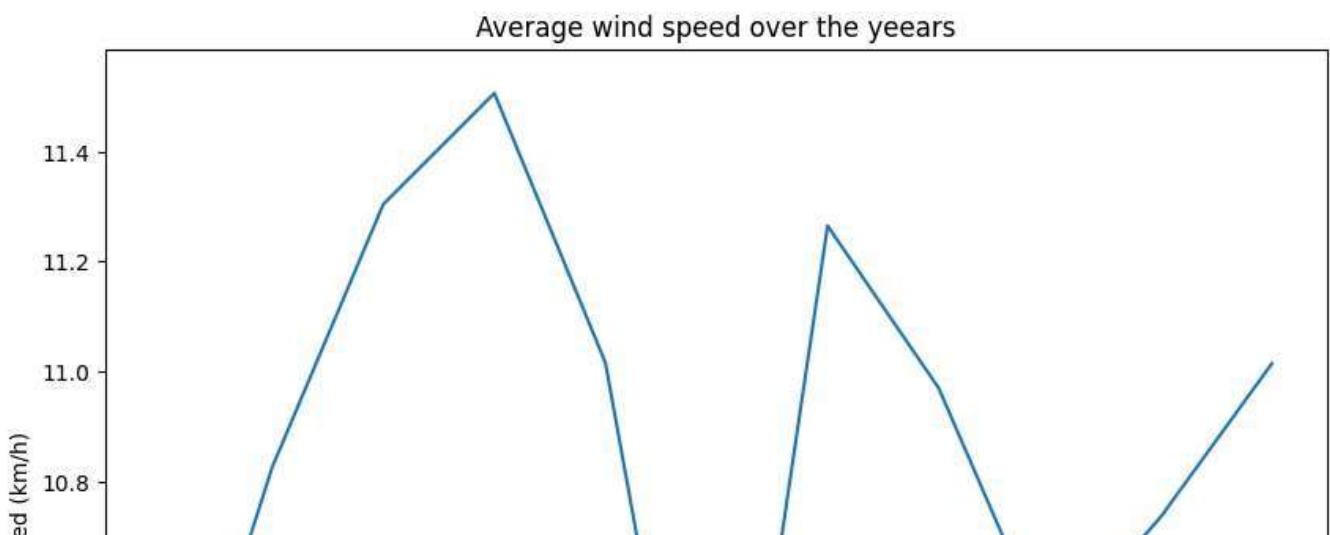
```
In [ ]: avg_wind_Speed=pd.DataFrame(df3.groupby("Year")["Wind Speed (km/h)"].mean())
avg_wind_Speed
#[ "Year", "AVG. Wind Speed (km/hr)"]
```

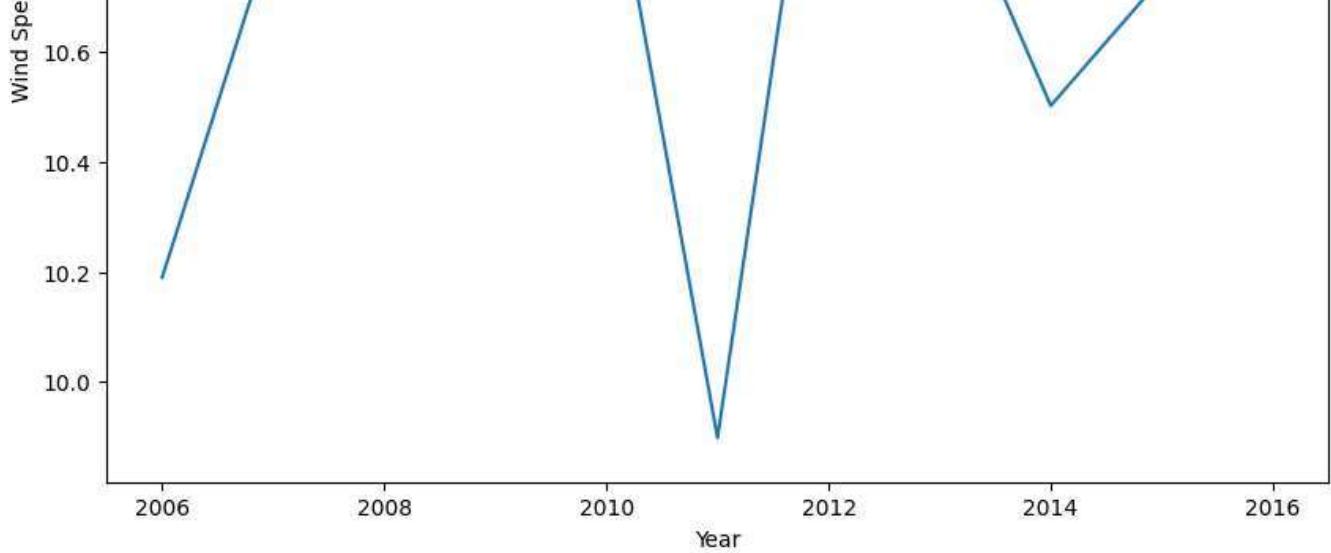
```
Out[ ]:      Wind Speed (km/h)
```

Year	Wind Speed (km/h)
2006	10.189852
2007	10.825392
2008	11.303897
2009	11.505948
2010	11.015628
2011	9.898262
2012	11.264545
2013	10.969389
2014	10.502473
2015	10.735247
2016	11.014142

```
In [ ]: fig,ax=plt.subplots(figsize=(10,8))
sns.lineplot(x=avg_wind_Speed.index,y=avg_wind_Speed["Wind Speed (km/h)"])
plt.title("Average wind speed over the yyears")
```

```
Out[ ]: Text(0.5, 1.0, 'Average wind speed over the yyears')
```





```
In [ ]: month_avg_wind_Speed=pd.DataFrame(df3.groupby("Month")["Wind Speed (km/h)"].mean())
order=["January", "February", "March", "April", "May", "June", "July", "August", "September",
       "October", "November", "December"]
monthly_wind_speed=month_avg_wind_Speed.reindex(index=order)
monthly_wind_speed
```

Out[]: Wind Speed (km/h)

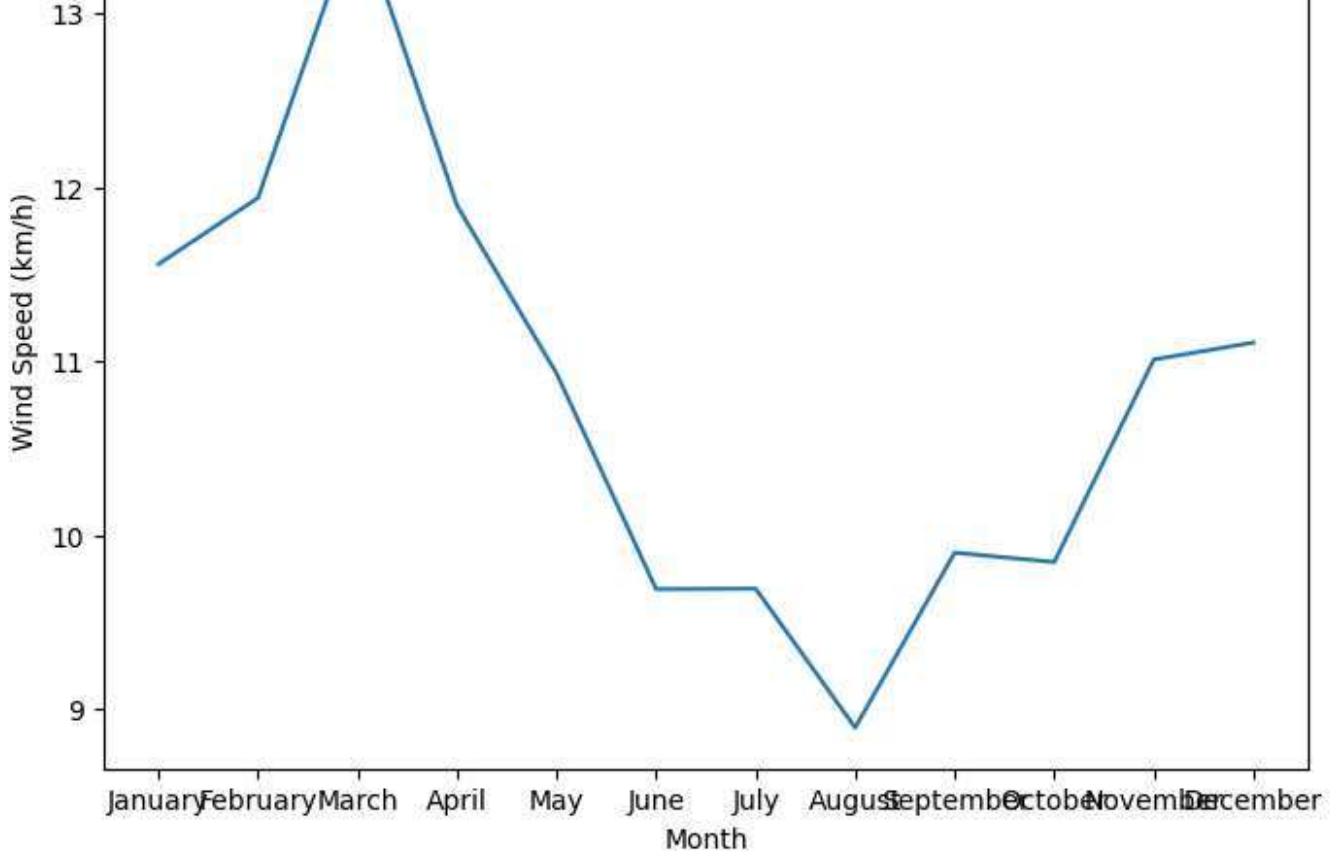
Month	
January	11.557404
February	11.938143
March	13.583733
April	11.893094
May	10.928663
June	9.689510
July	9.691968
August	8.893300
September	9.898669
October	9.844893
November	11.008592
December	11.106271

```
In [ ]: fig,ax=plt.subplots(figsize=(8,6))
sns.lineplot(x=monthly_wind_speed.index,y=monthly_wind_speed["Wind Speed (km/h)"])
plt.title("Monthly Average wind speed over the yeears")
```

Out[]: Text(0.5, 1.0, 'Monthly Average wind speed over the yeears')

Monthly Average wind speed over the yeears





```
In [ ]: df3["Summary"].value_counts()
```

```
Out[ ]: Partly Cloudy           29410
        Mostly Cloudy          25038
        Overcast                15437
        Clear                   10457
        Foggy                   6834
        Breezy and Mostly Cloudy  508
        Breezy and Overcast      486
        Breezy and Partly Cloudy 378
        Dry and Partly Cloudy   86
        Windy and Partly Cloudy 67
        Breezy                  53
        Windy and Overcast       45
        Breezy and Foggy         35
        Windy and Mostly Cloudy 35
        Dry                      34
        Humid and Mostly Cloudy 33
        Humid and Partly Cloudy 17
        Light Rain               17
        Dry and Mostly Cloudy   14
        Windy                   8
        Humid and Overcast       5
        Drizzle                 5
        Windy and Foggy          4
        Dangerously Windy and Partly Cloudy 1
        Windy and Dry             1
        Breezy and Dry            1
        Name: Summary, dtype: int64
```

```
In [ ]: weather_cond=pd.DataFrame(df3.groupby("Year")["Summary"].describe(include="O").top)
        weather_cond.rename(columns={"top":"most frequent weather"})
```

```
Out[ ]: most frequent weather
```

Year	
2006	Partly Cloudy
2007	Partly Cloudy
2008	Partly Cloudy
2009	Partly Cloudy
2010	Partly Cloudy
2011	Partly Cloudy
2012	Partly Cloudy
2013	Partly Cloudy
2014	Mostly Cloudy
2015	Partly Cloudy
2016	Partly Cloudy

```
In [ ]: m_weather_cond=pd.DataFrame(df3.groupby("Month")["Summary"].describe(include="0").top)
order=["January", "February", "March", "April", "May", "June", "July", "August", "September",
       "October", "November", "December"]
m_weather_cond.rename(columns={"top":"most frequent weather"})
monthly=m_weather_cond.reindex(index=order)
monthly
```

Out[]:

Month	
January	Overcast
February	Overcast
March	Partly Cloudy
April	Partly Cloudy
May	Partly Cloudy
June	Partly Cloudy
July	Partly Cloudy
August	Partly Cloudy
September	Partly Cloudy
October	Partly Cloudy
November	Mostly Cloudy
December	Overcast

```
In [ ]: month_avg_visibility=pd.DataFrame(df3.groupby("Month")["Visibility (km)"].mean())
order=["January", "February", "March", "April", "May", "June", "July", "August", "September",
       "October", "November", "December"]
monthly_visibility=month_avg_visibility.reindex(index=order)
monthly_visibility
```

Out[]:

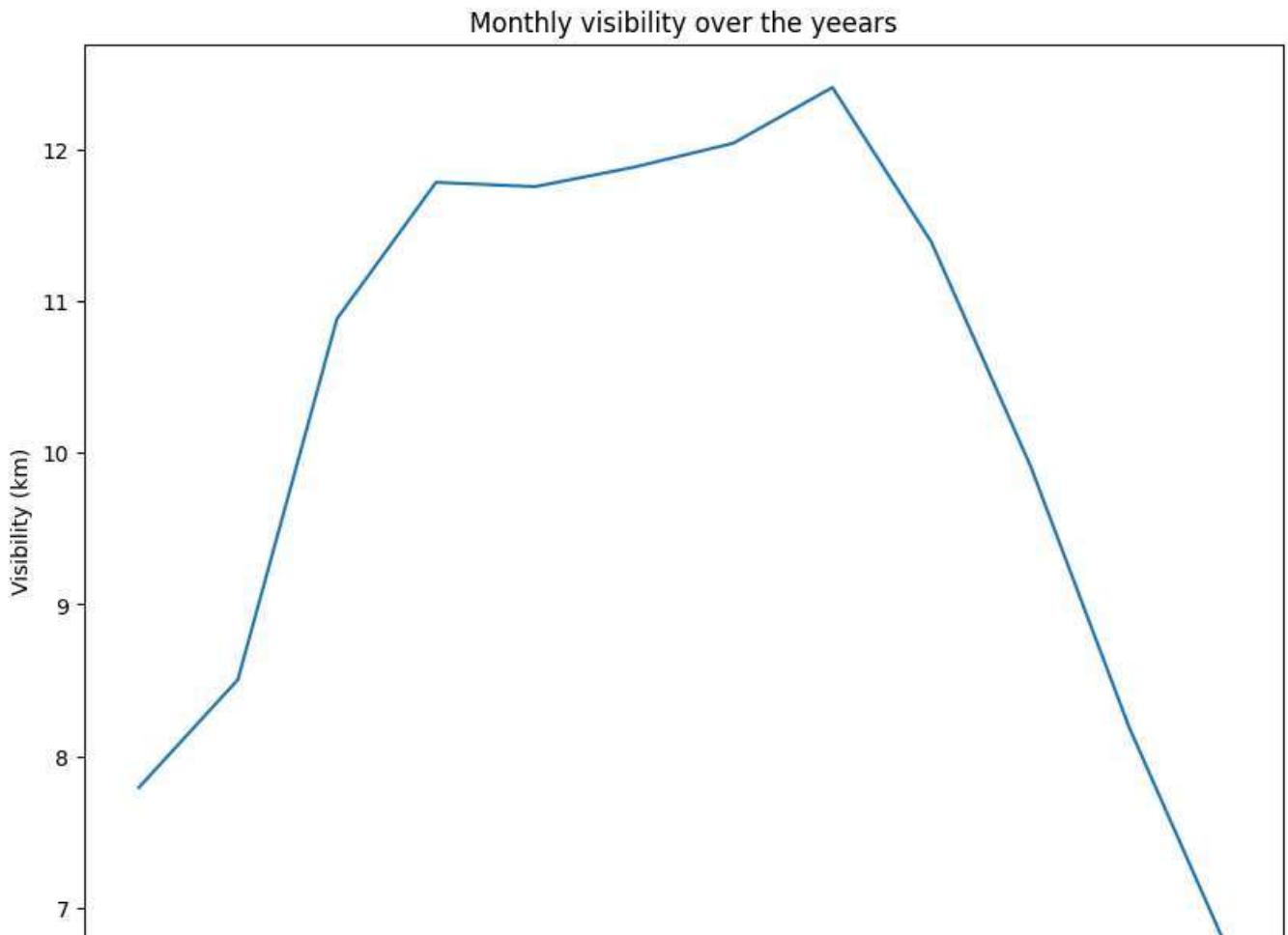
Visibility (km)

Month	Visibility (km)
January	7.791621
February	8.503892
March	10.884571
April	11.784224
May	11.756021
June	11.884640
July	12.042516
August	12.410527
September	11.389066
October	9.911756
November	8.181631
December	6.704159

In []:

```
fig,ax=plt.subplots(figsize=(10,8))
sns.lineplot(x=monthly_visibility.index,y=monthly_visibility["Visibility (km)"])
plt.title("Monthly visibility over the yyears")
```

Out[]: Text(0.5, 1.0, 'Monthly visibility over the yyears')



```
January February March April May June July August September October November December
```

Month

```
In [ ]: percip=pd.DataFrame(df3.groupby("Month")["Precip Type"].describe(include="0").top)
order=["January","February","March","April","May","June","July","August","September",
       "October","November","December"]
m_p=percip.rename(columns={"top":"Precip Type"})
monthly_percip=m_p.reindex(index=order)
monthly_percip
```

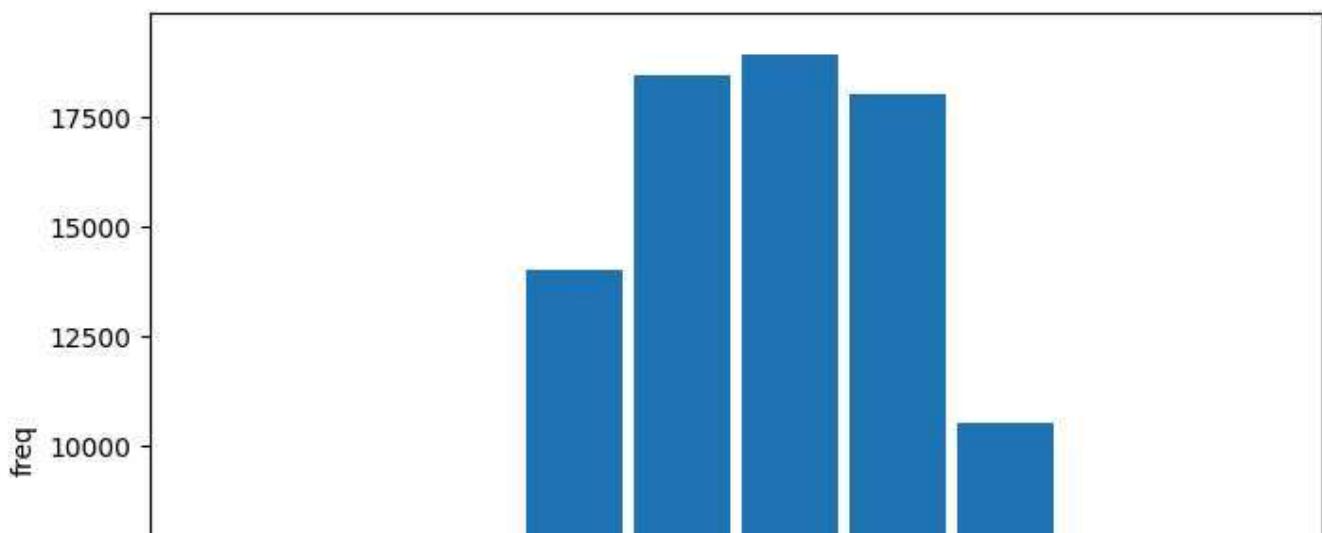
Out[]:

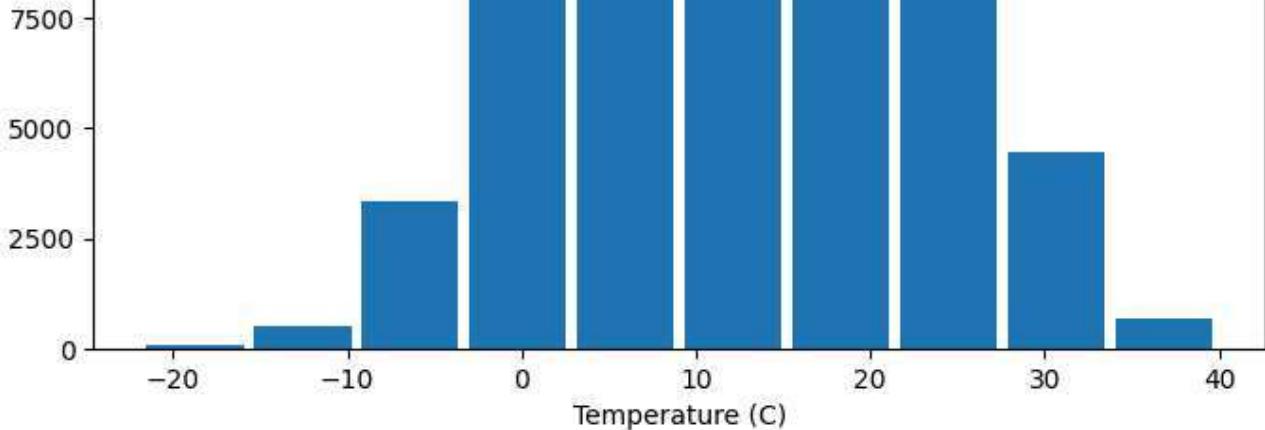
Precip Type

Month	Precip Type
January	rain
February	rain
March	rain
April	rain
May	rain
June	rain
July	rain
August	rain
September	rain
October	rain
November	rain
December	rain

```
In [ ]: fig,ax=plt.subplots(figsize=(8,6))
plt.hist(df3["Temperature (C)"],bins=10,rwidth=0.9)
plt.xlabel("Temperature (C)")
plt.ylabel("freq")
```

Out[]: Text(0, 0.5, 'freq')





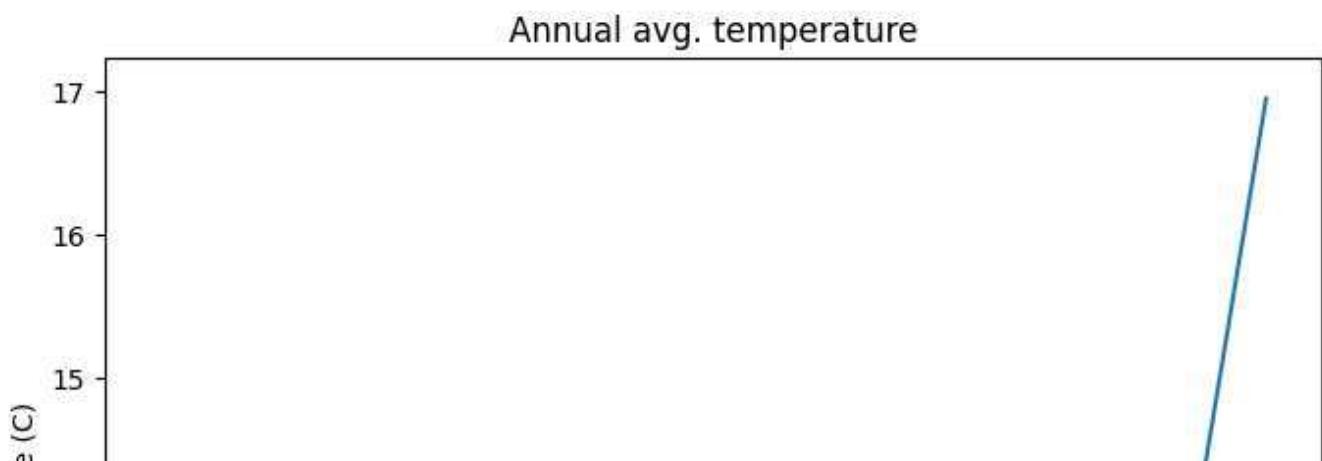
```
In [ ]: year_avg_temp=pd.DataFrame(df3.groupby("Year")["Temperature (C)"].mean())
year_avg_temp
```

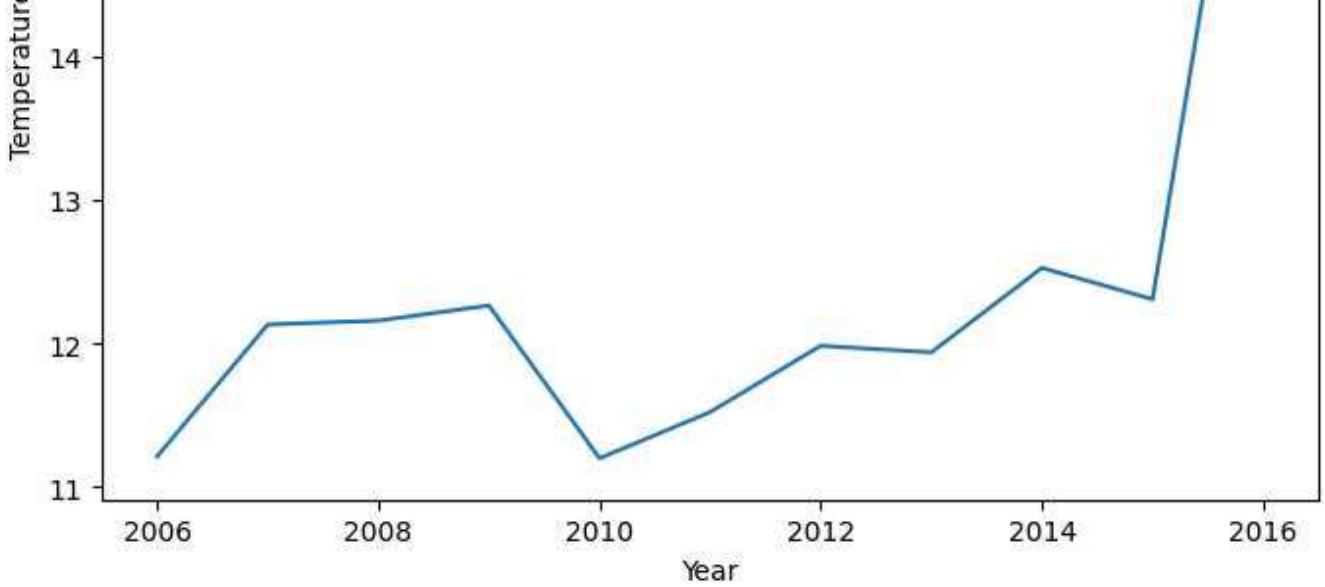
```
Out[ ]:    Temperature (C)
```

Year	Temperature (C)
2006	11.215365
2007	12.135239
2008	12.161876
2009	12.267910
2010	11.202061
2011	11.524453
2012	11.986726
2013	11.940719
2014	12.529737
2015	12.311370
2016	16.947973

```
In [ ]: fig,ax=plt.subplots(figsize=(8,6))
sns.lineplot(x=year_avg_temp.index,y=year_avg_temp["Temperature (C)"])
plt.title("Annual avg. temperature")
```

```
Out[ ]: Text(0.5, 1.0, 'Annual avg. temperature')
```





```
In [ ]: month_temp=pd.DataFrame(df3.groupby("Month")["Temperature (C)"].mean())
order=["January","February","March","April","May","June","July","August","September",
       "October","November","December"]

monthly_avg_temp=month_temp.reindex(index=order)
monthly_avg_temp
```

Out[]:

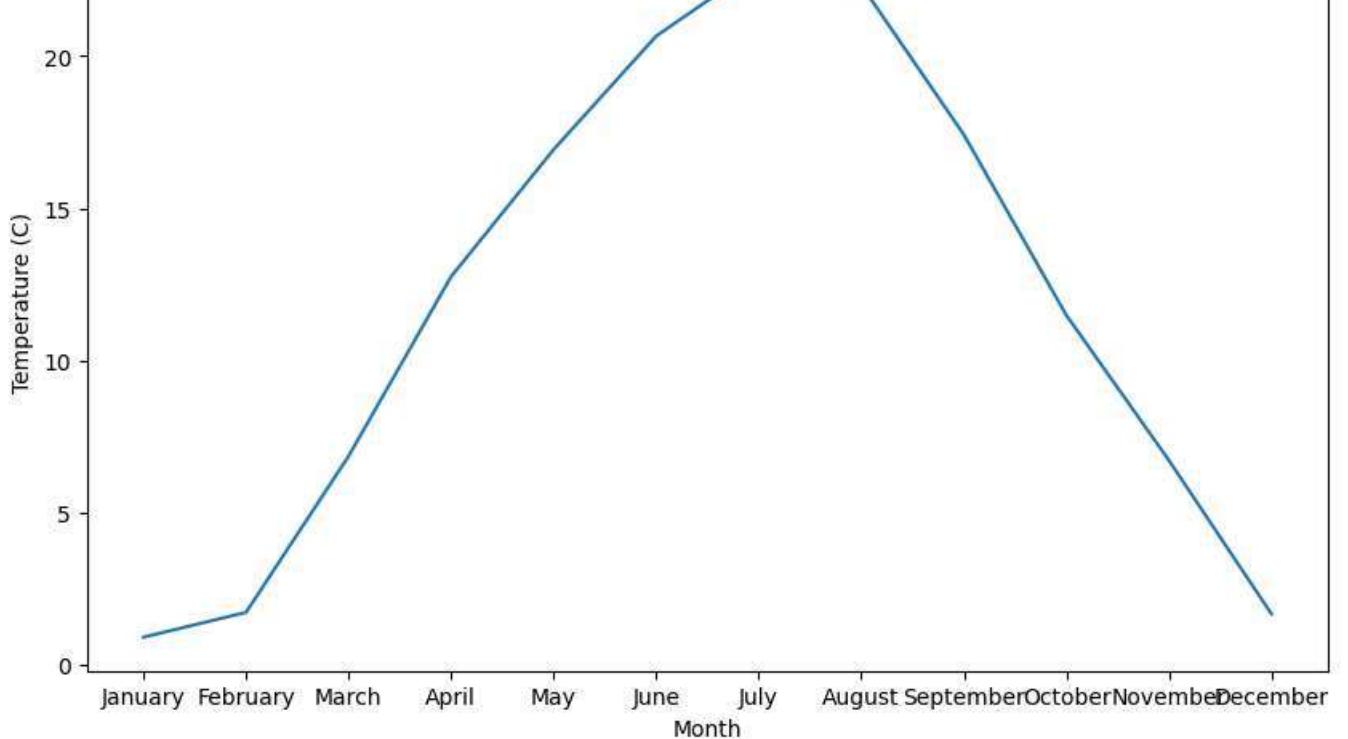
Temperature (C)

Month	
January	0.907546
February	1.722785
March	6.854623
April	12.756417
May	16.930651
June	20.657390
July	22.982112
August	22.334834
September	17.420623
October	11.486013
November	6.719990
December	1.673153

```
In [ ]: fig,ax=plt.subplots(figsize=(10,6))
sns.lineplot(x=monthly_avg_temp.index,y=monthly_avg_temp["Temperature (C)"])
plt.title("monthly avg. temperature")
```

Out[]: Text(0.5, 1.0, 'monthly avg. temperature')

monthly avg. temperature



```
In [ ]: df4=df3.drop(columns=["Year", "day", "Loud Cover"])
d_corr=df4.corr()
d_corr
```

<ipython-input-38-e8330dfc59bc>:2: FutureWarning: The default value of numeric_only in DataFrame.corr is deprecated. In a future version, it will default to False. Select only valid columns or specify the value of numeric_only to silence this warning.
d_corr=df4.corr()

	Temperature (C)	Apparent Temperature (C)	Humidity	Wind Speed (km/h)	Wind Bearing (degrees)	Visibility (km)	Pressure (millibars)
Temperature (C)	1.000000	0.992491	-0.630688	0.012145	0.029357	0.387268	-0.004241
Apparent Temperature (C)	0.992491	1.000000	-0.600427	-0.054672	0.027802	0.375889	0.001281
Humidity	-0.630688	-0.600427	1.000000	-0.228042	0.000196	-0.364369	0.004198
Wind Speed (km/h)	0.012145	-0.054672	-0.228042	1.000000	0.108131	0.107905	-0.049951
Wind Bearing (degrees)	0.029357	0.027802	0.000196	0.108131	1.000000	0.052318	-0.011925
Visibility (km)	0.387268	0.375889	-0.364369	0.107905	0.052318	1.000000	0.061838
Pressure (millibars)	-0.004241	0.001281	0.004198	-0.049951	-0.011925	0.061838	1.000000

```
In [ ]: fig,ax=plt.subplots(figsize=(10,8))
sns.heatmap(d_corr,annot=True,cmap='magma_r',linewidhts=0.2)
plt.title("correlations heat map")
```

Out[]: Text(0.5, 1.0, 'correlations heat map')

