Problem Statement- Visualize the data using Python libraries matplotlib,seaborn by plotting the graphs for assignment no. 2 and 3

Importing libraries and reading dataset

```python
In [181...   import pandas as pd
            import numpy as np
            import seaborn
            import matplotlib.pyplot as plt
```

```python
In [182...   data=pd.read_csv("airquality.csv")
            data
```

Out[182]:

	Unnamed: 0	Ozone	Solar.R	Wind	Temp	Month	Day	Humidity
0	1	41.0	190.0	7.4	67	5	1	High
1	2	36.0	118.0	8.0	72	5	2	High
2	3	12.0	149.0	12.6	74	5	3	Medium
3	4	18.0	313.0	11.5	62	5	4	Medium
4	5	NaN	NaN	14.3	56	5	5	NaN
...	...	...	...	...	...	...	...	...
148	149	30.0	193.0	6.9	70	9	26	Low
149	150	NaN	145.0	13.2	77	9	27	Low
150	151	14.0	191.0	14.3	75	9	28	High
151	152	18.0	131.0	8.0	76	9	29	High
152	153	20.0	223.0	11.5	68	9	30	Medium

153 rows × 8 columns

Sum of null values in each column

```python
In [183...   data.isnull().sum()
```

```
Out[183]:  Unnamed: 0      0
           Ozone          37
           Solar.R         7
           Wind            0
           Temp            0
           Month           0
           Day             0
           Humidity        6
           dtype: int64
```

Replacing null values

```
In [184…  data["Ozone"] = data["Ozone"].fillna(data["Ozone"].mean())
          data["Solar.R"] = data["Solar.R"].fillna(data["Solar.R"].mean())
          data
```

Out[184]:

	Unnamed: 0	Ozone	Solar.R	Wind	Temp	Month	Day	Humidity
0	1	41.00000	190.000000	7.4	67	5	1	High
1	2	36.00000	118.000000	8.0	72	5	2	High
2	3	12.00000	149.000000	12.6	74	5	3	Medium
3	4	18.00000	313.000000	11.5	62	5	4	Medium
4	5	42.12931	185.931507	14.3	56	5	5	NaN
...	...	...	...	...	...	...	...	...
148	149	30.00000	193.000000	6.9	70	9	26	Low
149	150	42.12931	145.000000	13.2	77	9	27	Low
150	151	14.00000	191.000000	14.3	75	9	28	High
151	152	18.00000	131.000000	8.0	76	9	29	High
152	153	20.00000	223.000000	11.5	68	9	30	Medium

153 rows × 8 columns

Sum of null values in each column

```
In [185…  data.isnull().sum()
```

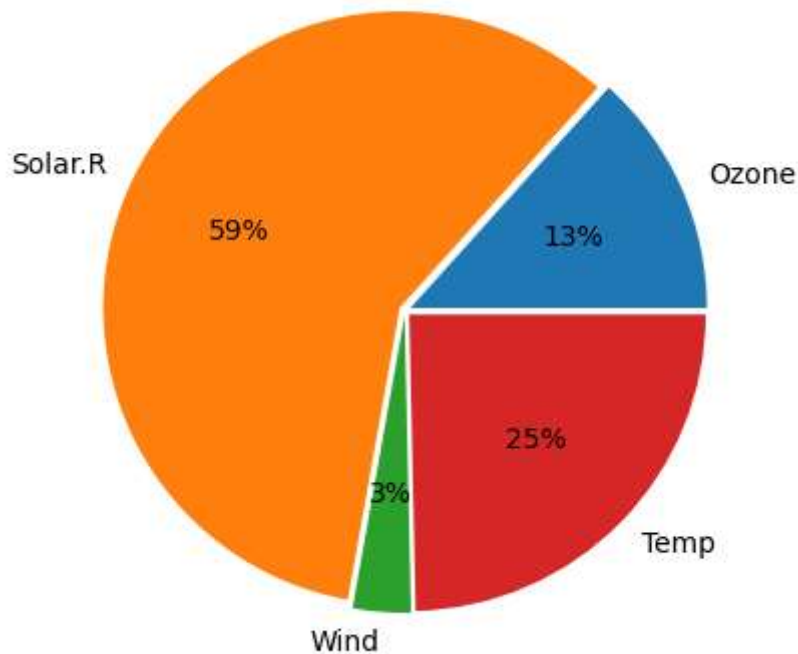```
Out[185]: Unnamed: 0   0
           Ozone   0
           Solar.R   0
           Wind   0
           Temp   0
           Month   0
           Day   0
           Humidity   6
           dtype: int64
```

# Ploting a PieChart

```
In [186…   plt.pie([data["Ozone"].mean(),data["Solar.R"].mean(),data["Wind"].mean(),data["Temp
           plt.plot()
```

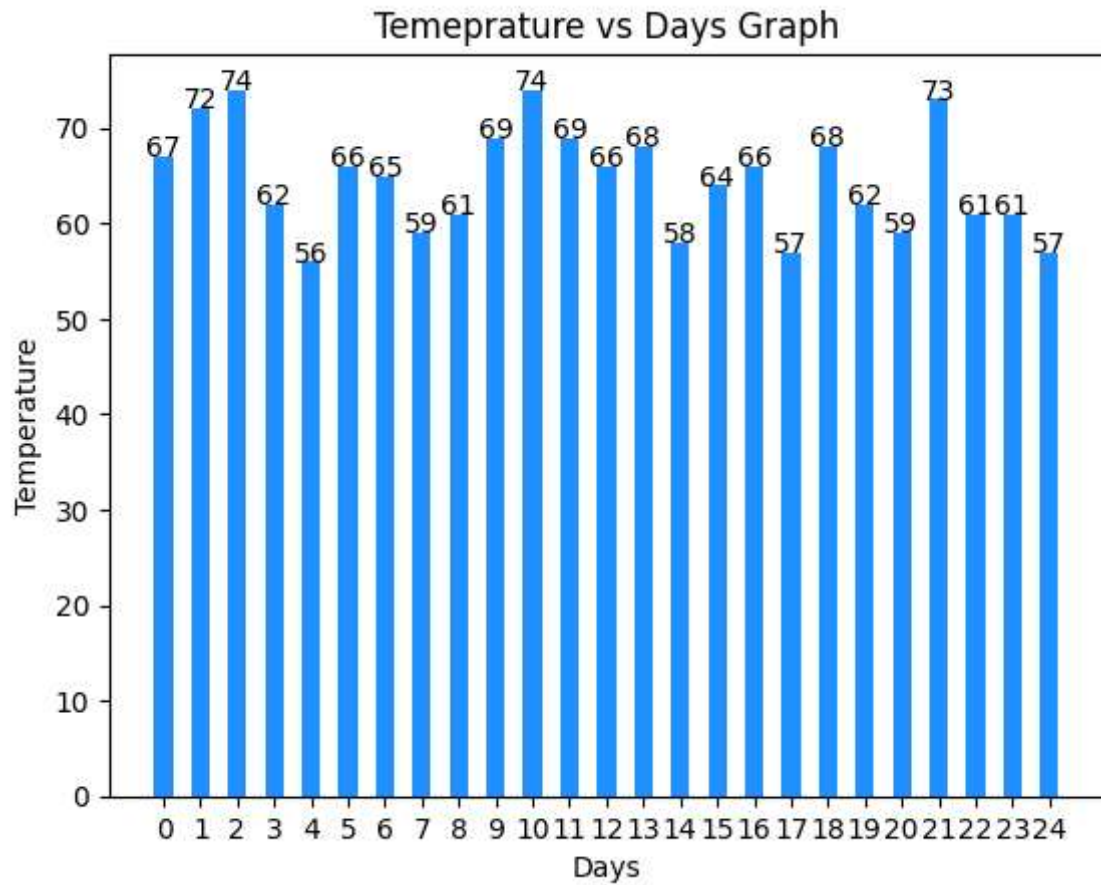Out[186]: []

Ploting a Bar Graph

```
In [187…   y = np.arange(len(data.iloc[0:25, 4]))

           # add labels
           for i in range(len(data.iloc[0:25, 4])):
                   plt.text(i, data.iloc[0:25, 4][i], data.iloc[0:25, 4][i], ha = 'center')

           plt.bar(x=y, height=data.iloc[0:25, 4], tick_label=y,width=0.5, color="dodgerblue")
           plt.title("Temeprature vs Days Graph")
           plt.xlabel("Days")
           plt.ylabel("Temperature")
           plt.show()
```
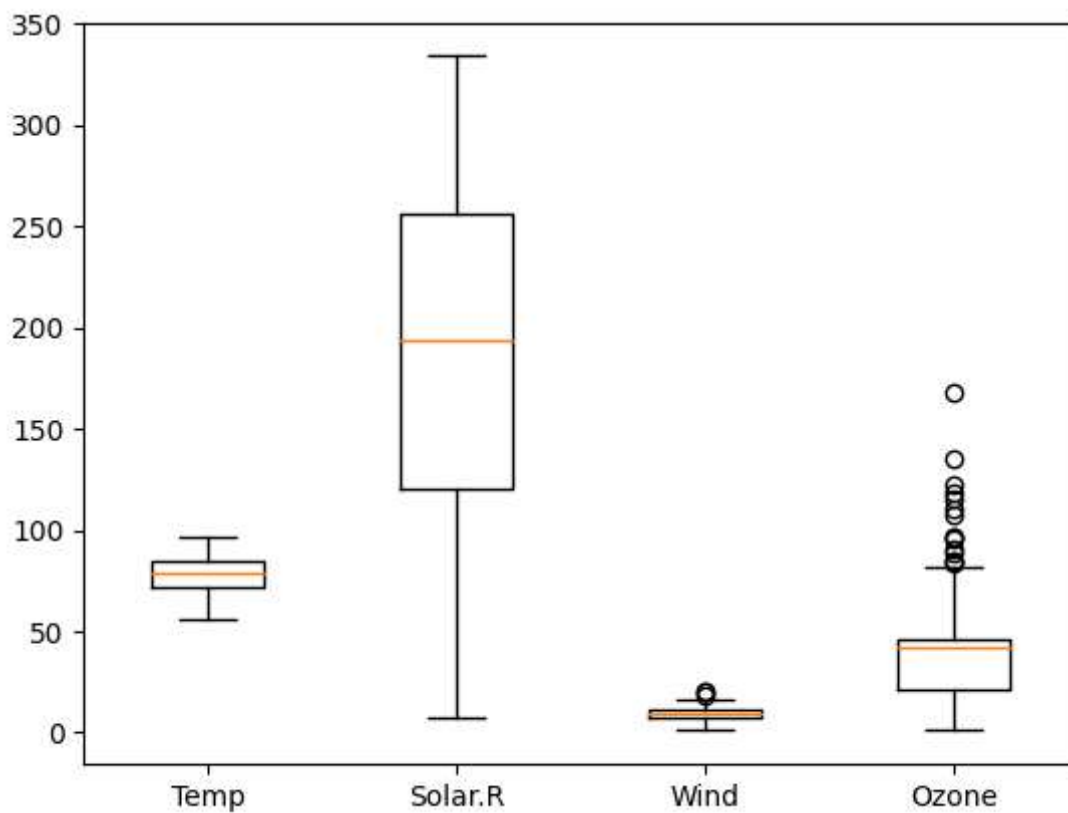
Ploting a BoxPlot

In [188...
```python
plt.boxplot(x=data[["Temp", "Solar.R", "Wind", "Ozone"]], labels=["Temp", "Solar.R"
plt.plot()
```
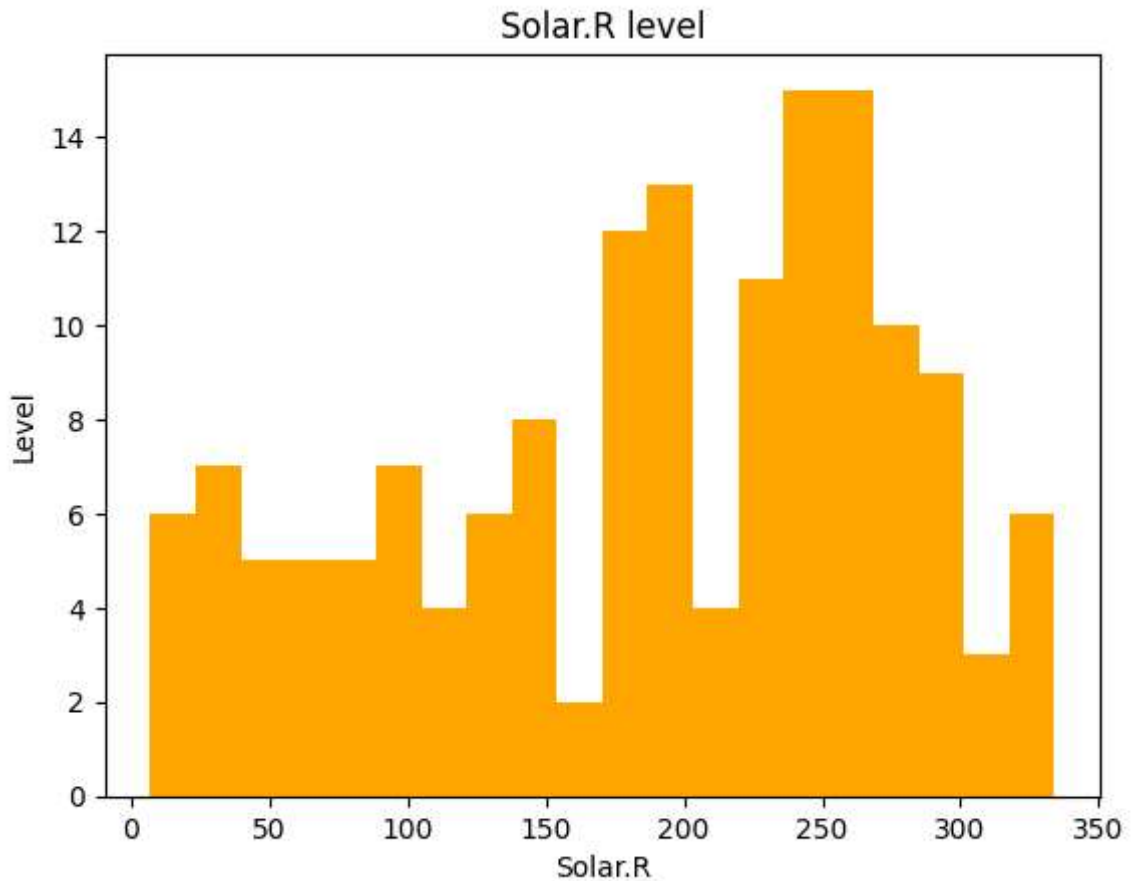
Out[188]: []

Ploting a graph for Solar Radiation level

```
In [189…   plt.hist(x=data["Solar.R"], bins=20, color="orange", label="Solar.R")
           plt.title("Solar.R level")
           plt.xlabel("Solar.R")
           plt.ylabel("Level")
           plt.plot()
```
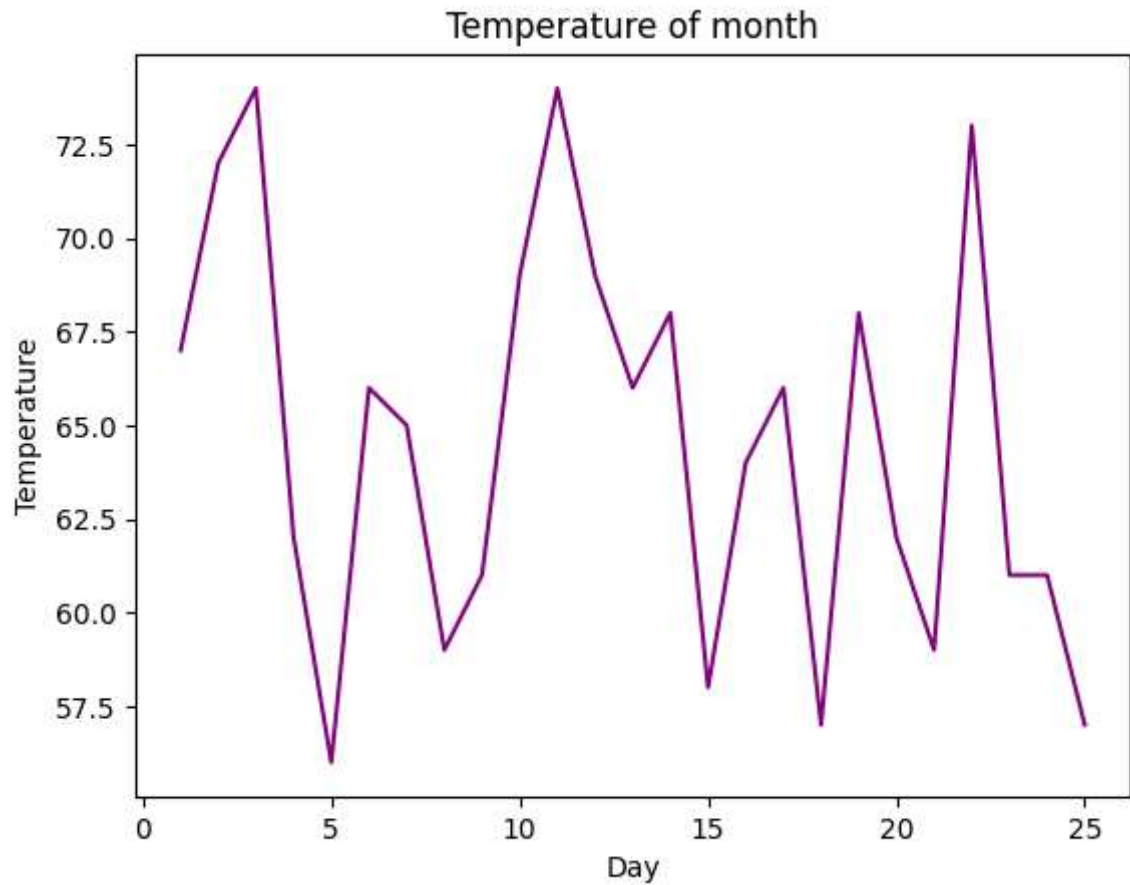
Out[189]:   []

Ploting a graph for Temperature of month

```
In [190…   plt.plot(data.iloc[0:25, 6], data.iloc[0:25, 4], color="purple")
           plt.title("Temperature of month")
           plt.xlabel("Day")
           plt.ylabel("Temperature")
```
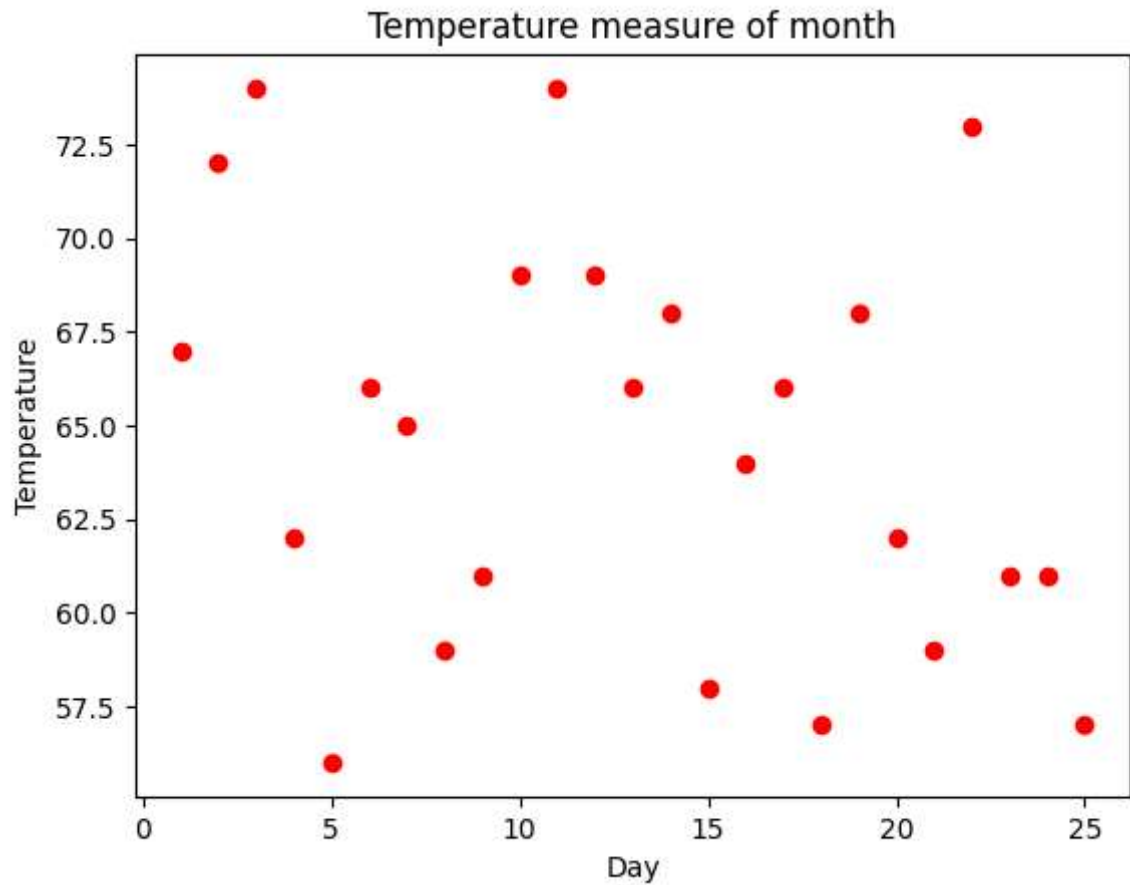
Out[190]:   Text(0, 0.5, 'Temperature')

Temperature of month

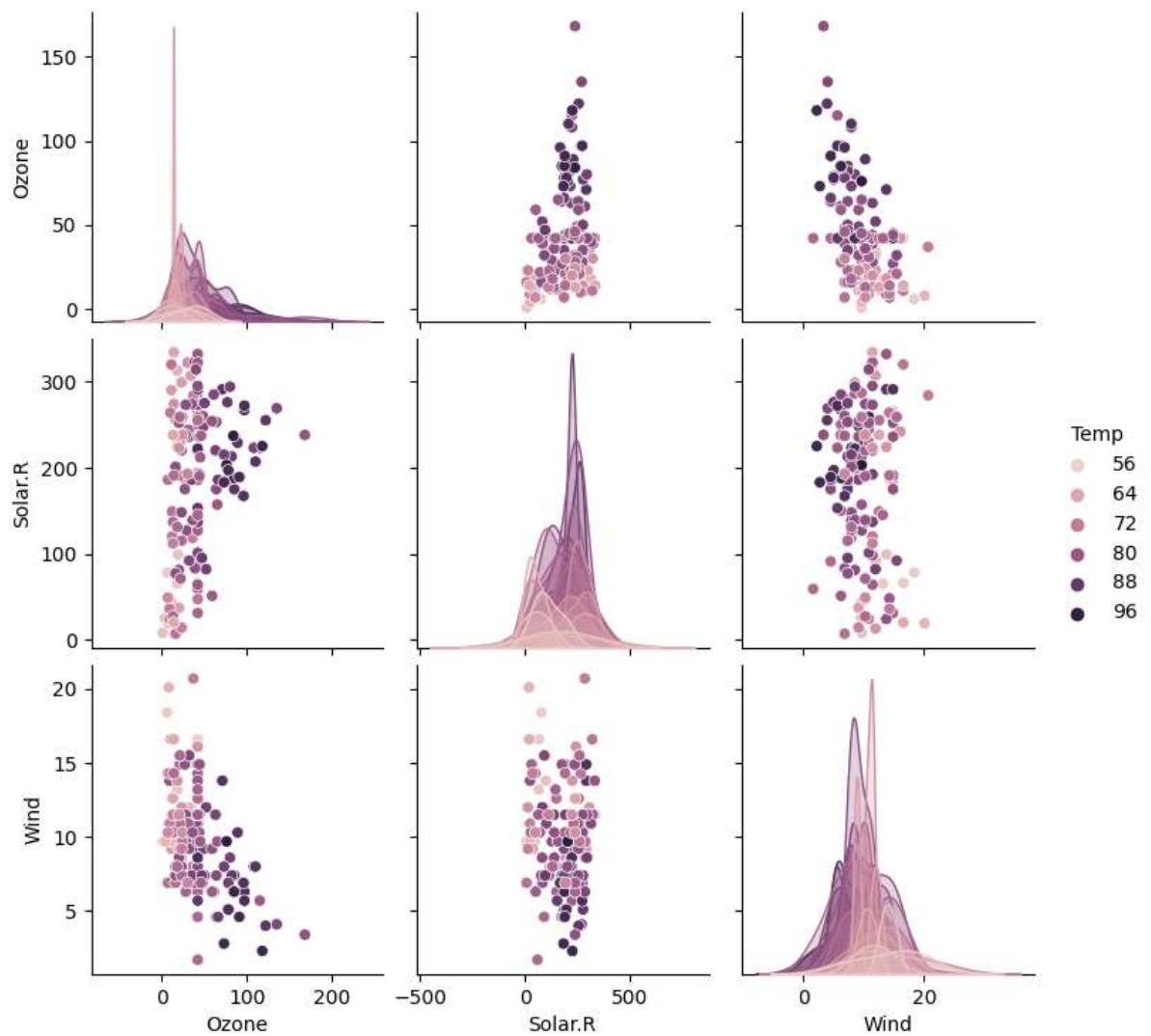# Ploting a scatter plot for Temperature of month

```
In [191… plt.scatter(data.iloc[0:25, 6], data.iloc[0:25, 4], color="red")
         plt.title("Temperature measure of month")
         plt.xlabel("Day")
         plt.ylabel("Temperature")
```

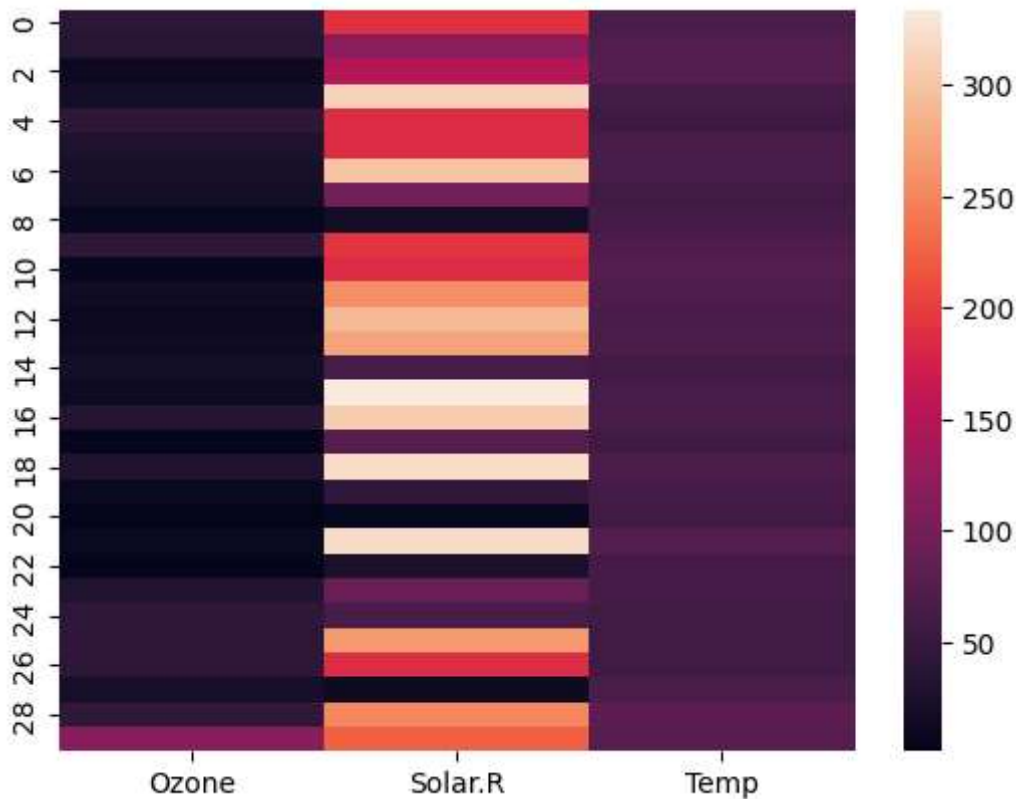Out[191]: Text(0, 0.5, 'Temperature')

Temperature measure of month

Ploting a pair-plot

```
In [192…   seaborn.pairplot(data.iloc[:,[1,2,3,4,7]], hue ='Temp')
           plt.show()
```

Ploting a Heat-Map

```
In [193…  seaborn.heatmap(data=data.iloc[0:30,[1,2,4]])
          plt.show()
```

Visualizing text using wordcloud

```
In [194...    from wordcloud import WordCloud, STOPWORDS

             stopwords = set(STOPWORDS)
             text = """Lorem Ipsum is simply dummy text of the printing and typesetting industry
             Lorem Ipsum has been the industry's standard dummy text ever since the 1500s,
             when an unknown printer took a galley of type and scrambled it to make a type speci
             It has survived not only five centuries, but also the leap into electronic typesett
             It was popularised in the 1960s with the release of Letraset sheets containing Lore
             and more recently with desktop publishing software like Aldus PageMaker including v

             wordcloud = WordCloud(width = 800, height = 800,
                         background_color ='white',
                         stopwords = stopwords,
                         min_font_size = 10).generate(text)

             plt.imshow(wordcloud)
             plt.axis("off")
             plt.tight_layout(pad = 0)
             plt.show()
```