Group B Assignment - 4

Visualize the data using Python libraries matplotlib , seaborn by plotting the graphs for assignment no 2 and 3 (Group B)

Importing libraries

In [2]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
```

Reading the data

In [3]:
```
data = pd.read_csv("airquality (1).csv")
```

In [4]:
```
data
```

Out[4]:

	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	Low
3	4	18.0	313.0	11.5	62	5	4	NaN
4	5	NaN	NaN	14.3	56	5	5	High
...	...	...	...	...	...	...	...	...
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	Medium
152	153	20.0	223.0	11.5	68	9	30	Low

153 rows × 8 columns

Replacing the null values by mean values

In [5]:
```
data["Ozone"] = data["Ozone"].fillna(data["Ozone"].mean())
data["Solar.R"] = data["Solar.R"].fillna(data["Solar.R"].mean())
```

In [6]:
```
data
```

Out[6]:

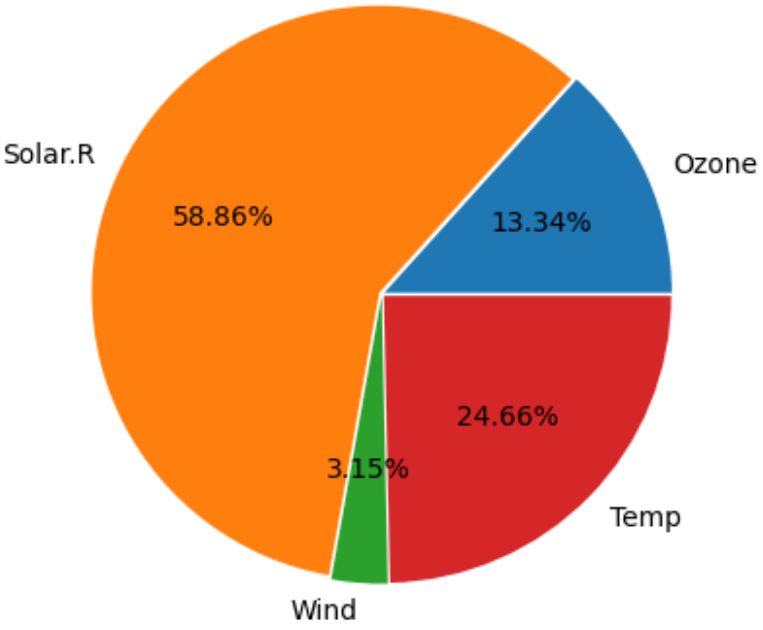
	Unnamed: 0	Ozone	Solar.R	Wind	Temp	Month	Day	Humidity
0	1	41.00000	190.000000	7.4	67	5	1	High
1	2	36.00000	118.000000	8.0	72	5	2	High
2	3	12.00000	149.000000	12.6	74	5	3	Low
3	4	18.00000	313.000000	11.5	62	5	4	NaN
4	5	42.12931	185.931507	14.3	56	5	5	High
...	...	...	...	...	...	...	...	...
148	149	30.00000	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	Medium
152	153	20.00000	223.000000	11.5	68	9	30	Low

153 rows × 8 columns

Pie Chart

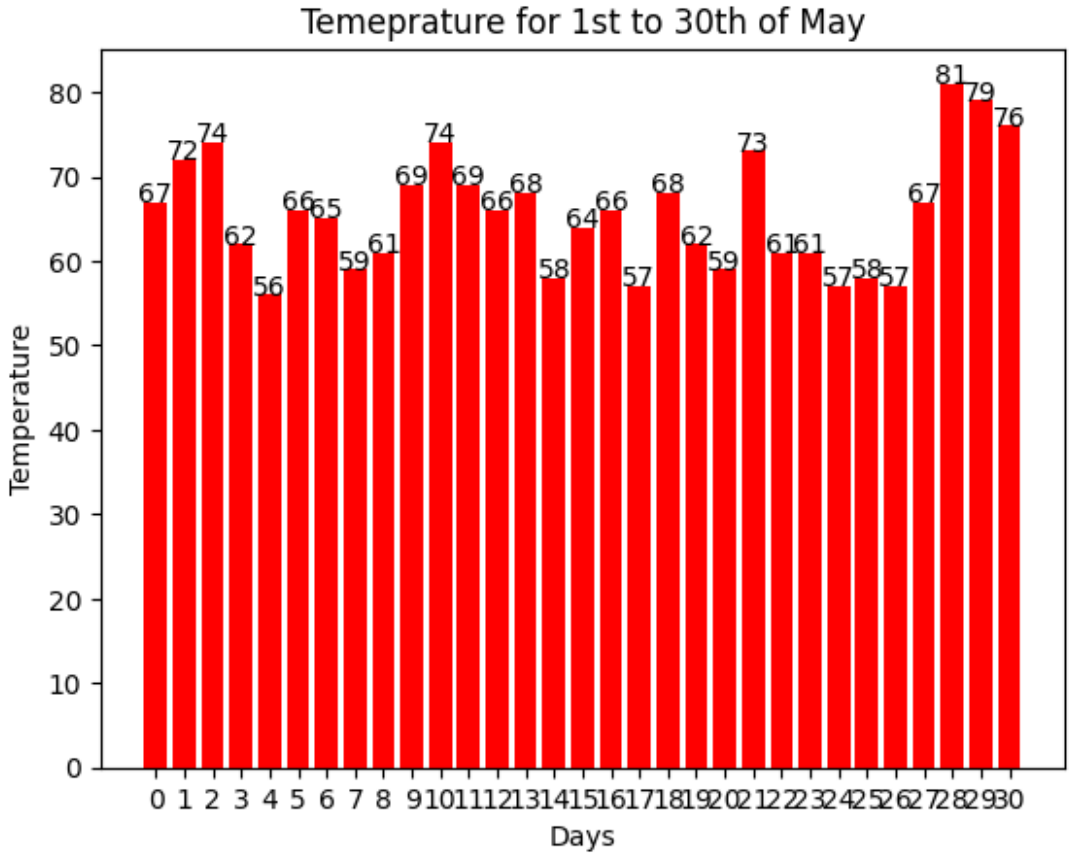
In [7]:
```
plt.pie([data["Ozone"].mean(),data["Solar.R"].mean(),data["Wind"].mean(),data[
plt.plot()
```

Out[7]:
```
[]
```



Bar Plot

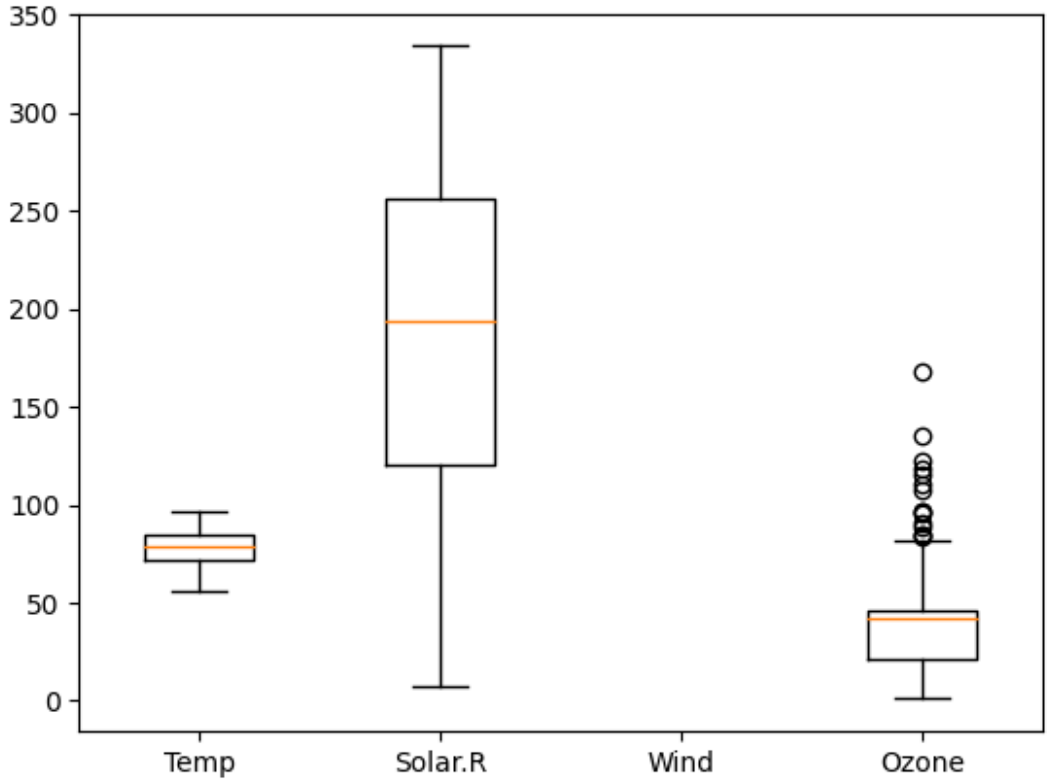
In [8]:
```
def addlabels():
    for i in range(len(data.iloc[0:31, 4])):
        plt.text(i, data.iloc[0:31, 4][i], data.iloc[0:31, 4][i], ha = 'center

y = np.arange(len(data.iloc[0:31, 4]))
addlabels()
plt.bar(x=y, height=data.iloc[0:31, 4], tick_label=y, color="red")
plt.title("Temeprature for 1st to 30th of May")
plt.xlabel("Days")
plt.ylabel("Temperature")
plt.show()
```



Box Plot

In [26]:
```
plt.boxplot(x=data[["Temp", "Solar.R", "Wind", "Ozone"]], labels=["Temp", "Sol
plt.plot()
```

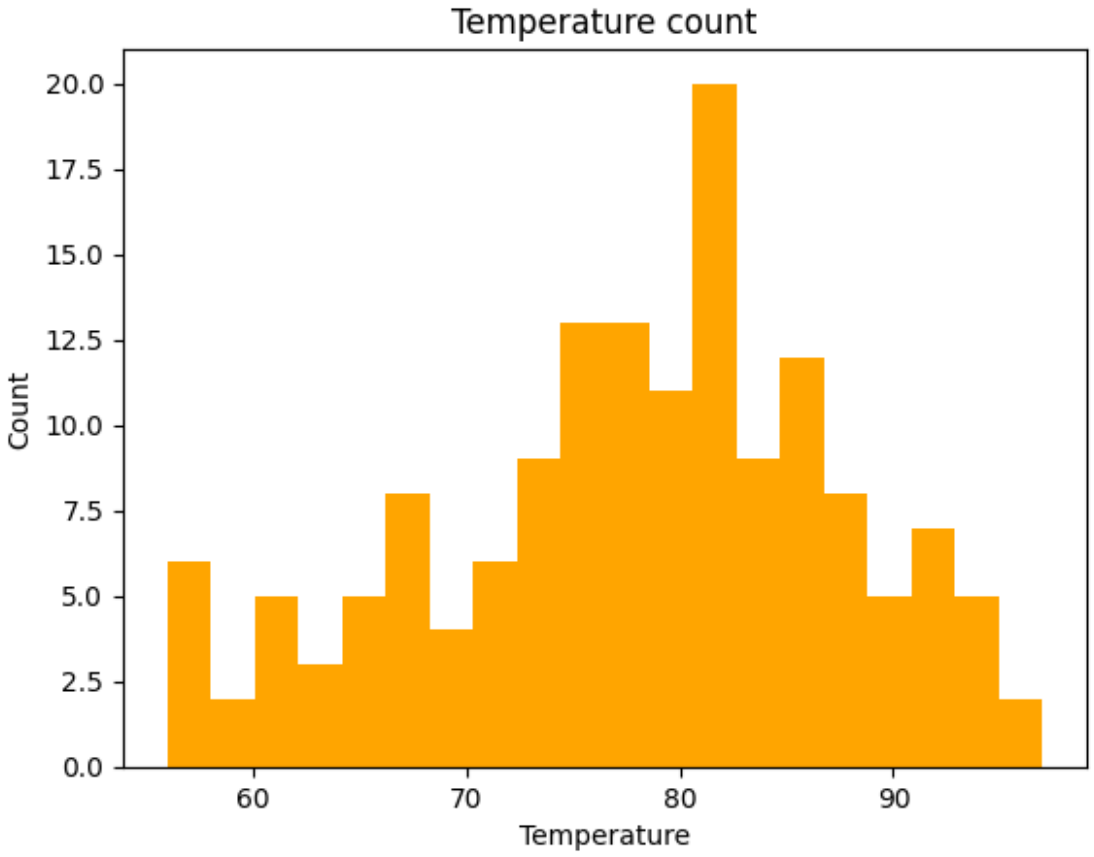
Out[26]: []



Histogram

In [27]:
```
plt.hist(x=data["Temp"], bins=20, color="orange", label="Temp")
plt.title("Temperature count")
plt.xlabel("Temperature")
plt.ylabel("Count")
plt.plot()
```

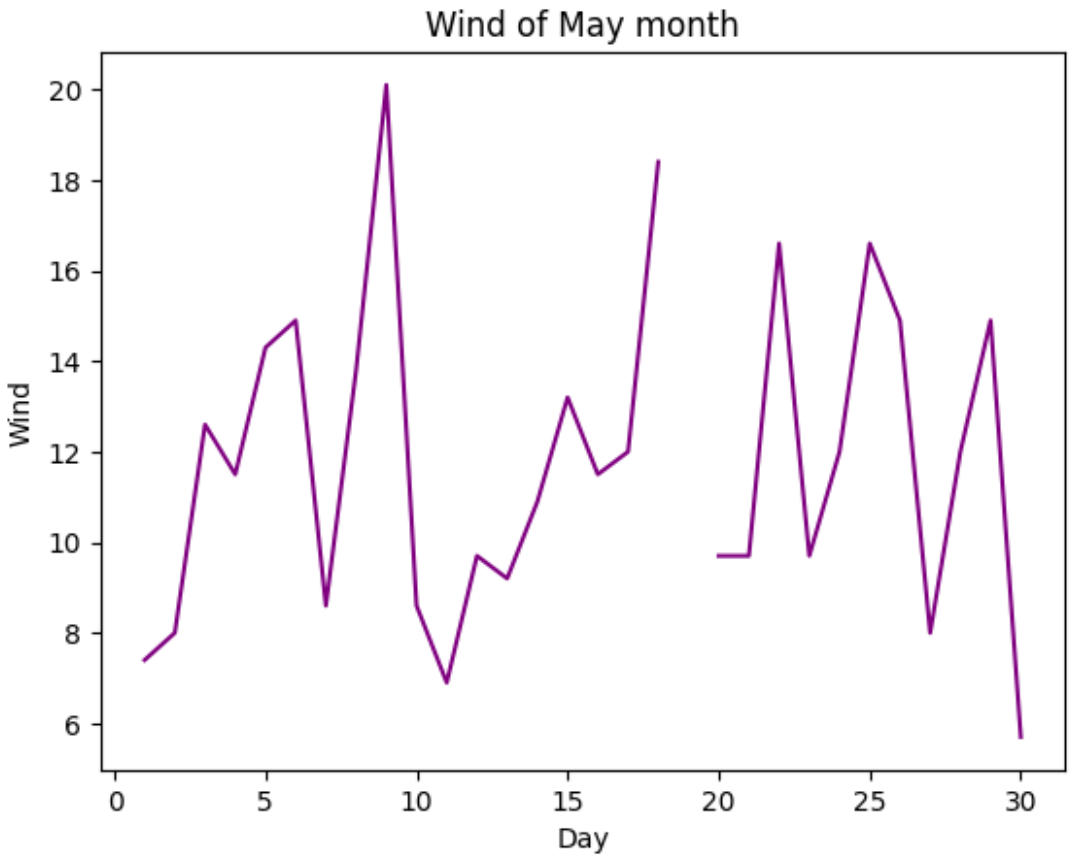
Out[27]:
```
[]
```



Line Graph

In [28]:
```
plt.plot(data.iloc[0:30, 6], data.iloc[0:30, 3], color="purple")
plt.title("Wind of May month")
plt.xlabel("Day")
plt.ylabel("Wind")
```

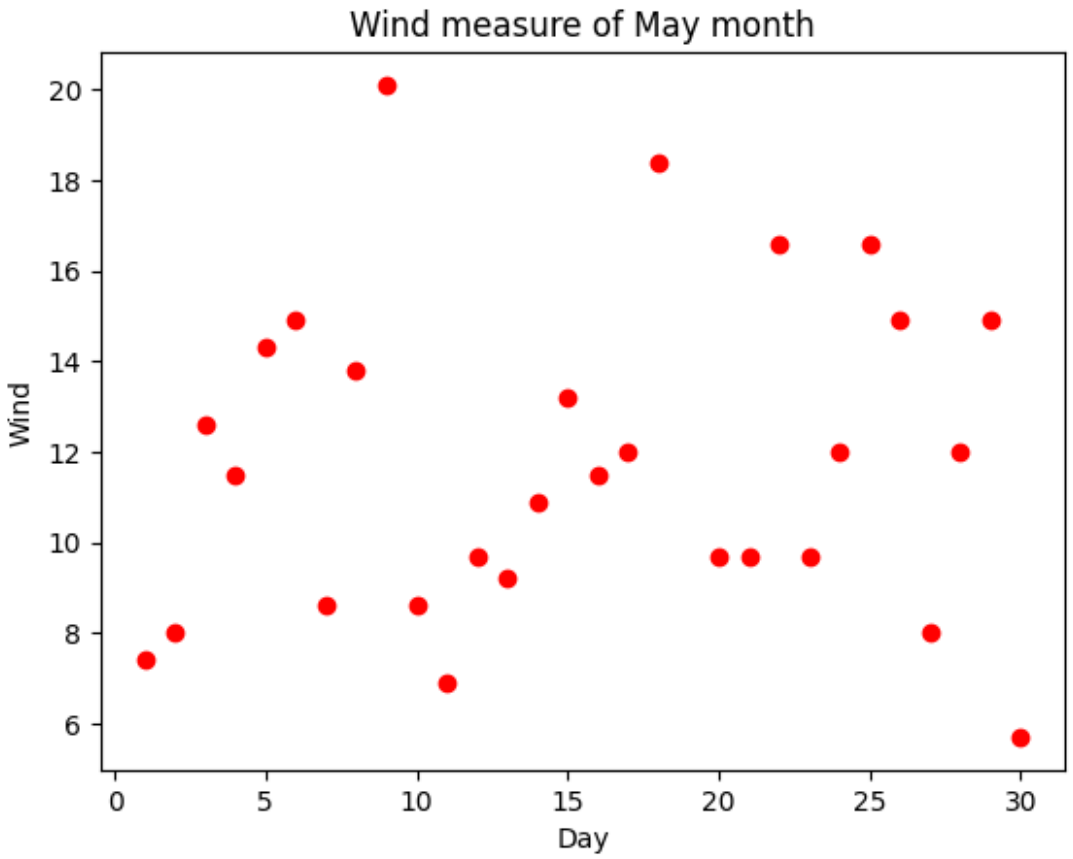
Out[28]:
```
Text(0, 0.5, 'Wind')
```



Scatter Plot

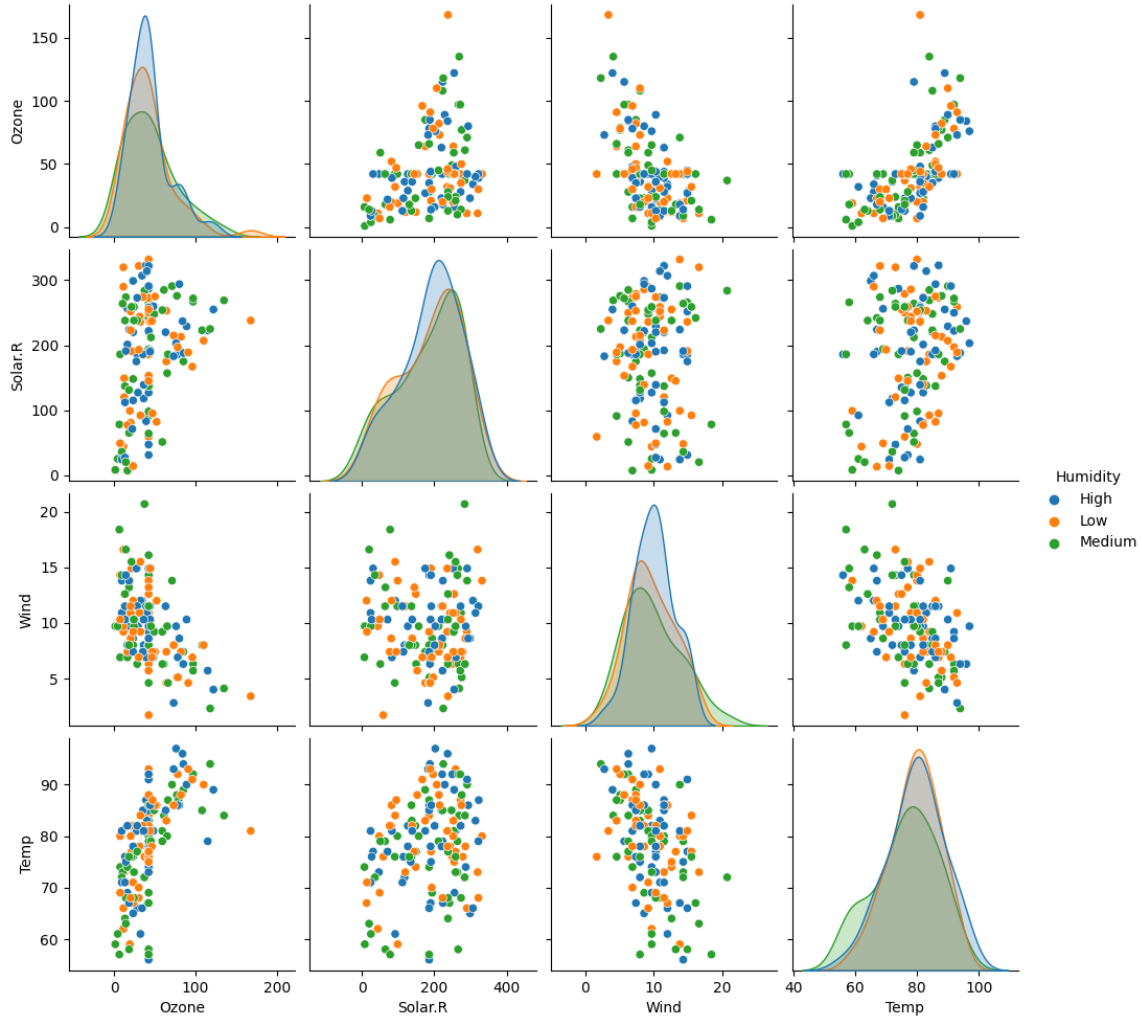
In [29]:
```
plt.scatter(data.iloc[0:30, 6], data.iloc[0:30, 3], color="red")
plt.title("Wind measure of May month")
plt.xlabel("Day")
plt.ylabel("Wind")
```

Out[29]:
```
Text(0, 0.5, 'Wind')
```



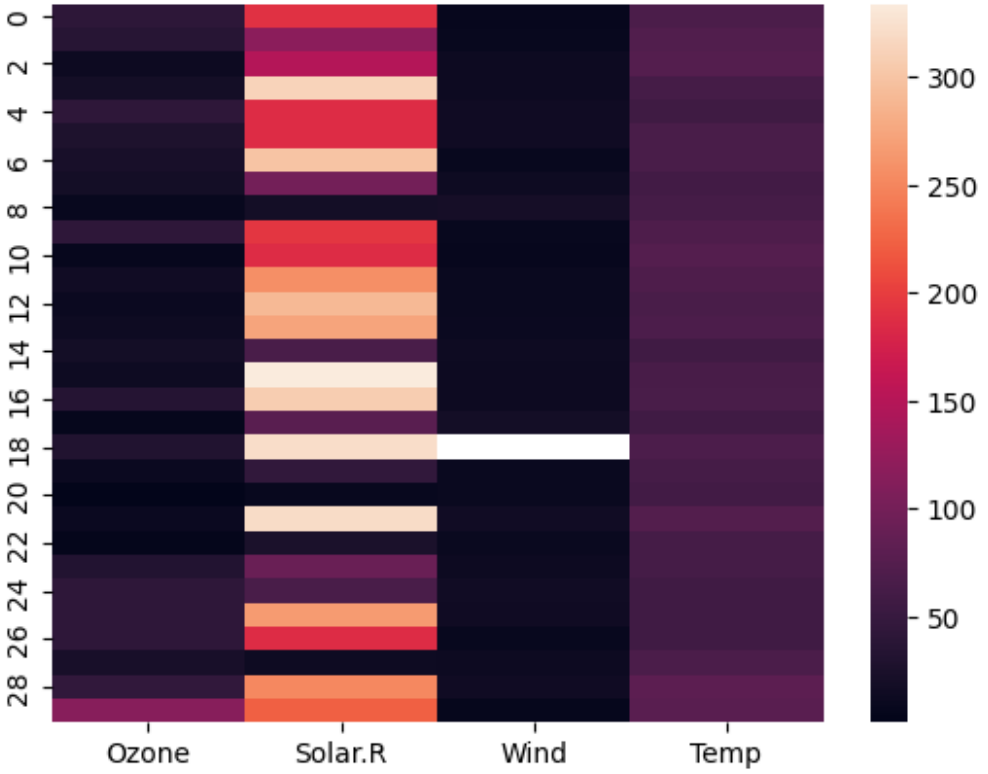
Pair Plot

In [30]:
```
import seaborn

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



Heatmap

In [31]:
```
seaborn.heatmap(data=data.iloc[0:30,[1,2,3,4]])
plt.show()
```



Word Cloud

In [35]:

```
from wordcloud import WordCloud, STOPWORDS

stopwords = set(STOPWORDS)
text = """Lorem Ipsum is simply dummy text of the printing and typesetting ind
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
It has survived not only five centuries, but also the leap into electronic typ
It was popularised in the 1960s with the release of Letraset sheets containing
and more recently with desktop publishing software like Aldus PageMaker includ

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()
```