```python
import numpy as np
import pandas as pd
import seaborn as sn
import matplotlib.pyplot as plt

df=pd.read_csv("forestfires.csv")

df
```

```
     X  Y month  day  FFMC    DMC     DC   ISI  temp  RH  wind  rain
area
0    7  5   mar  fri  86.2   26.2   94.3   5.1   8.2  51   6.7   0.0
0.00
1    7  4   oct  tue  90.6   35.4  669.1   6.7  18.0  33   0.9   0.0
0.00
2    7  4   oct  sat  90.6   43.7  686.9   6.7  14.6  33   1.3   0.0
0.00
3    8  6   mar  fri  91.7   33.3   77.5   9.0   8.3  97   4.0   0.2
0.00
4    8  6   mar  sun  89.3   51.3  102.2   9.6  11.4  99   1.8   0.0
0.00
..  .. ..   ...  ...   ...    ...    ...   ...   ...  ..   ...   ...
...
512  4  3   aug  sun  81.6   56.7  665.6   1.9  27.8  32   2.7   0.0
6.44
513  2  4   aug  sun  81.6   56.7  665.6   1.9  21.9  71   5.8   0.0
54.29
514  7  4   aug  sun  81.6   56.7  665.6   1.9  21.2  70   6.7   0.0
11.16
515  1  4   aug  sat  94.4  146.0  614.7  11.3  25.6  42   4.0   0.0
0.00
516  6  3   nov  tue  79.5    3.0  106.7   1.1  11.8  31   4.5   0.0
0.00

[517 rows x 13 columns]
```

```python
df.head()
```

```
   X  Y month  day  FFMC   DMC     DC  ISI  temp  RH  wind  rain  area
0  7  5   mar  fri  86.2  26.2   94.3  5.1   8.2  51   6.7   0.0   0.0
1  7  4   oct  tue  90.6  35.4  669.1  6.7  18.0  33   0.9   0.0   0.0
2  7  4   oct  sat  90.6  43.7  686.9  6.7  14.6  33   1.3   0.0   0.0
3  8  6   mar  fri  91.7  33.3   77.5  9.0   8.3  97   4.0   0.2   0.0
4  8  6   mar  sun  89.3  51.3  102.2  9.6  11.4  99   1.8   0.0   0.0
```

```python
df.tail
```

```
<bound method NDFrame.tail of      X  Y month  day  FFMC    DMC     DC
ISI  temp  RH  wind  rain   area
0    7  5   mar  fri  86.2   26.2   94.3  5.1   8.2  51   6.7   0.0
0.00
```

```
1     7   4    oct   tue   90.6    35.4   669.1    6.7   18.0   33    0.9    0.0
0.00
2     7   4    oct   sat   90.6    43.7   686.9    6.7   14.6   33    1.3    0.0
0.00
3     8   6    mar   fri   91.7    33.3    77.5    9.0    8.3   97    4.0    0.2
0.00
4     8   6    mar   sun   89.3    51.3   102.2    9.6   11.4   99    1.8    0.0
0.00
..   ..  ..   ...   ...   ...     ...    ...      ...   ...   ..   ...    ...
...
512   4   3    aug   sun   81.6    56.7   665.6    1.9   27.8   32    2.7    0.0
6.44
513   2   4    aug   sun   81.6    56.7   665.6    1.9   21.9   71    5.8    0.0
54.29
514   7   4    aug   sun   81.6    56.7   665.6    1.9   21.2   70    6.7    0.0
11.16
515   1   4    aug   sat   94.4   146.0   614.7   11.3   25.6   42    4.0    0.0
0.00
516   6   3    nov   tue   79.5     3.0   106.7    1.1   11.8   31    4.5    0.0
0.00

[517 rows x 13 columns]>

df.dtypes

X           int64
Y           int64
month      object
day        object
FFMC      float64
DMC       float64
DC        float64
ISI       float64
temp      float64
RH          int64
wind      float64
rain      float64
area      float64
dtype: object
```

Using Matplotlib

```python
#Scatter Plot of month and rain

plt.scatter(df['month'], df['rain'])

# Adding Title to the Plot
plt.title("Scatter Plot")

# Setting the X and Y labels
```
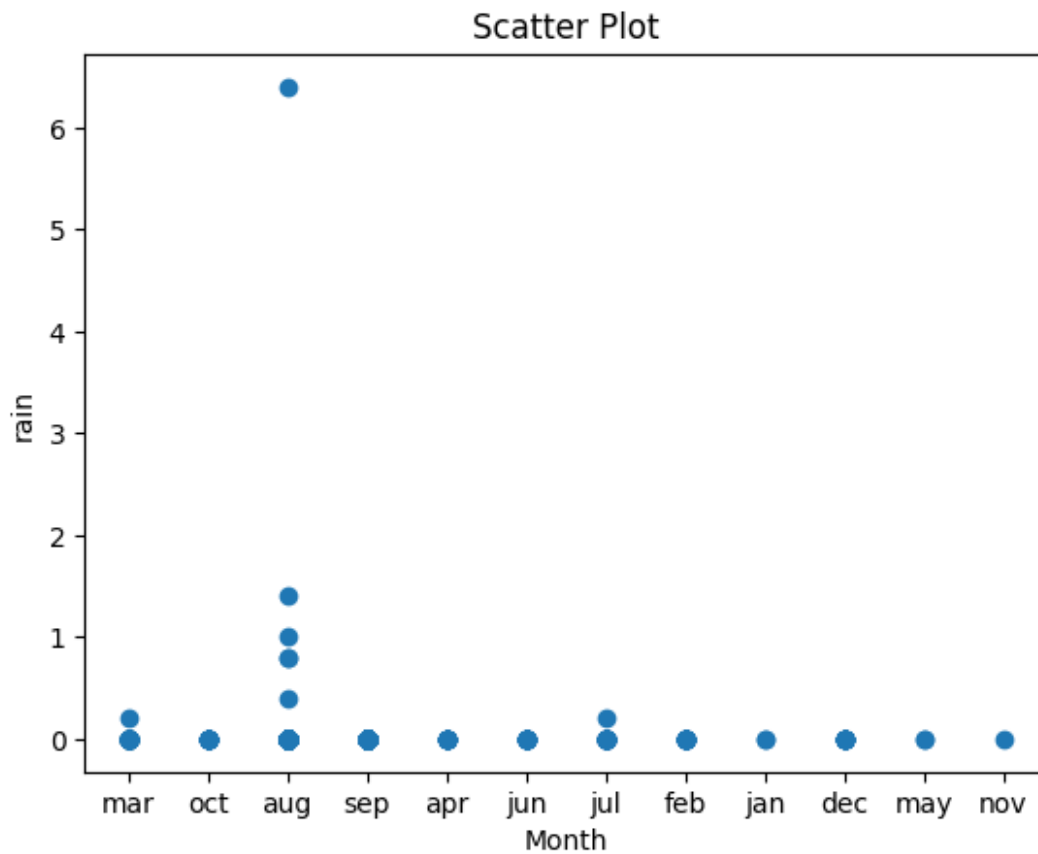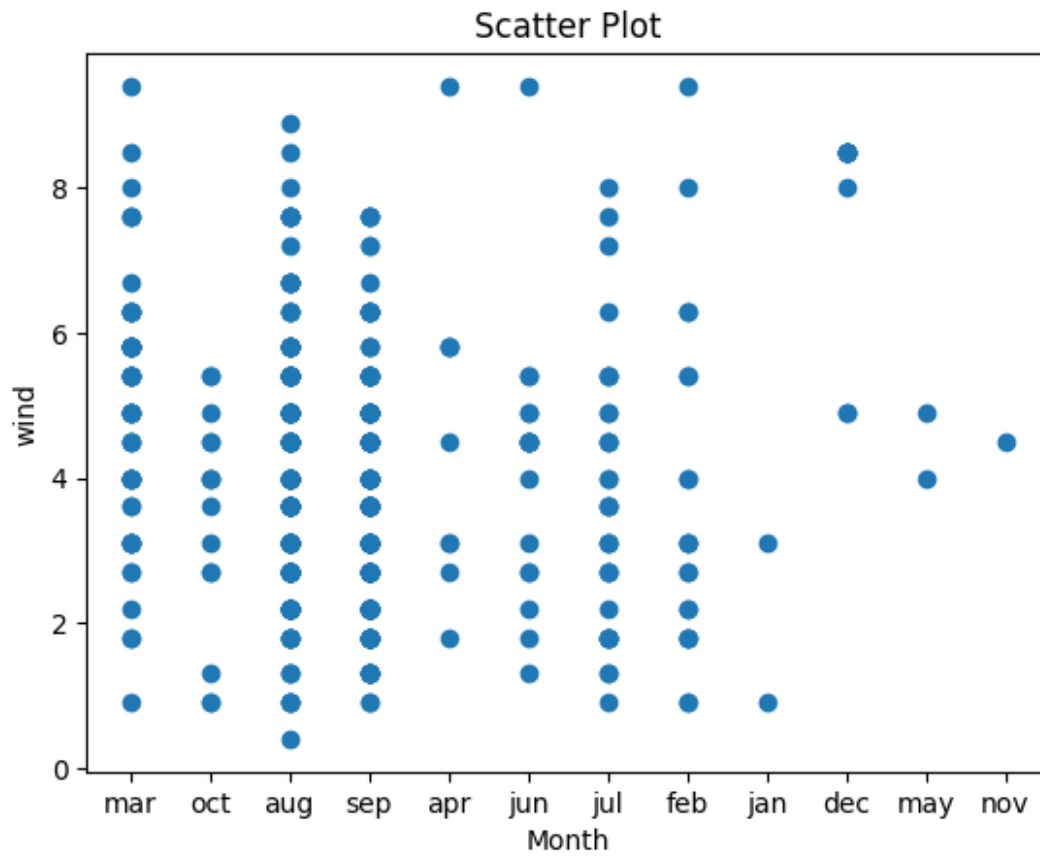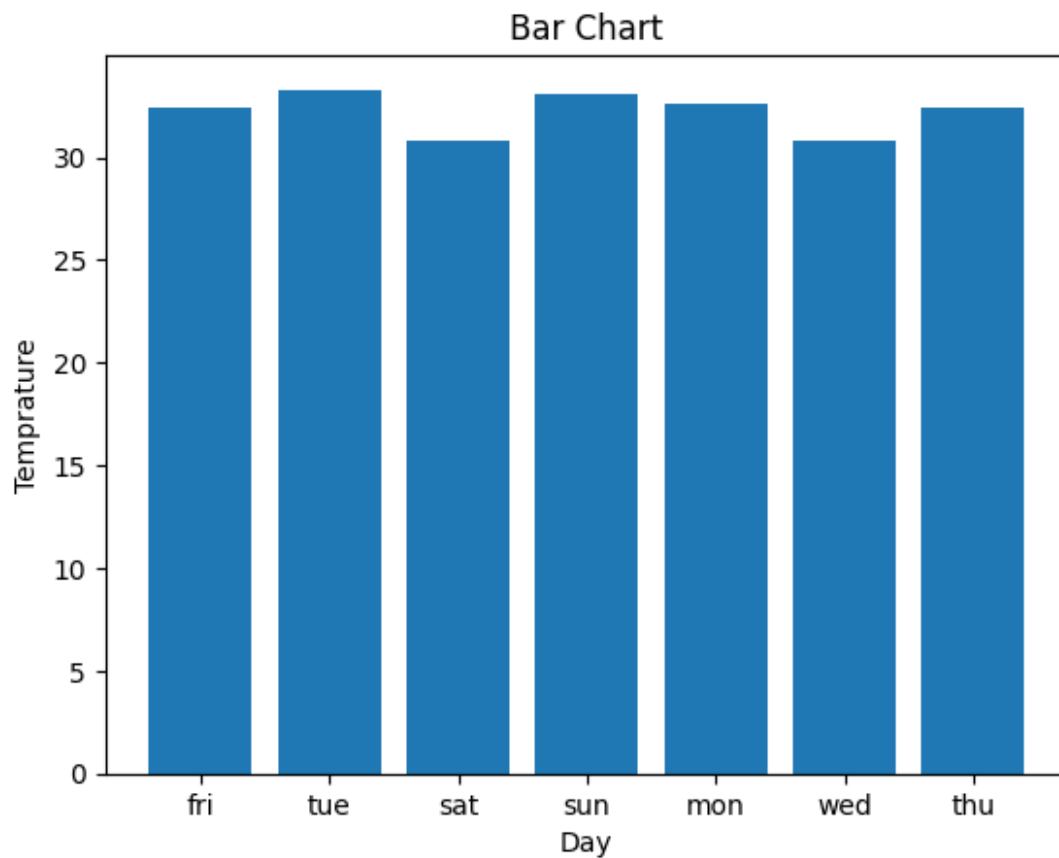
```python
plt.xlabel('Month')
plt.ylabel('rain')


plt.show()
```


Scatter Plot

```python
#Scatter Plot of month and wind

plt.scatter(df['month'], df['wind'])

# Adding Title to the Plot
plt.title("Scatter Plot")

# Setting the X and Y labels
plt.xlabel('Month')
plt.ylabel('wind')


plt.show()
```
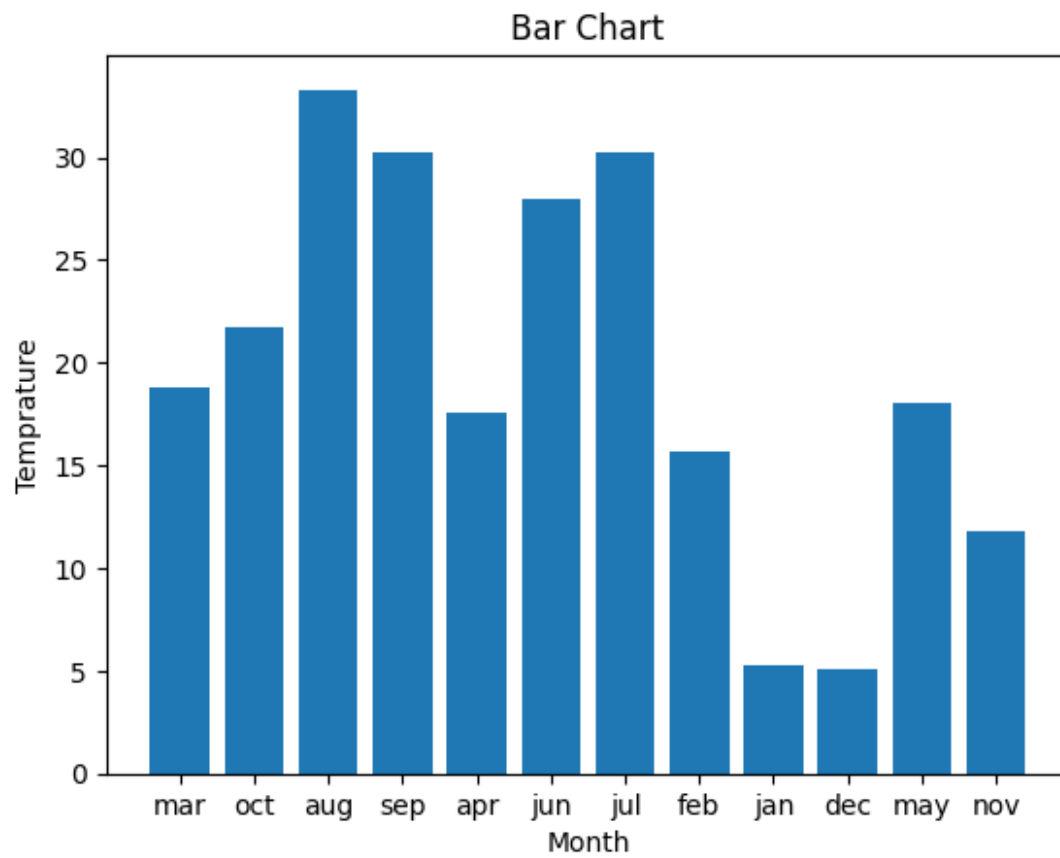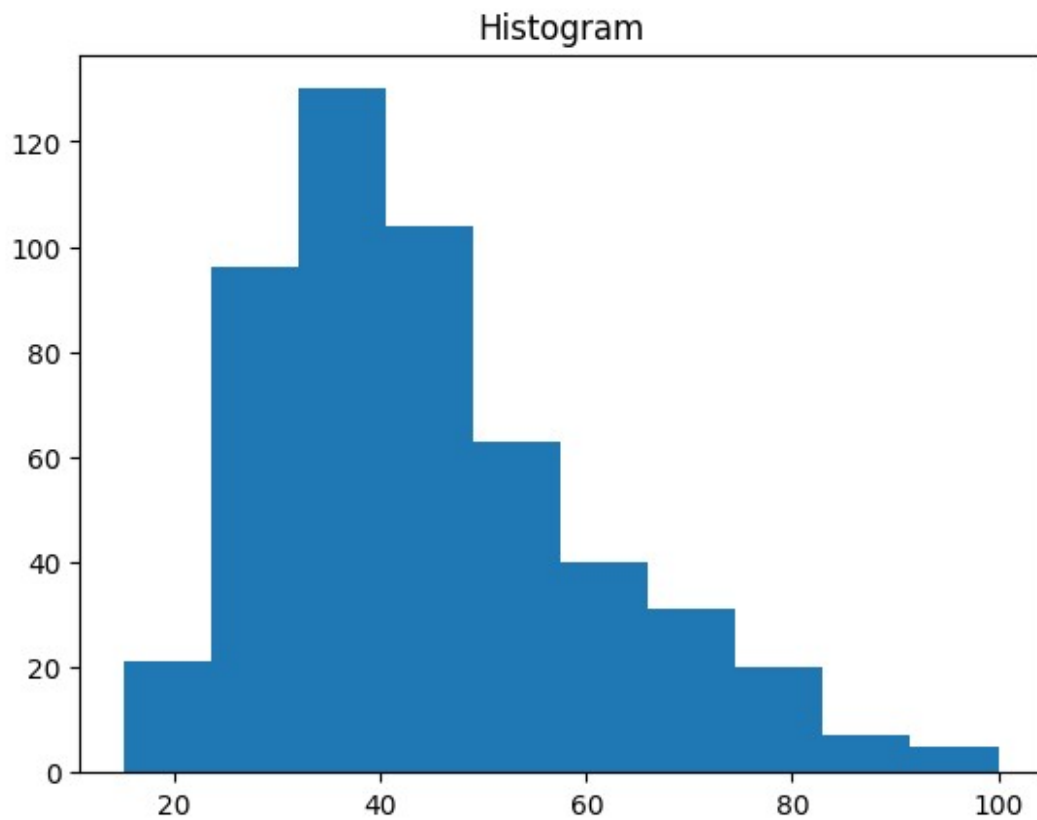
Scatter Plot

```python
#Bar chart with day against temprature
plt.bar(df['day'], df['temp'])

plt.title("Bar Chart")
plt.xlabel('Day')
plt.ylabel('Temprature')

plt.show()
```

Bar Chart


```python
#Bar chart with month against temprature
plt.bar(df['month'], df['temp'])

plt.title("Bar Chart")
plt.xlabel('Month')
plt.ylabel('Temprature')

plt.show()
```
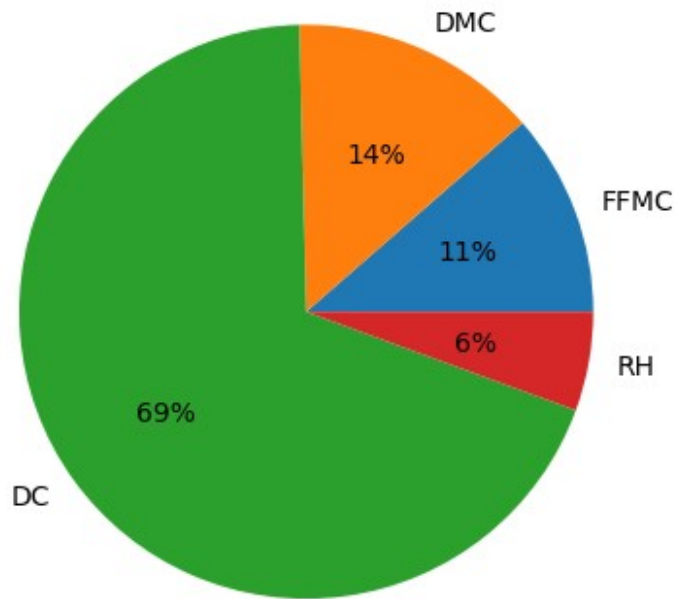
Bar Chart

```
#Histogram
plt.hist(df['RH'])
plt.title("Histogram")
plt.show()
```
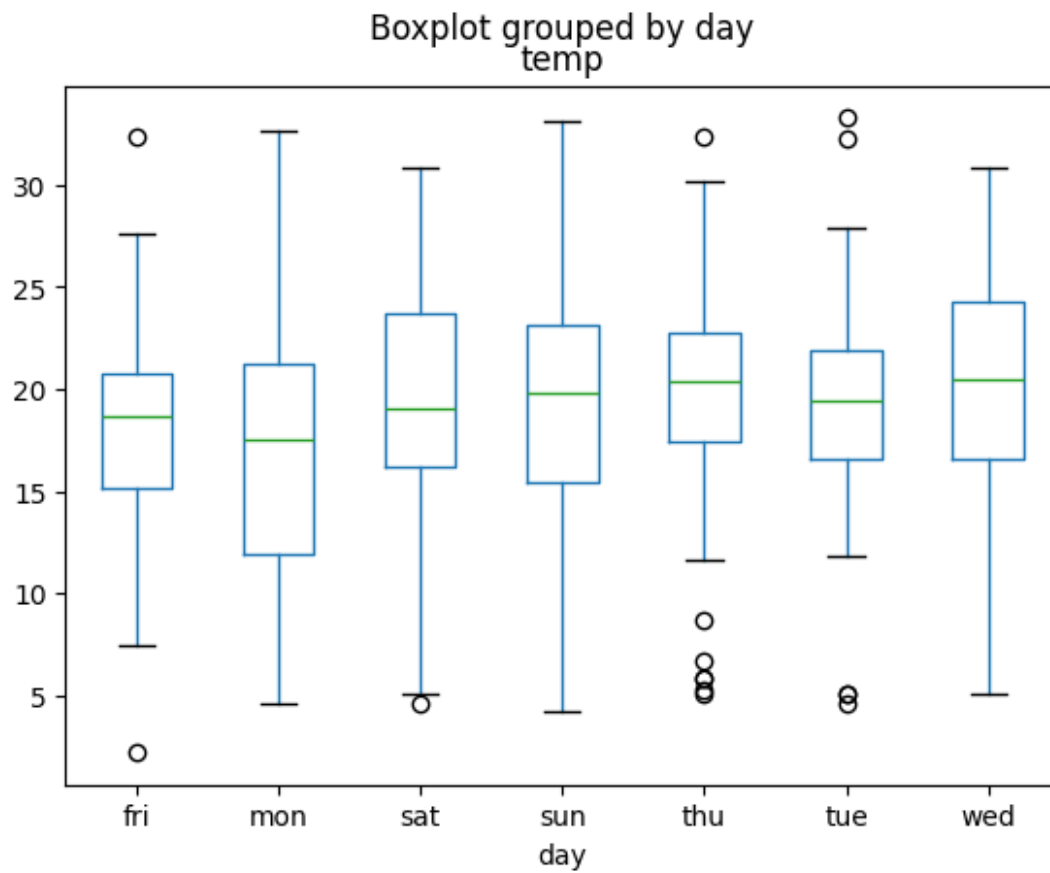
Histogram

```
#Pie Chart
keys='FFMC','DMC','DC','RH'
sizes=[df['FFMC'].sum(),df['DMC'].sum(),df['DC'].sum(),df['RH'].sum()]
plt.pie(sizes, labels=keys, autopct='%.0f%%')
plt.title('Values of fire weather indices')
plt.show()
```

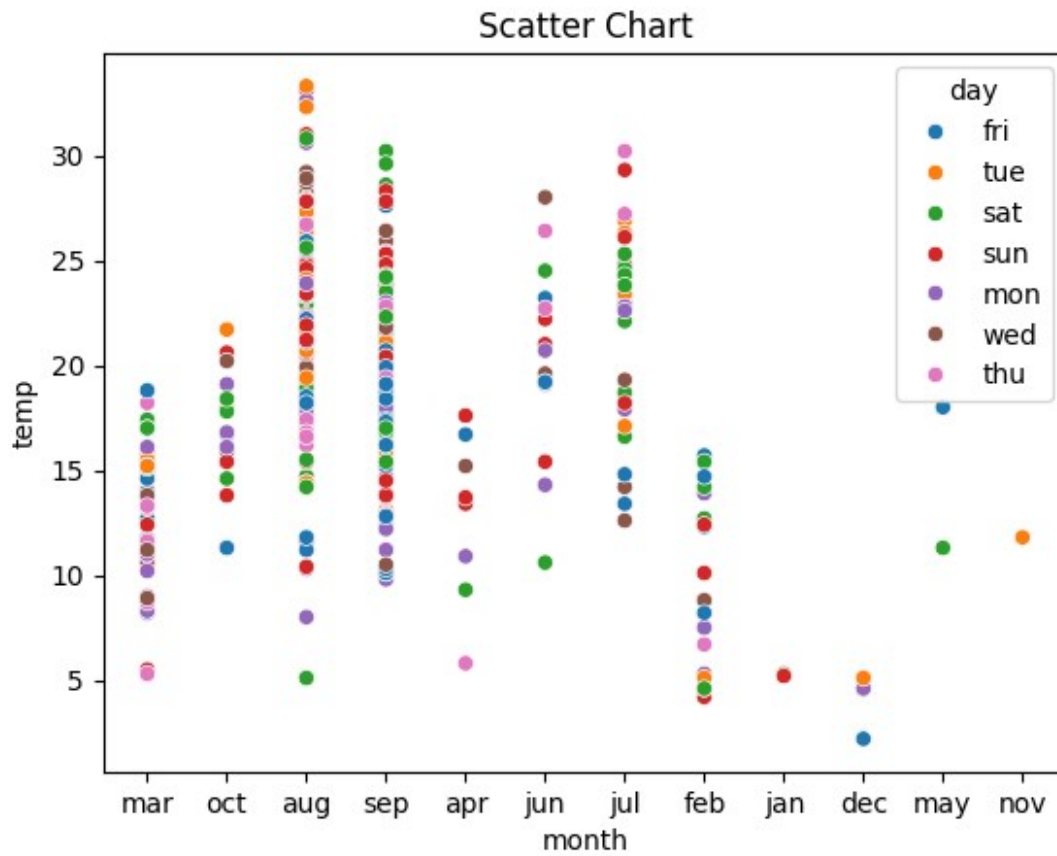## Values of fire weather indices


```
#Box Plot
df
df.boxplot(by ='day', column =['temp'], grid = False)

<Axes: title={'center': 'temp'}, xlabel='day'>
```
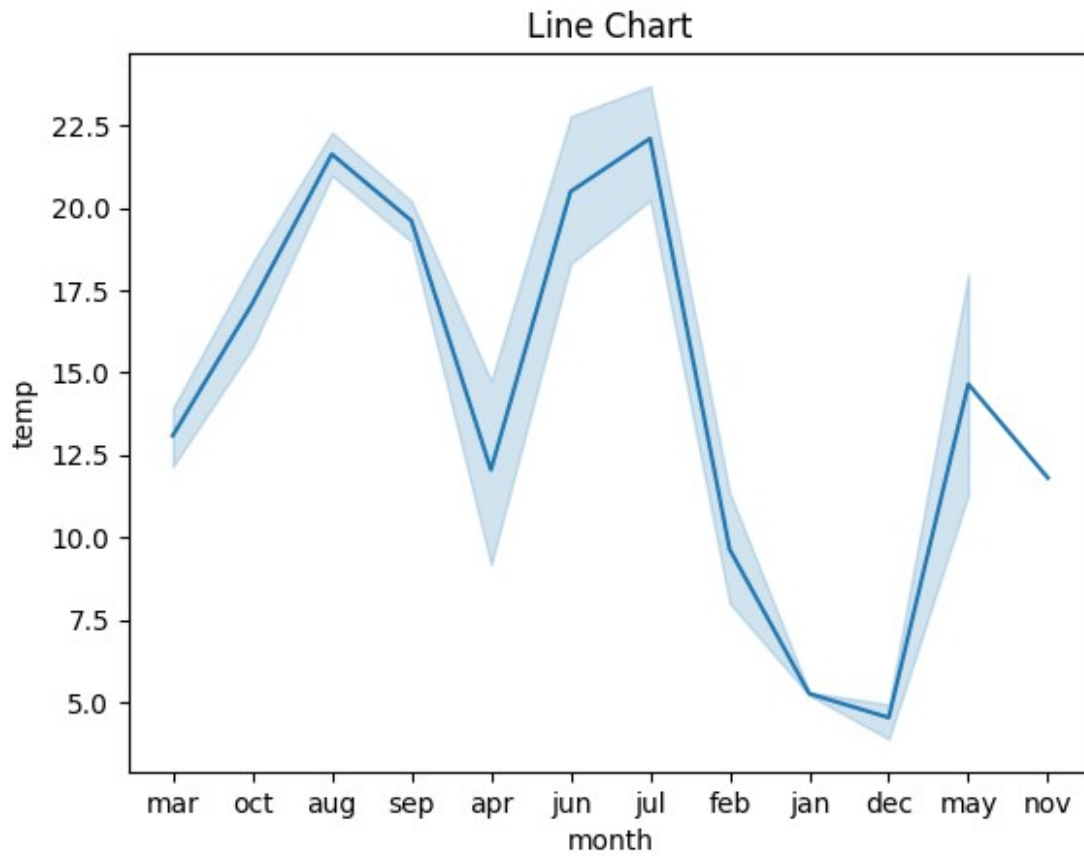
Boxplot grouped by day
temp

Using Seaborn

```python
#Scatter Chart

sn.scatterplot(x='month', y='temp', data=df,hue='day')
plt.title('Scatter Chart')
plt.show()
```
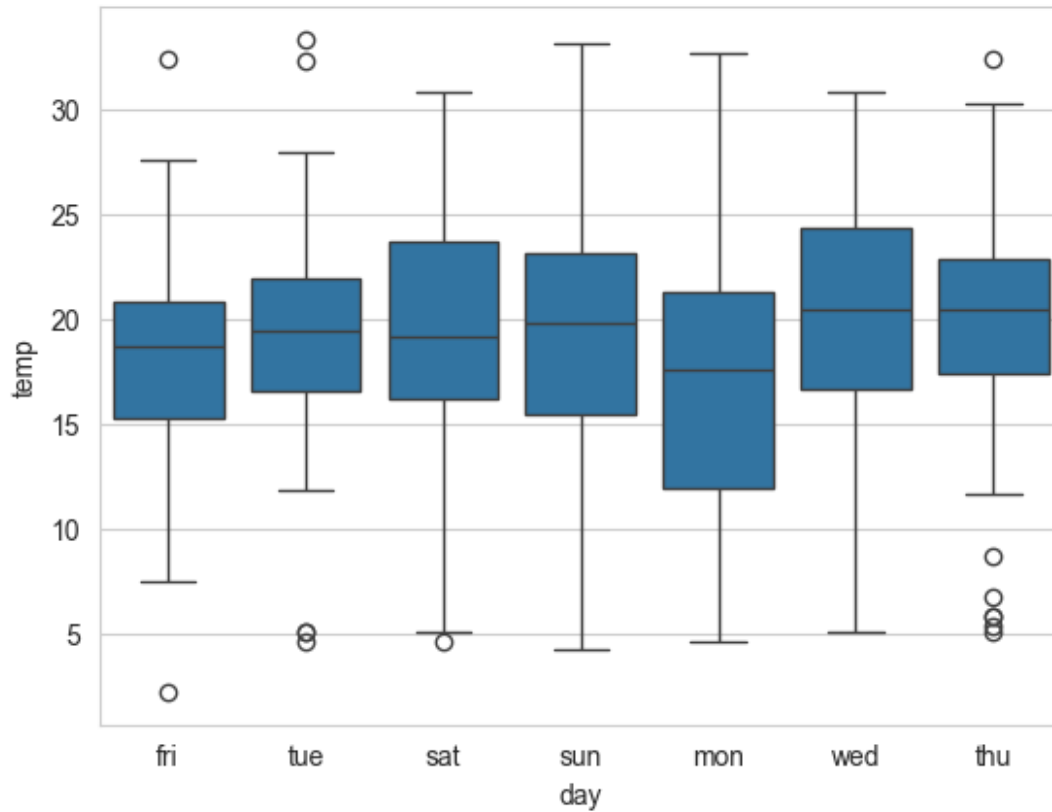
```
# Line Plot
sn.lineplot(x="month", y="temp", data=df)
plt.title('Line Chart')
plt.show()
```

Line Chart


```python
# Drawing a vertical boxplot grouped
sn.set_style("whitegrid")
sn.boxplot(x = 'day', y = 'temp', data = df)
```

```
<Axes: xlabel='day', ylabel='temp'>
```

```python
# Keep only numeric columns
numeric_df = df.select_dtypes(include=['number'])

# Calculate the correlation matrix
corr_matrix = numeric_df.corr()

# Plot the heatmap
sns.heatmap(corr_matrix, annot=True, cmap='coolwarm')
plt.title('Correlation Heatmap')
plt.show()
```

Correlation Heatmap