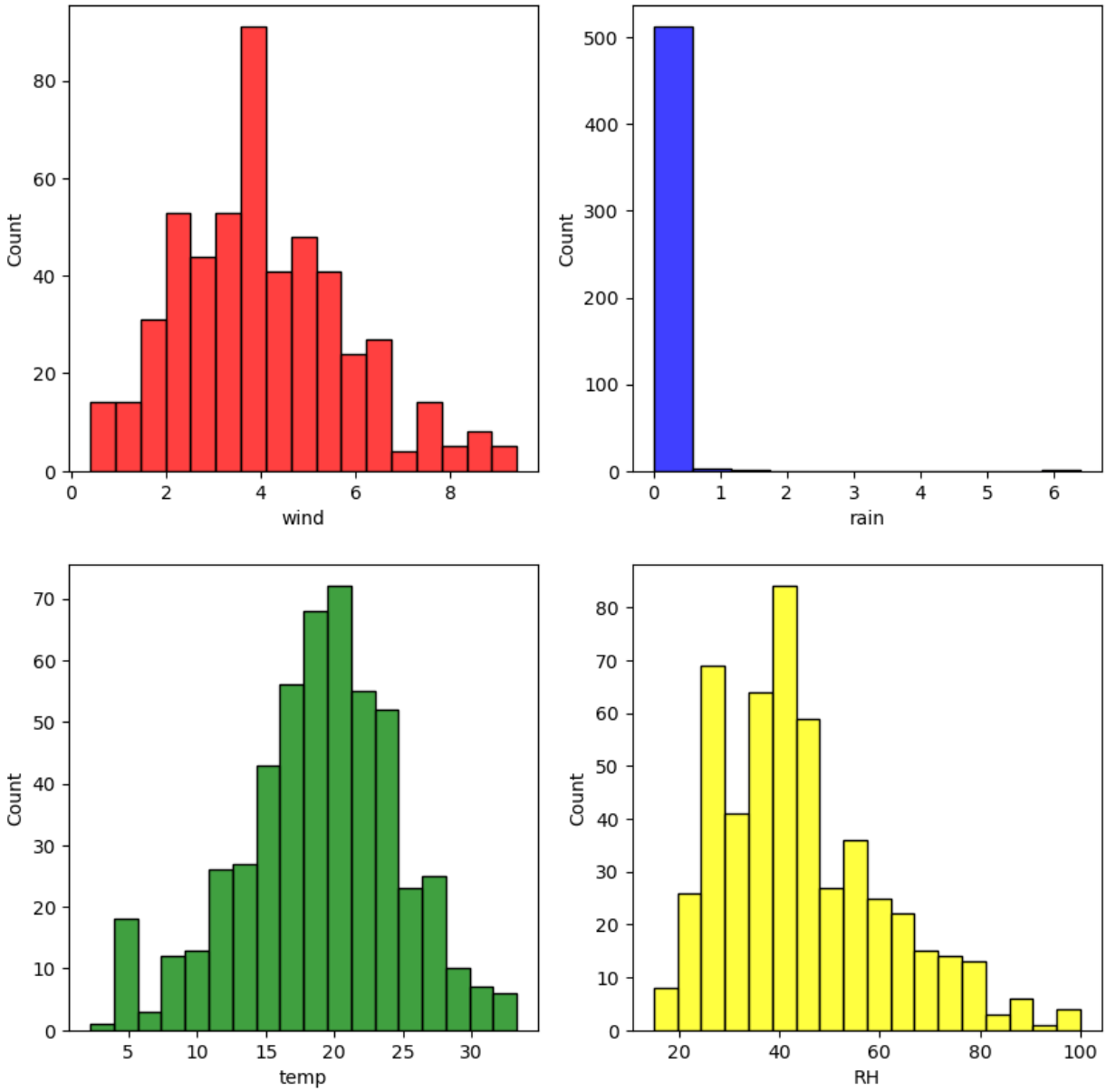
In [14]:
```
# Exercise 1:

import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns

df = pd.read_csv (r"C:\Users\kriti\Downloads\forestfires.csv")

f, axes = plt.subplots (2, 2, figsize = (10, 10), sharex = False, sharey = False)
sns.histplot(df["wind"], color = 'red', ax = axes[0, 0])
sns.histplot(df["rain"], color = 'blue', ax = axes[0, 1])
sns.histplot(df["temp"], color = 'green', ax = axes[1, 0])
sns.histplot(df["RH"], color = 'yellow', ax = axes[1, 1])
```

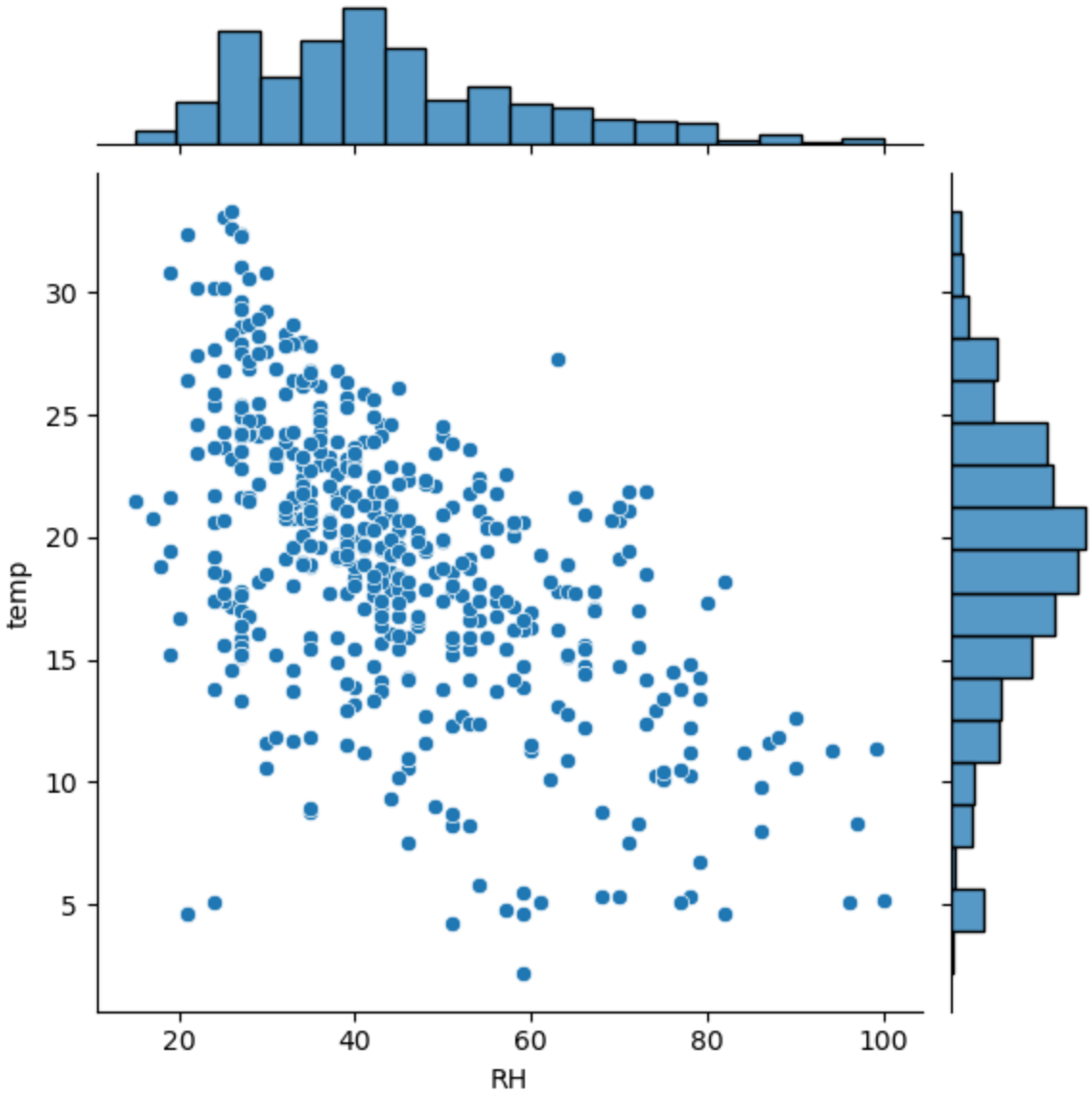
Out[14]:
```
<Axes: xlabel='RH', ylabel='Count'>
```



In [15]:
```
# Exercise 2:

sns.jointplot (x= df["RH"], y= df["temp"], kind= 'scatter')
```

Out[15]: <seaborn.axisgrid.JointGrid at 0x183cce7b850>



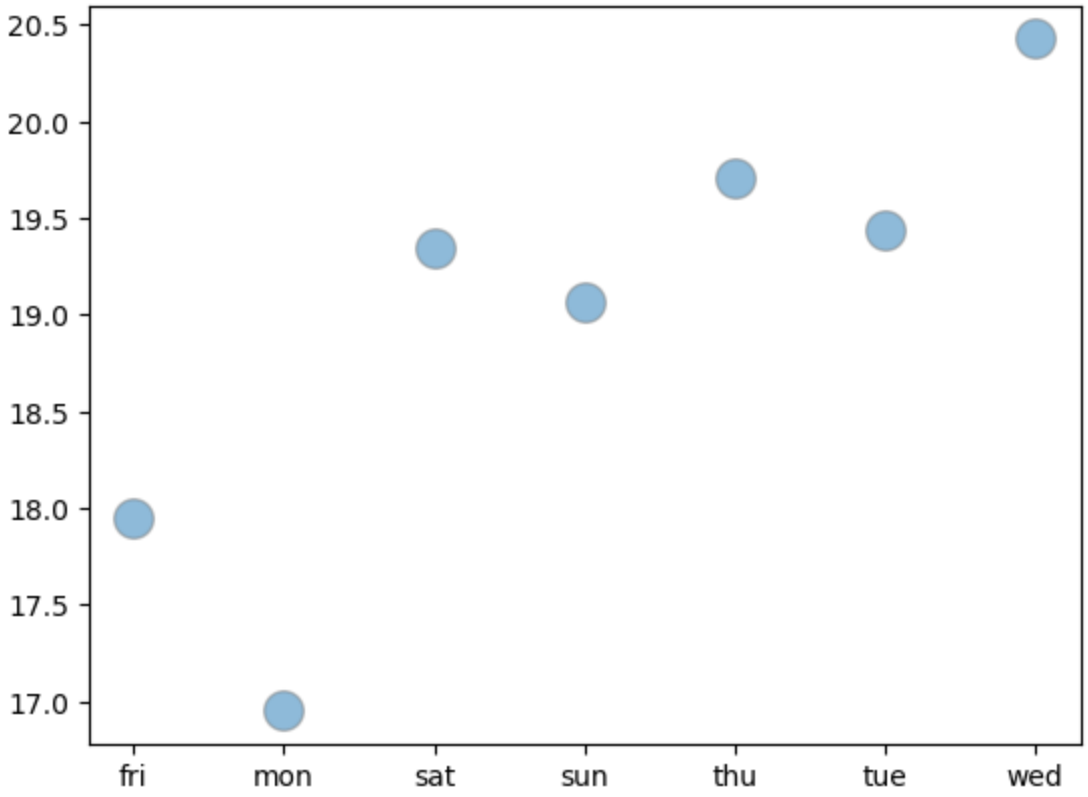
In [17]:
```
# Exercise 3:

df2= df.groupby ("day")['temp'].aggregate ('mean')
df2= pd.DataFrame (df2)
df2.reset_index (inplace= True)

plt.scatter (x= df2['day'], y= df2['temp'], s= 200, cmap= 'green', alpha= 0.5, edgecol
```

```
C:\Users\kriti\AppData\Local\Temp\ipykernel_14260\1587965550.py:7: UserWarning: No da
ta for colormapping provided via 'c'. Parameters 'cmap' will be ignored
  plt.scatter (x= df2['day'], y= df2['temp'], s= 200, cmap= 'green', alpha= 0.5, edge
colors= 'gray')
```

Out[17]: <matplotlib.collections.PathCollection at 0x183d1c5d550>

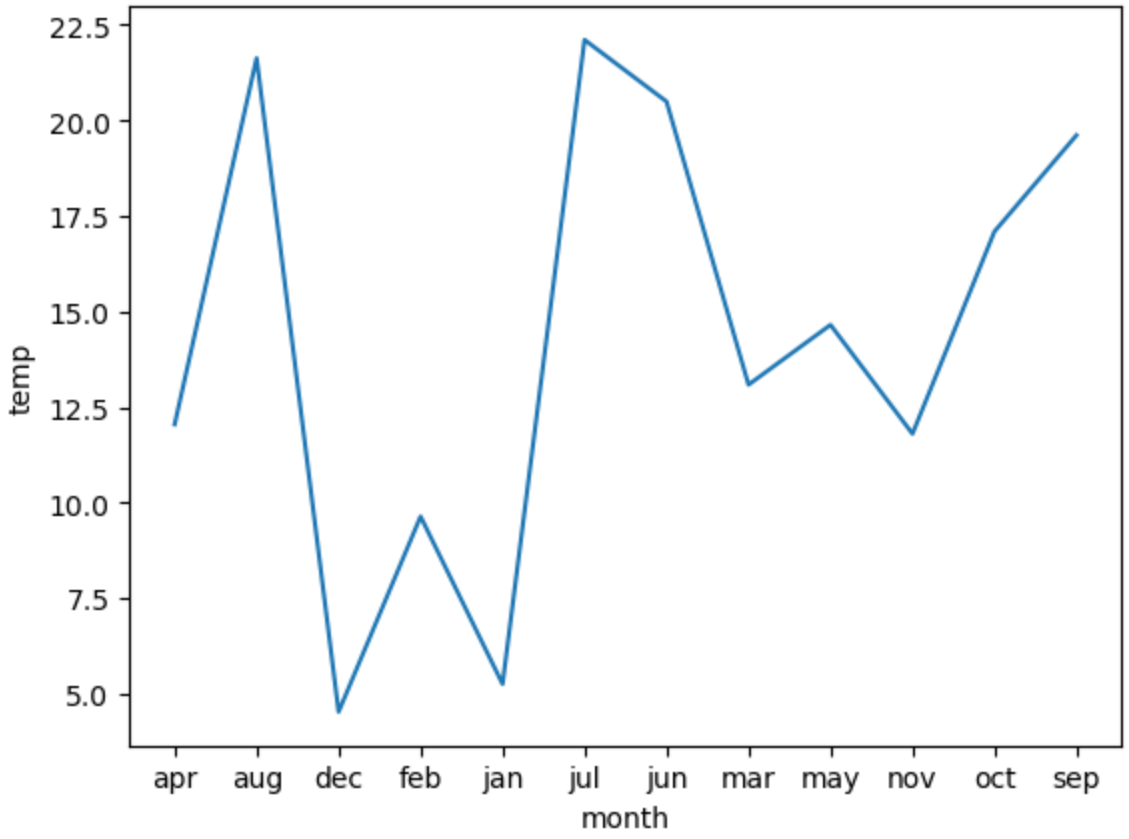


In [18]:
```
# Exercise 4:

df3= df.groupby ("month")['temp'].aggregate ('mean')
df3= pd.DataFrame (df3)
df3.reset_index (inplace= True)

sns.lineplot(x= "month", y= "temp", data= df3)
```

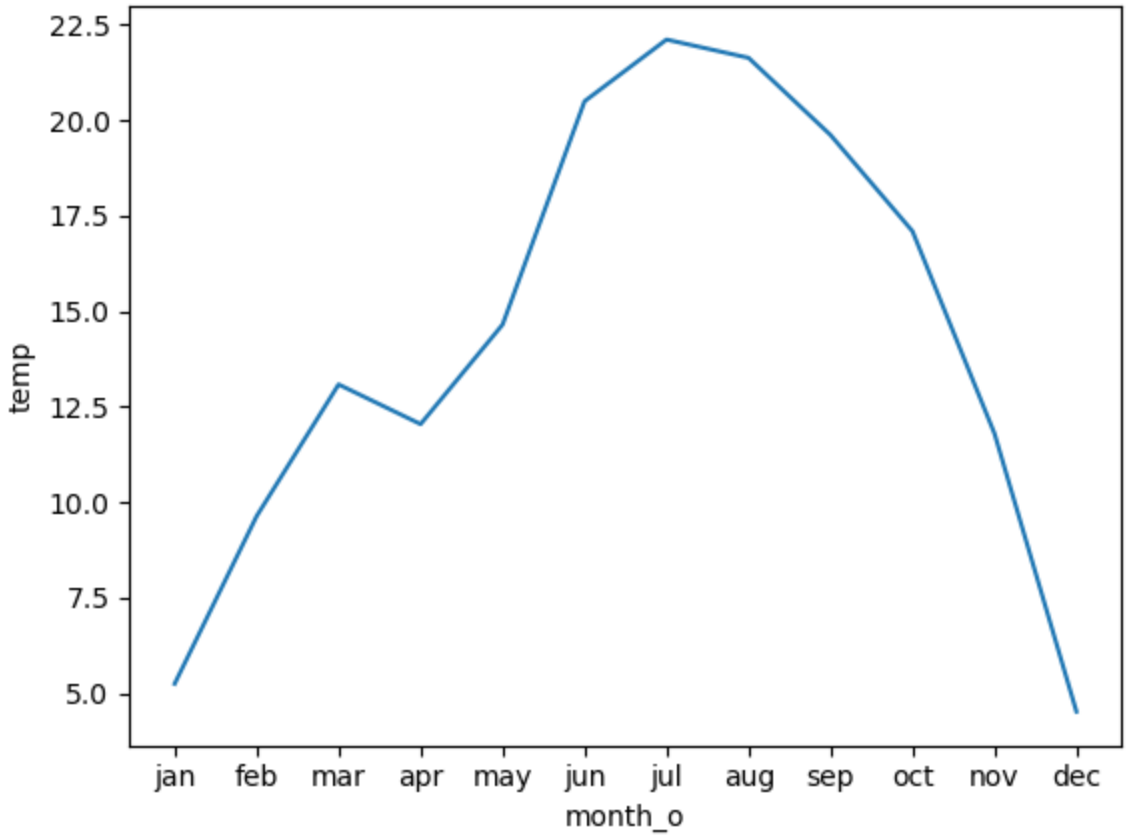
Out[18]:
```
<Axes: xlabel='month', ylabel='temp'>
```



In [21]:
```
# Exercise 5:

df3['month_o']= pd.Categorical (df3['month'], ['jan', 'feb', 'mar', 'apr', 'may', 'jur
sns.lineplot (x= "month_o", y= "temp", data= df3)
```

Out[21]:
```
<Axes: xlabel='month_o', ylabel='temp'>
```



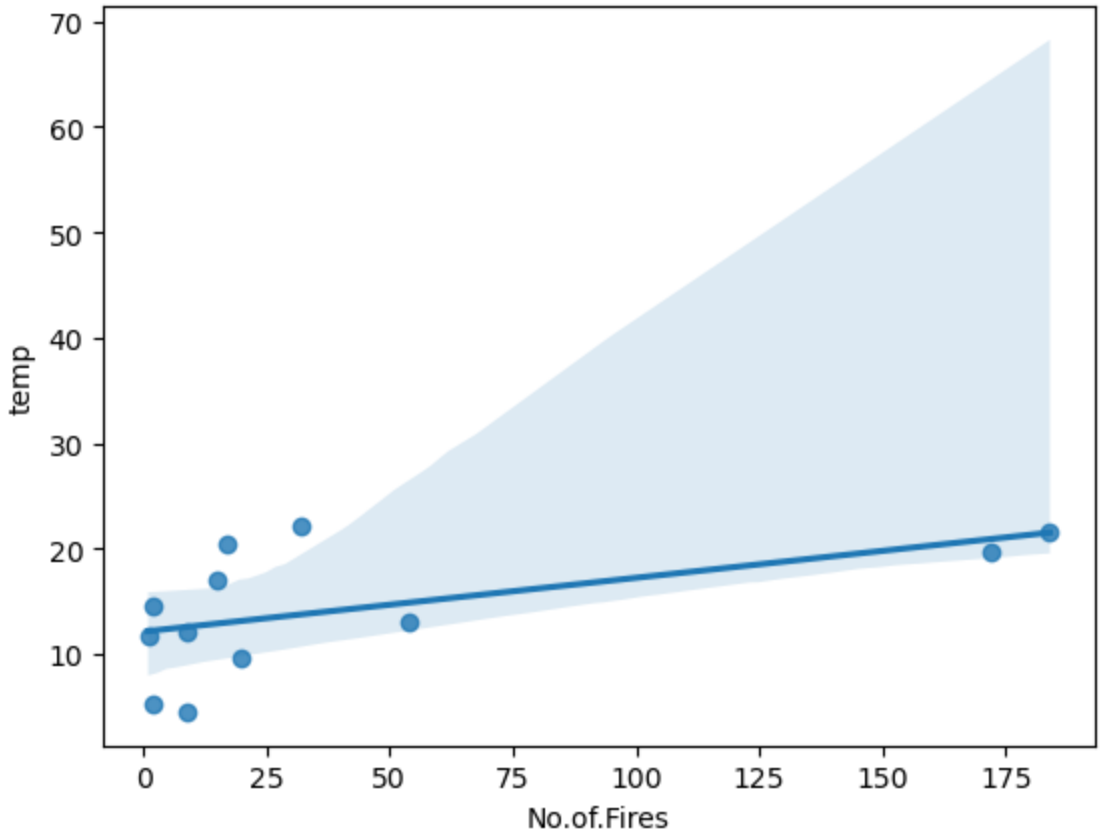
In [25]:
```
# Exercise 6:

df4= df.groupby('month')['month'].aggregate('count')
df4= pd.DataFrame (df4)
df4.rename (columns= {"month": "No.of.Fires"}, inplace= True)
df4.reset_index()

final= df4.merge(df3, on= "month")

sns.regplot(x= final['No.of.Fires'], y= final['temp'])
```

Out[25]:
```
<Axes: xlabel='No.of.Fires', ylabel='temp'>
```


70
60
50
40
30
20
10
temp
0
25
50
75
100
125
150
175
No.of.Fires