In [1]:
```
import pandas as pd
import numpy as np
```

In [4]:
```
df=pd.read_csv("forestfires.csv")
```

In [5]:
```
df.head()
```

Out[5]:

	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

In [6]:
```
df.info
```

Out[6]:
```
<bound method DataFrame.info 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]>
```

In [7]:
```
df.describe()
```

Out[7]:

	X	Y	FFMC	DMC	DC	ISI	te
count	517.000000	517.000000	517.000000	517.000000	517.000000	517.000000	517.000(
mean	4.669246	4.299807	90.644681	110.872340	547.940039	9.021663	18.889
std	2.313778	1.229900	5.520111	64.046482	248.066192	4.559477	5.806(
min	1.000000	2.000000	18.700000	1.100000	7.900000	0.000000	2.200(
25%	3.000000	4.000000	90.200000	68.600000	437.700000	6.500000	15.500(
50%	4.000000	4.000000	91.600000	108.300000	664.200000	8.400000	19.300(
75%	7.000000	5.000000	92.900000	142.400000	713.900000	10.800000	22.800(
max	9.000000	9.000000	96.200000	291.300000	860.600000	56.100000	33.300(

In [8]:
```
df.shape
```

Out[8]: (517, 13)

In [9]:
```
df1=df[[ 'X' , 'Y' , 'month' , 'day']]
df1
```

Out[9]:

	X	Y	month	day
0	7	5	mar	fri
1	7	4	oct	tue
2	7	4	oct	sat
3	8	6	mar	fri
4	8	6	mar	sun
...	...	...	...	...
512	4	3	aug	sun
513	2	4	aug	sun
514	7	4	aug	sun
515	1	4	aug	sat
516	6	3	nov	tue

517 rows × 4 columns

In [11]:
```
df2=df[[ 'FFMC' , 'DMC' , 'DC' , 'temp']]
df2
```

Out[11]:

	FFMC	DMC	DC	temp
0	86.2	26.2	94.3	8.2
1	90.6	35.4	669.1	18.0
2	90.6	43.7	686.9	14.6
3	91.7	33.3	77.5	8.3
4	89.3	51.3	102.2	11.4
...	...	...	...	...
512	81.6	56.7	665.6	27.8
513	81.6	56.7	665.6	21.9
514	81.6	56.7	665.6	21.2
515	94.4	146.0	614.7	25.6
516	79.5	3.0	106.7	11.8

517 rows × 4 columns

In [12]:
```
df3=df[[ 'temp' , 'wind' , 'rain' , 'area']]
df3
```

Out[12]:

	temp	wind	rain	area
0	8.2	6.7	0.0	0.00
1	18.0	0.9	0.0	0.00
2	14.6	1.3	0.0	0.00
3	8.3	4.0	0.2	0.00
4	11.4	1.8	0.0	0.00
...	...	...	...	...
512	27.8	2.7	0.0	6.44
513	21.9	5.8	0.0	54.29
514	21.2	6.7	0.0	11.16
515	25.6	4.0	0.0	0.00
516	11.8	4.5	0.0	0.00

517 rows × 4 columns

In [13]:
```
#merging subset
merging=pd.concat([df1,df2,df3])
merging
```

Out[13]:

	X	Y	month	day	FFMC	DMC	DC	temp	wind	rain	area
0	7.0	5.0	mar	fri	NaN	NaN	NaN	NaN	NaN	NaN	NaN
1	7.0	4.0	oct	tue	NaN	NaN	NaN	NaN	NaN	NaN	NaN
2	7.0	4.0	oct	sat	NaN	NaN	NaN	NaN	NaN	NaN	NaN
3	8.0	6.0	mar	fri	NaN	NaN	NaN	NaN	NaN	NaN	NaN
4	8.0	6.0	mar	sun	NaN	NaN	NaN	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...	...	...	...
512	NaN	NaN	NaN	NaN	NaN	NaN	NaN	27.8	2.7	0.0	6.44
513	NaN	NaN	NaN	NaN	NaN	NaN	NaN	21.9	5.8	0.0	54.29
514	NaN	NaN	NaN	NaN	NaN	NaN	NaN	21.2	6.7	0.0	11.16
515	NaN	NaN	NaN	NaN	NaN	NaN	NaN	25.6	4.0	0.0	0.00
516	NaN	NaN	NaN	NaN	NaN	NaN	NaN	11.8	4.5	0.0	0.00

1551 rows × 11 columns

In [14]:
```
#sort data
sort_values=df.sort_values('X' , ascending=True)
sort_values
```

Out[14]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
439	1	3	sep	fri	91.1	91.3	738.1	7.2	19.1	46	2.2	0.0	0.33
87	1	2	sep	thu	92.9	137.0	706.4	9.2	22.4	34	2.2	0.0	0.00
86	1	2	sep	thu	92.9	137.0	706.4	9.2	25.4	27	2.2	0.0	0.00
85	1	2	sep	thu	92.9	137.0	706.4	9.2	21.5	15	0.9	0.0	0.00
84	1	2	aug	thu	91.7	114.3	661.3	6.3	20.2	45	3.6	0.0	0.00
...	...	...	...	...	...	...	...	...	...	...	...	...	...
473	9	4	jun	sat	90.5	61.1	252.6	9.4	24.5	50	3.1	0.0	70.32
383	9	6	aug	thu	91.6	248.4	753.8	6.3	20.5	58	2.7	0.0	42.87
412	9	4	jul	mon	92.3	92.1	442.1	9.8	22.8	27	4.5	0.0	1.63
402	9	9	aug	fri	94.8	227.0	706.7	12.0	25.0	36	4.0	0.0	0.00
76	9	9	feb	fri	86.6	13.2	43.0	5.3	15.7	43	3.1	0.0	0.00

517 rows × 13 columns

In [15]:
```
#sort data
sort_values=df.sort_values('RH' , ascending=True)
sort_values
```

Out[15]:

	X	Y	month	day	FFMC	DMC	DC	ISI	temp	RH	wind	rain	area
85	1	2	sep	thu	92.9	137.0	706.4	9.2	21.5	15	0.9	0.0	0.00
197	4	5	sep	thu	92.9	137.0	706.4	9.2	21.5	15	0.9	0.0	11.06
50	4	4	sep	thu	92.9	137.0	706.4	9.2	20.8	17	1.3	0.0	0.00
111	3	4	mar	fri	91.7	33.3	77.5	9.0	18.8	18	4.5	0.0	0.00
218	4	5	sep	wed	92.9	133.3	699.6	9.2	19.4	19	1.3	0.0	31.72
...	...	...	...	...	...	...	...	...	...	...	...	...	...
304	6	5	may	sat	85.1	28.0	113.8	3.5	11.3	94	4.9	0.0	0.00
211	7	4	aug	sat	93.5	139.4	594.2	20.3	5.1	96	5.8	0.0	26.00
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
379	4	5	jan	sun	18.7	1.1	171.4	0.0	5.2	100	0.9	0.0	0.00

517 rows × 13 columns

In [16]:
```
df.transpose()
```

Out[16]:

	0	1	2	3	4	5	6	7	8	9	...	507	50
X	7	7	7	8	8	8	8	8	8	7	...	2	
Y	5	4	4	6	6	6	6	6	6	5	...	4	
month	mar	oct	oct	mar	mar	aug	aug	aug	sep	sep	...	aug	au
day	fri	tue	sat	fri	sun	sun	mon	mon	tue	sat	...	fri	f
FFMC	86.2	90.6	90.6	91.7	89.3	92.3	92.3	91.5	91.0	92.5	...	91.0	91.
DMC	26.2	35.4	43.7	33.3	51.3	85.3	88.9	145.4	129.5	88.0	...	166.9	166.
DC	94.3	669.1	686.9	77.5	102.2	488.0	495.6	608.2	692.6	698.6	...	752.6	752.
ISI	5.1	6.7	6.7	9.0	9.6	14.7	8.5	10.7	7.0	7.1	...	7.1	7.
temp	8.2	18.0	14.6	8.3	11.4	22.2	24.1	8.0	13.1	22.8	...	25.9	25.
RH	51	33	33	97	99	29	27	86	63	40	...	41	4
wind	6.7	0.9	1.3	4.0	1.8	5.4	3.1	2.2	5.4	4.0	...	3.6	3.
rain	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.
area	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	...	0.0	0.

13 rows × 517 columns

In [17]:
```
#shaping data
shaping=df.shape
shaping
```

Out[17]:
```
(517, 13)
```

In [21]:
```
pivot_table=pd.pivot_table(df, index=['wind' , 'area'],values='RH')
pivot_table
```

Out[21]:

		RH
wind	**area**	
0.4	**2.47**	40.000
0.9	**0.00**	51.875
	2.29	47.000
	6.84	59.000
	7.40	42.000
...	**...**	...
8.9	**0.61**	51.000
9.4	**2.53**	26.000
	3.19	40.000
	4.62	53.000
	61.13	33.000

286 rows × 1 columns

In [22]:
```
#reading data
reshaping_arr=np.array([1,2,3,4,5,6])
reshaping_arr.reshape(3,2)
```

Out[22]:
```
array([[1, 2],
       [3, 4],
       [5, 6]])
```

In [23]:
```
df_integration=pd.concat([df1,df2])
df_integration
```

Out[23]:

	X	Y	month	day	FFMC	DMC	DC	temp
0	7.0	5.0	mar	fri	NaN	NaN	NaN	NaN
1	7.0	4.0	oct	tue	NaN	NaN	NaN	NaN
2	7.0	4.0	oct	sat	NaN	NaN	NaN	NaN
3	8.0	6.0	mar	fri	NaN	NaN	NaN	NaN
4	8.0	6.0	mar	sun	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...
512	NaN	NaN	NaN	NaN	81.6	56.7	665.6	27.8
513	NaN	NaN	NaN	NaN	81.6	56.7	665.6	21.9
514	NaN	NaN	NaN	NaN	81.6	56.7	665.6	21.2
515	NaN	NaN	NaN	NaN	94.4	146.0	614.7	25.6
516	NaN	NaN	NaN	NaN	79.5	3.0	106.7	11.8

1034 rows × 8 columns

In [24]:
```
df_integration.transpose()
```

Out[24]:

	0	1	2	3	4	5	6	7	8	9	...	507	508	5
X	7.0	7.0	7.0	8.0	8.0	8.0	8.0	8.0	8.0	7.0	...	NaN	NaN	N
Y	5.0	4.0	4.0	6.0	6.0	6.0	6.0	6.0	6.0	5.0	...	NaN	NaN	N
month	mar	oct	oct	mar	mar	aug	aug	aug	sep	sep	...	NaN	NaN	N
day	fri	tue	sat	fri	sun	sun	mon	mon	tue	sat	...	NaN	NaN	N
FFMC	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	91.0	91.0	9
DMC	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	166.9	166.9	16
DC	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	752.6	752.6	75
temp	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	...	25.9	25.9	2

8 rows × 1034 columns

In [25]:
```
df.drop(columns='FFMC')
```

Out[25]:

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

517 rows × 12 columns

In [26]:
```
df_merged=pd.concat([df1,df2])
df_merged
```

Out[26]:

	X	Y	month	day	FFMC	DMC	DC	temp
0	7.0	5.0	mar	fri	NaN	NaN	NaN	NaN
1	7.0	4.0	oct	tue	NaN	NaN	NaN	NaN
2	7.0	4.0	oct	sat	NaN	NaN	NaN	NaN
3	8.0	6.0	mar	fri	NaN	NaN	NaN	NaN
4	8.0	6.0	mar	sun	NaN	NaN	NaN	NaN
...	...	...	...	...	...	...	...	...
512	NaN	NaN	NaN	NaN	81.6	56.7	665.6	27.8
513	NaN	NaN	NaN	NaN	81.6	56.7	665.6	21.9
514	NaN	NaN	NaN	NaN	81.6	56.7	665.6	21.2
515	NaN	NaN	NaN	NaN	94.4	146.0	614.7	25.6
516	NaN	NaN	NaN	NaN	79.5	3.0	106.7	11.8

1034 rows × 8 columns

In [27]:
```
from sklearn.model_selection import train_test_split
from sklearn import linear_model , metrics
x=df[["X"]]
y=df[["temp"]]
```

In [29]:
```
x_train , x_test , y_train , y_test = train_test_split(x,y,test_size=0.2,random_
```

In [30]:
```
len(x_train)
```

Out[30]: 413

In [31]:
```
len(x_test)
```

Out[31]: 104

In [33]:
```
df.shape
```

Out[33]: (517, 13)

In [34]:
```
reg=linear_model.LinearRegression()
```

In [35]:
```
print(x_train)
```

```
     X
135  3
218  4
119  3
463  6
42   4
..  ..
129  2
144  2
72   5
235  8
37   7

[413 rows x 1 columns]
```

In [36]:
```
model=reg.fit(x_train,y_train)
```

In [37]:
```
r_sq=reg.score(x_train,y_train)
```

In [38]:
```
print("Determination coeff" , r_sq)
```

```
Determination coeff 0.0040996940667922255
```

In [40]:
```
print("intercept" , model.intercept_)
```

```
intercept [19.58351711]
```

In [41]:
```
print("slope" , model.coef_)
```

```
slope [[-0.16678537]]
```

In [ ]:
```
y_predict=model.predict)
```