In [1]:
```
import pandas as pd
import seaborn as sns
import matplotlib.pyplot as plt
d1=pd.read_csv("India Air Quality Data.csv")
d2=pd.read_csv("heart (2).csv")
```

```
C:\Users\RUTIKA\anaconda3\lib\site-packages\IPython\core\interactiveshell.py:
3146: DtypeWarning: Columns (0) have mixed types.Specify dtype option on impo
rt or set low_memory=False.
  has_raised = await self.run_ast_nodes(code_ast.body, cell_name,
```

In [2]:
```
d1
```

Out[2]:

	stn_code	sampling_date	state	location	agency	type	so2	no2	rsp
0	150	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	4.8	17.4	Na
1	151	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	3.1	7.0	Na
2	152	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.2	28.5	Na
3	150	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.3	14.7	Na
4	151	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	4.7	7.5	Na
...	...	...	...	...	...	...	...	...	...
435737	SAMP	24-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	22.0	50.0	143
435738	SAMP	29-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	20.0	46.0	171
435739	NaN	NaN	andaman-and-nicobar-islands	NaN	NaN	NaN	NaN	NaN	Na
435740	NaN	NaN	Lakshadweep	NaN	NaN	NaN	NaN	NaN	Na
435741	NaN	NaN	Tripura	NaN	NaN	NaN	NaN	NaN	Na

435742 rows × 13 columns

In [3]:
```
d2
```

Out[3]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	targ
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
1020	59	1	1	140	221	0	1	164	1	0.0	2	0	2	
1021	60	1	0	125	258	0	0	141	1	2.8	1	1	3	
1022	47	1	0	110	275	0	0	118	1	1.0	1	1	2	
1023	50	0	0	110	254	0	0	159	0	0.0	2	0	2	
1024	54	1	0	120	188	0	1	113	0	1.4	1	1	3	

1025 rows × 14 columns

In [4]:
```
d1.isnull().sum()
```

Out[4]:
```
stn_code                     144077
sampling_date                     3
state                             0
location                          3
agency                       149481
type                           5393
so2                           34646
no2                           16233
rspm                          40222
spm                          237387
location_monitoring_station   27491
pm2_5                        426428
date                              7
dtype: int64
```

In [5]:
```
d1.dropna(thresh=0.3*len(d1),axis=1,inplace=True)
d1
```

Out[5]:

	stn_code	sampling_date	state	location	agency	type	so2	no2	rsp
0	150	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	4.8	17.4	Na
1	151	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	3.1	7.0	Na
2	152	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.2	28.5	Na
3	150	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.3	14.7	Na
4	151	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	4.7	7.5	Na
...	...	...	...	...	...	...	...	...	...
435737	SAMP	24-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	22.0	50.0	143
435738	SAMP	29-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	20.0	46.0	171
435739	NaN	NaN	andaman-and-nicobar-islands	NaN	NaN	NaN	NaN	NaN	Na
435740	NaN	NaN	Lakshadweep	NaN	NaN	NaN	NaN	NaN	Na
435741	NaN	NaN	Tripura	NaN	NaN	NaN	NaN	NaN	Na

435742 rows × 12 columns

In [6]:
```
d1.isnull().sum()
```

Out[6]:
```
stn_code                       144077
sampling_date                       3
state                               0
location                            3
agency                         149481
type                             5393
so2                             34646
no2                             16233
rspm                            40222
spm                            237387
location_monitoring_station     27491
date                                7
dtype: int64
```

In [7]:
```
d2.duplicated().sum()
```

Out[7]:
```
723
```

In [8]:
```
d2.drop_duplicates(inplace=True)
```

In [9]:
```
d2
```

Out[9]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	targe
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	(
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	(
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	(
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	(
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	(
...	...	...	...	...	...	...	...	...	...	...	...	...	...	..
723	68	0	2	120	211	0	0	115	0	1.5	1	0	2	'
733	44	0	2	108	141	0	1	175	0	0.6	1	0	2	'
739	52	1	0	128	255	0	1	161	1	0.0	2	1	3	(
843	59	1	3	160	273	0	0	125	0	0.0	2	0	2	(
878	54	1	0	120	188	0	1	113	0	1.4	1	1	3	(

302 rows × 14 columns

In [10]:
```
d2.duplicated().sum()
```

Out[10]:
```
0
```

In [11]:
```
df1=d2[['age','sex','cp','thal']].loc[0:15]
df1
```

Out[11]:

	age	sex	cp	thal
0	52	1	0	3
1	53	1	0	3
2	70	1	0	3
3	61	1	0	3
4	62	0	0	2
5	58	0	0	2
6	58	1	0	1
7	55	1	0	3
8	46	1	0	3
9	54	1	0	2
10	71	0	0	2
11	43	0	0	3
12	34	0	1	2
13	51	1	0	3
14	52	1	0	0

In [12]:
```
df2=d2[['age','sex','cp','thal']].loc[16:30]
df2
```

Out[12]:

	age	sex	cp	thal
16	51	0	2	2
17	54	1	0	3
18	50	0	1	2
19	58	1	2	2
20	60	1	2	2
21	67	0	0	2
22	45	1	0	2
23	63	0	2	2
24	42	0	2	2
25	61	0	0	3
26	44	1	2	2
27	58	0	1	2
28	56	1	2	1
29	55	0	0	2
30	44	1	0	1

In [13]:
```
merge=pd.merge(df1,df2,on='age',how='inner')
merge
```

Out[13]:

	age	sex_x	cp_x	thal_x	sex_y	cp_y	thal_y
0	61	1	0	3	0	0	3
1	58	0	0	2	1	2	2
2	58	0	0	2	0	1	2
3	58	1	0	1	1	2	2
4	58	1	0	1	0	1	2
5	55	1	0	3	0	0	2
6	54	1	0	2	1	0	3
7	51	1	0	3	0	2	2

In [14]:
```
d2['target']=d2['target'].apply(lambda  x :1 if x>0 else 0)
```

In [15]:
```
d2
```

Out[15]:

	age	sex	cp	trestbps	chol	fbs	restecg	thalach	exang	oldpeak	slope	ca	thal	targe
0	52	1	0	125	212	0	1	168	0	1.0	2	2	3	(
1	53	1	0	140	203	1	0	155	1	3.1	0	0	3	(
2	70	1	0	145	174	0	1	125	1	2.6	0	0	3	(
3	61	1	0	148	203	0	1	161	0	0.0	2	1	3	(
4	62	0	0	138	294	1	1	106	0	1.9	1	3	2	(
...	...	...	...	...	...	...	...	...	...	...	...	...	...	..
723	68	0	2	120	211	0	0	115	0	1.5	1	0	2	'
733	44	0	2	108	141	0	1	175	0	0.6	1	0	2	'
739	52	1	0	128	255	0	1	161	1	0.0	2	1	3	(
843	59	1	3	160	273	0	0	125	0	0.0	2	0	2	(
878	54	1	0	120	188	0	1	113	0	1.4	1	1	3	(

302 rows × 14 columns

In [16]:
```
del d1['rspm']
d1
```

Out[16]:

	stn_code	sampling_date	state	location	agency	type	so2	no2	spr
0	150	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	4.8	17.4	Na
1	151	February - M021990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	3.1	7.0	Na
2	152	February - M021990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.2	28.5	Na
3	150	March - M031990	Andhra Pradesh	Hyderabad	NaN	Residential, Rural and other Areas	6.3	14.7	Na
4	151	March - M031990	Andhra Pradesh	Hyderabad	NaN	Industrial Area	4.7	7.5	Na
...	...	...	...	...	...	...	...	...	...
435737	SAMP	24-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	22.0	50.0	Na
435738	SAMP	29-12-15	West Bengal	ULUBERIA	West Bengal State Pollution Control Board	RIRUO	20.0	46.0	Na
435739	NaN	NaN	andaman-and-nicobar-islands	NaN	NaN	NaN	NaN	NaN	Na
435740	NaN	NaN	Lakshadweep	NaN	NaN	NaN	NaN	NaN	Na
435741	NaN	NaN	Tripura	NaN	NaN	NaN	NaN	NaN	Na

435742 rows × 11 columns

In [17]:
```
from sklearn.model_selection import train_test_split
x= merge.drop(['age'],axis=1)
x
```

Out[17]:

	sex_x	cp_x	thal_x	sex_y	cp_y	thal_y
0	1	0	3	0	0	3
1	0	0	2	1	2	2
2	0	0	2	0	1	2
3	1	0	1	1	2	2
4	1	0	1	0	1	2
5	1	0	3	0	0	2
6	1	0	2	1	0	3
7	1	0	3	0	2	2

In [19]:
```
y=merge['thal_y']
y
```

Out[19]:
```
0    3
1    2
2    2
3    2
4    2
5    2
6    3
7    2
Name: thal_y, dtype: int64
```

In [21]:
```
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.3,random_state=
```

In [22]:
```
from sklearn.linear_model import LogisticRegression
logreg=LogisticRegression()
```

In [23]:
```
logreg.fit(x_train,y_train)
```

Out[23]:
```
LogisticRegression()
```

In [24]:
```
from sklearn.metrics import classification_report,confusion_matrix
y_pred=logreg.predict(x_test)
```

In [26]:
```
print(confusion_matrix(y_test,y_pred))
print(classification_report(y_test,y_pred))
```

```
[[2 0]
 [1 0]]
              precision    recall  f1-score   support

           2       0.67      1.00      0.80         2
           3       0.00      0.00      0.00         1

    accuracy                           0.67         3
   macro avg       0.33      0.50      0.40         3
weighted avg       0.44      0.67      0.53         3


C:\Users\RUTIKA\anaconda3\lib\site-packages\sklearn\metrics\_classification.p
y:1221: UndefinedMetricWarning: Precision and F-score are ill-defined and bei
ng set to 0.0 in labels with no predicted samples. Use `zero_division` parame
ter to control this behavior.
  _warn_prf(average, modifier, msg_start, len(result))
```

In [ ]: