Perform the following operations using Python on the Air quality and Heart Diseases data sets

a. Data cleaning b. Data integration c. Data transformation d. Error correcting e. Data model building

In [1]:
```
import pandas as pd
import numpy as np
import matplotlib.pyplot as pyplot
# Import the Python machine learning libraries we need
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
from sklearn.metrics import accuracy_score
```

In [2]:
```
import warnings
warnings.filterwarnings('ignore')
df = pd.read_csv('AirQuality.csv', sep = ';')
df
```

Out[2]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	1559.0	972.0
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	1555.0	1074.0
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	1584.0	1203.0
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	1490.0	1110.0
...	...	...	...	...	...	...	...	...	...	...	...	...
9466	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9467	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9468	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9469	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)
9470	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

9471 rows × 17 columns

In [3]:
```
df.shape
```

Out[3]:
```
(9471, 17)
```

In [4]:
```
df.isnull()
```

Out[4]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T	I
0	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
1	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
2	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
3	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
4	False	False	False	False	False	False	False	False	False	False	False	False	False	Fa
...	...	...	...	...	...	...	...	...	...	...	...	...	...	
9466	True	True	True	True	True	True	True	True	True	True	True	True	True	Tr
9467	True	True	True	True	True	True	True	True	True	True	True	True	True	Tr
9468	True	True	True	True	True	True	True	True	True	True	True	True	True	Tr
9469	True	True	True	True	True	True	True	True	True	True	True	True	True	Tr
9470	True	True	True	True	True	True	True	True	True	True	True	True	True	Tr

9471 rows × 17 columns

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

Out[6]:
```
(9471, 17)
```

In [5]:
```
df.isnull().sum()
```

Out[5]:
```
Date             114
Time             114
CO(GT)           114
PT08.S1(CO)      114
NMHC(GT)         114
C6H6(GT)         114
PT08.S2(NMHC)    114
NOx(GT)          114
PT08.S3(NOx)     114
NO2(GT)          114
PT08.S4(NO2)     114
PT08.S5(O3)      114
T                114
RH               114
AH               114
Unnamed: 15     9471
Unnamed: 16     9471
dtype: int64
```

In [8]:
```
# Gets rows numbered from 0 to 4
df1 = df.loc[0:4]
df1
```

Out[8]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13,6
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	1559.0	972.0	13,3
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	1555.0	1074.0	11,9
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	1584.0	1203.0	11,0
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	1490.0	1110.0	11,2

In [10]:
```
df1.isnull().sum()
```

Out[10]:
```
Date             0
Time             0
CO(GT)           0
PT08.S1(CO)      0
NMHC(GT)         0
C6H6(GT)         0
PT08.S2(NMHC)    0
NOx(GT)          0
PT08.S3(NOx)     0
NO2(GT)          0
PT08.S4(NO2)     0
PT08.S5(O3)      0
T                0
RH               0
AH               0
Unnamed: 15      5
Unnamed: 16      5
dtype: int64
```

In [11]:
```
# Does the same thing as isnull()
df1.isna().any()
```

Out[11]:
```
Date             False
Time             False
CO(GT)           False
PT08.S1(CO)      False
NMHC(GT)         False
C6H6(GT)         False
PT08.S2(NMHC)    False
NOx(GT)          False
PT08.S3(NOx)     False
NO2(GT)          False
PT08.S4(NO2)     False
PT08.S5(O3)      False
T                False
RH               False
AH               False
Unnamed: 15       True
```

```
Unnamed: 16       True
dtype: bool
```

In [12]:
```
df1.drop_duplicates(subset=['Unnamed: 15', 'Unnamed: 16'])
```

Out[12]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13,6

In [13]:
```
df1.duplicated().sum()
```

Out[13]:
```
0
```

In [14]:
```
df1.columns
```

Out[14]:
```
Index(['Date', 'Time', 'CO(GT)', 'PT08.S1(CO)', 'NMHC(GT)', 'C6H6(GT)',
       'PT08.S2(NMHC)', 'NOx(GT)', 'PT08.S3(NOx)', 'NO2(GT)', 'PT08.S4(NO2)',
       'PT08.S5(O3)', 'T', 'RH', 'AH', 'Unnamed: 15', 'Unnamed: 16'],
      dtype='object')
```

In [15]:
```
df1 = df.loc[1:4,['C6H6(GT)','PT08.S2(NMHC)']]
df1
```

Out[15]:

	C6H6(GT)	PT08.S2(NMHC)
1	9,4	955.0
2	9,0	939.0
3	9,2	948.0
4	6,5	836.0

In [16]:
```
df2=df.loc[9466:9470,['C6H6(GT)','PT08.S2(NMHC)']]
df2
```

Out[16]:

	C6H6(GT)	PT08.S2(NMHC)
9466	NaN	NaN
9467	NaN	NaN
9468	NaN	NaN
9469	NaN	NaN
9470	NaN	NaN

In [17]:

```
#merge two data frames with concat function
merged = pd.concat([df1, df2])
merged
```

Out[17]:

	C6H6(GT)	PT08.S2(NMHC)
1	9,4	955.0
2	9,0	939.0
3	9,2	948.0
4	6,5	836.0
9466	NaN	NaN
9467	NaN	NaN
9468	NaN	NaN
9469	NaN	NaN
9470	NaN	NaN

In [18]:

```
df1 = df.loc[0:4]
df1
```

Out[18]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13,6

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	1559.0	972.0	13,3
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	1555.0	1074.0	11,9
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	1584.0	1203.0	11,0
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	1490.0	1110.0	11,2

In [19]:
```
df.melt()
```

Out[19]:

	variable	value
0	Date	10/03/2004
1	Date	10/03/2004
2	Date	10/03/2004
3	Date	10/03/2004
4	Date	10/03/2004
...	...	...
161002	Unnamed: 16	NaN
161003	Unnamed: 16	NaN
161004	Unnamed: 16	NaN
161005	Unnamed: 16	NaN
161006	Unnamed: 16	NaN

161007 rows × 2 columns

In [20]:
```
df1["Unnamed: 15"].fillna("mean")
```

Out[20]:
```
0    mean
1    mean
2    mean
3    mean
4    mean
Name: Unnamed: 15, dtype: object
```

In [21]:
```
df1["Unnamed: 15"] = df1["Unnamed: 15"].fillna("mean")
df1
```

Out[21]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13,6
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	1559.0	972.0	13,3
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	1555.0	1074.0	11,9
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	1584.0	1203.0	11,0
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	1490.0	1110.0	11,2

In [22]:
```
df1["Unnamed: 16"].fillna(df1["Unnamed: 16"].mean() , inplace= True)
df1
```

Out[22]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13,6
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	1559.0	972.0	13,3
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	1555.0	1074.0	11,9
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	1584.0	1203.0	11,0
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	1490.0	1110.0	11,2

In [25]:
```
df["PT08.S4(NO2)"].fillna(df["PT08.S4(NO2)"].mean(), inplace=True)
df
```

Out[25]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	NaN	1268.0
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	NaN	972.0
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	NaN	1074.0
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	NaN	1203.0
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	NaN	1110.0
...	...	...	...	...	...	...	...	...	...	...	...	...
9466	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9467	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9468	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9469	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN
9470	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN	NaN

9471 rows × 17 columns

In [26]:
```
df1
```

Out[26]:

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
0	10/03/2004	18.00.00	2,6	1360.0	150.0	11,9	1046.0	166.0	1056.0	113.0	1692.0	1268.0	13,6
1	10/03/2004	19.00.00	2	1292.0	112.0	9,4	955.0	103.0	1174.0	92.0	1559.0	972.0	13,3
2	10/03/2004	20.00.00	2,2	1402.0	88.0	9,0	939.0	131.0	1140.0	114.0	1555.0	1074.0	11,9
3	10/03/2004	21.00.00	2,2	1376.0	80.0	9,2	948.0	172.0	1092.0	122.0	1584.0	1203.0	11,0

	Date	Time	CO(GT)	PT08.S1(CO)	NMHC(GT)	C6H6(GT)	PT08.S2(NMHC)	NOx(GT)	PT08.S3(NOx)	NO2(GT)	PT08.S4(NO2)	PT08.S5(O3)	T
4	10/03/2004	22.00.00	1,6	1272.0	51.0	6,5	836.0	131.0	1205.0	116.0	1490.0	1110.0	11,2

In [ ]:

```
```

In [27]:

```
y = df1["Date"]
X = df1[["NO2(GT)","PT08.S5(O3)"]]
X.head()
```

Out[27]:

	NO2(GT)	PT08.S5(O3)
0	113.0	1268.0
1	92.0	972.0
2	114.0	1074.0
3	122.0	1203.0
4	116.0	1110.0

In [29]:

```
test_size = 0.33
seed = 7
X_train, X_test, y_train, y_test = train_test_split(X, y, test_size=test_size, random_state=None, shuffle = None)
```

In [30]:

```
X_train
```

Out[30]:

	NO2(GT)	PT08.S5(O3)
0	113.0	1268.0
2	114.0	1074.0
1	92.0	972.0

In [31]:
```
X_test
```

Out[31]:

	NO2(GT)	PT08.S5(O3)
3	122.0	1203.0
4	116.0	1110.0

In [32]:
```
y_train
```

Out[32]:
```
0    10/03/2004
2    10/03/2004
1    10/03/2004
Name: Date, dtype: object
```

In [33]:
```
y_test
```

Out[33]:
```
3    10/03/2004
4    10/03/2004
Name: Date, dtype: object
```

In [34]:
```
model = DecisionTreeClassifier()
```

In [35]:
```
model.fit(X_train, y_train)
```

Out[35]:
```
DecisionTreeClassifier()
```

In [36]:
```
model
```

Out[36]:
```
DecisionTreeClassifier()
```

In [ ]:
```

```