In [20]:

```
import numpy as np
import pandas as pd
import seaborn as sns
from sklearn.model_selection import train_test_split
from sklearn.tree import DecisionTreeClassifier
```

In [21]:

```
df=pd.read_csv(r"C:\Users\user\Downloads\drug200.csv")
df
```

Out[21]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

In [22]:

```
df.head()
```

Out[22]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	HIGH	HIGH	25.355	drugY
1	47	M	LOW	HIGH	13.093	drugC
2	47	M	LOW	HIGH	10.114	drugC
3	28	F	NORMAL	HIGH	7.798	drugX
4	61	F	LOW	HIGH	18.043	drugY

In [23]:

```
df.tail()
```

Out[23]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
195	56	F	LOW	HIGH	11.567	drugC
196	16	M	LOW	HIGH	12.006	drugC
197	52	M	NORMAL	HIGH	9.894	drugX
198	23	M	NORMAL	NORMAL	14.020	drugX
199	40	F	LOW	NORMAL	11.349	drugX

In [24]:

```
df.shape
```

Out[24]:

```
(200, 6)
```

In [25]:

```
df.isnull().sum()
```

Out[25]:

```
Age            0
Sex            0
BP             0
Cholesterol    0
Na_to_K        0
Drug           0
dtype: int64
```

In [26]:

```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 200 entries, 0 to 199
Data columns (total 6 columns):
 #   Column       Non-Null Count  Dtype  
---  ------       --------------  -----  
 0   Age          200 non-null    int64  
 1   Sex          200 non-null    object 
 2   BP           200 non-null    object 
 3   Cholesterol  200 non-null    object 
 4   Na_to_K      200 non-null    float64
 5   Drug         200 non-null    object 
dtypes: float64(1), int64(1), object(4)
memory usage: 9.5+ KB
```

In [27]:

```
df['Drug'].value_counts()
```

Out[27]:

```
Drug
drugY    91
drugX    54
drugA    23
drugC    16
drugB    16
Name: count, dtype: int64
```

In [28]:

```
df['BP'].value_counts()
```

Out[28]:

```
BP
HIGH      77
LOW       64
NORMAL    59
Name: count, dtype: int64
```

In [29]:

```
convert={'Sex':{"M":0,"F":1}}
df=df.replace(convert)
df
```

Out[29]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	1	HIGH	HIGH	25.355	drugY
1	47	0	LOW	HIGH	13.093	drugC
2	47	0	LOW	HIGH	10.114	drugC
3	28	1	NORMAL	HIGH	7.798	drugX
4	61	1	LOW	HIGH	18.043	drugY
...	...	...	...	...	...	...
195	56	1	LOW	HIGH	11.567	drugC
196	16	0	LOW	HIGH	12.006	drugC
197	52	0	NORMAL	HIGH	9.894	drugX
198	23	0	NORMAL	NORMAL	14.020	drugX
199	40	1	LOW	NORMAL	11.349	drugX

200 rows × 6 columns

In [30]:

```
convert={'Cholesterol':{"HIGH":1,"NORMAL":0}}
df=df.replace(convert)
df
```

Out[30]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	1	HIGH	1	25.355	drugY
1	47	0	LOW	1	13.093	drugC
2	47	0	LOW	1	10.114	drugC
3	28	1	NORMAL	1	7.798	drugX
4	61	1	LOW	1	18.043	drugY
...	...	...	...	...	...	...
195	56	1	LOW	1	11.567	drugC
196	16	0	LOW	1	12.006	drugC
197	52	0	NORMAL	1	9.894	drugX
198	23	0	NORMAL	0	14.020	drugX
199	40	1	LOW	0	11.349	drugX

200 rows × 6 columns

In [31]:

```
X=["Age","Sex","BP","Cholesterol"]
Y=["drugY","drugC","drugX","drugA","drugB"]
all_inputs=df[X]
all_classes=df["Drug"]
```

In [32]:

```
(X_train,X_test,Y_train,Y_test)=train_test_split(all_inputs,all_classes,test_si
```

In [33]:

```
clf=DecisionTreeClassifier(random_state=0)
```

In [ ]: