

In [1]:

```

1 import numpy as np
2 import pandas as pd
3 import seaborn as sns
4 from sklearn.model_selection import train_test_split
5 from sklearn.tree import DecisionTreeClassifier

```

In [2]:

```

1 df=pd.read_csv(r"C:\Users\91955\Downloads\drug200.csv")
2 df

```

Out[2]:

	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 [3]:

```
1 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 [4]:

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

Out[4]:

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

In [5]:

```
1 df['Cholesterol'].value_counts()
```

Out[5]:

Cholesterol
HIGH 103
NORMAL 97
Name: count, dtype: int64

In [6]:

```
1 convert={'BP':{'LOW':1,"NORMAL":2,"HIGH":3}}  
2 df=df.replace(convert)  
3 df
```

Out[6]:

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

200 rows × 6 columns

In [12]:

```
1 convert={'Drug':{'drugX':1,"drugY":2,"drugA":3,"drugB":4,"drugC":5}}
2 df=df.replace(convert)
3 df
```

Out[12]:

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

200 rows × 6 columns

In [29]:

```
1 x=["Drug","BP"]
2 y=["M","F"]
3 all_inputs=df[x]
4 all_classes=df["Sex"]
```

In [30]:

```
1 (x_train,x_test,y_train,y_test)=train_test_split(all_inputs,all_classes,test_size=0.
```

In [31]:

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

In [32]:

```
1 clf.fit(x_train,y_train)
```

Out[32]:

▼

DecisionTreeClassifier

DecisionTreeClassifier(random_state=0)

In [33]:

```
1 score=clf.score(x_test,y_test)
2 print(score)
```

0.46

In []:

1