In [1]:
```
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 [2]:
```
df=pd.read_csv(r"C:\Users\my pc\downloads\drug200.csv")
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]:
```
df.head()
```

Out[3]:

	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 [4]:
```
df.tail()
```

Out[4]:

	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 [5]:
```
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 [8]:
```
df.isna().any()
```

Out[8]:
```
Age            False
Sex            False
BP             False
Cholesterol    False
Na_to_K        False
Drug           False
dtype: bool
```

In [9]:
```
df["Drug"].value_counts()
```

Out[9]:
```
Drug
drugY    91
drugX    54
drugA    23
drugC    16
drugB    16
Name: count, dtype: int64
```

In [10]:
```
df["BP"].value_counts()
```

Out[10]:
```
BP
HIGH      77
LOW       64
NORMAL    59
Name: count, dtype: int64
```

In [11]:
```
df["Sex"].value_counts()
```

Out[11]:
```
Sex
M    104
F     96
Name: count, dtype: int64
```

In [13]:
```
df["Cholesterol"].value_counts()
```

Out[13]:
```
Cholesterol
HIGH      103
NORMAL     97
Name: count, dtype: int64
```

In [14]:
```
c={"Cholesterol":{"HIGH":1,"NORMAL":0}}
df=df.replace(c)
df
```

Out[14]:

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

200 rows × 6 columns

In [15]:
```
c={"BP":{"HIGH":1,"LOW":2,"NORMAL":3}}
df=df.replace(c)
df
```

Out[15]:

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

200 rows × 6 columns

In [16]:
```
c={"Sex":{"F":1,"M":0}}
df=df.replace(c)
df
```

Out[16]:

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

200 rows × 6 columns

In [17]:
```
c={"Drug":{"drugX":1,"drugY":2,"drugA":3,"drugB":4,"drugC":5}}
df=df.replace(c)
df
```

Out[17]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	1	1	1	25.355	2
1	47	0	2	1	13.093	5
2	47	0	2	1	10.114	5
3	28	1	3	1	7.798	1
4	61	1	2	1	18.043	2
...	...	...	...	...	...	...
195	56	1	2	1	11.567	5
196	16	0	2	1	12.006	5
197	52	0	3	1	9.894	1
198	23	0	3	0	14.020	1
199	40	1	2	0	11.349	1

200 rows × 6 columns

In [18]:
```
x=["Age","Sex","BP","Cholesterol","Na_to_K"]
y=["drugX","drugY","drugA","drugB","drugC"]
all_inputs=df[x]
all_classes=df["Drug"]
```

In [19]:
```
x_train,x_test,y_train,y_test=train_test_split(all_inputs,all_classes,test_size=0.25)
```

In [20]:
```
clf=DecisionTreeClassifier(random_state=0)
```

In [21]:
```
clf.fit(x_train,y_train)
```

Out[21]:
```
▾ DecisionTreeClassifier
DecisionTreeClassifier(random_state=0)
```

In [23]:
```
score=clf.score(x_test,y_test)
```

In [24]:
```
print(score)
```

```
0.94
```

In [25]:
```
clf.score(x_train,y_train)
```

Out[25]:
```
1.0
```