

```
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
        6
```

```
In [2]: 1 df=pd.read_csv(r"C:\Users\P. VIJAY KUMAR\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['Sex'].value_counts()
        2
```

```
Out[4]: Sex
M      104
F       96
Name: count, dtype: int64
```

```
In [5]: 1 df['BP'].value_counts()
        2
```

```
Out[5]: BP
HIGH      77
LOW       64
NORMAL    59
Name: count, dtype: int64
```

```
In [6]: 1 df['Cholesterol'].value_counts()
        2
```

```
Out[6]: Cholesterol
HIGH      103
NORMAL     97
Name: count, dtype: int64
```

```
In [7]: 1 df['Drug'].value_counts()
```

```
Out[7]: Drug
drugY      91
drugX      54
drugA      23
drugC      16
drugB      16
Name: count, dtype: int64
```

```
In [8]: 1 convert={"Sex":{"M":1,"F":0}}
        2 df=df.replace(convert)
        3 df
```

Out[8]:

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

200 rows × 6 columns

```
In [9]: 1 convert={"BP":{"HIGH":1,"NORMAL":2,"LOW":3}}
        2 df=df.replace(convert)
        3 df
```

Out[9]:

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

200 rows × 6 columns

```
In [10]: 1 convert={"Cholesterol":{"HIGH":1,"NORMAL":2}}
2 df=df.replace(convert)
3 df
```

Out[10]:

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

200 rows × 6 columns

```
In [11]: 1 x=["Sex","BP","Cholesterol"]
2 y=["DrugY","DrugX","DrugA","DrugC","DrugB"]
3 all_inputs=df[x]
4 all_classes=df["Drug"]
5
```

```
In [12]: 1 x_train,x_test,y_train,y_test=train_test_split(all_inputs,all_classes,tes
2 clf=DecisionTreeClassifier(random_state=0)
3 clf.fit(x_train,y_train)
4
```

Out[12]:

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

```
In [13]: 1 score=clf.score(x_test,y_test)
2 print(score)
```

0.36666666666666664

```
In [ ]: 1
```

