

```
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\LENOVO\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.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]: df['BP'].value_counts()
df['Na_to_K'].value_counts()

Out[4]: Na_to_K

12.006	2
18.295	2
25.355	1
11.939	1
16.347	1
..	
24.658	1
24.276	1
13.967	1
19.675	1
11.349	1

Name: count, Length: 198, dtype: int64

```
In [5]: convert={"BP":{"HIGH":129,"NORMAL":80,"LOW":50}}
df=df.replace(convert)
df
```

Out[5]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	129	HIGH	25.355	drugY
1	47	M	50	HIGH	13.093	drugC
2	47	M	50	HIGH	10.114	drugC
3	28	F	80	HIGH	7.798	drugX
4	61	F	50	HIGH	18.043	drugY
...
195	56	F	50	HIGH	11.567	drugC
196	16	M	50	HIGH	12.006	drugC
197	52	M	80	HIGH	9.894	drugX
198	23	M	80	NORMAL	14.020	drugX
199	40	F	50	NORMAL	11.349	drugX

200 rows × 6 columns

```
In [6]: convert={"Cholesterol":{"HIGH":103,"NORMAL":97}}
df=df.replace(convert)
df
```

Out[6]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	129	103	25.355	drugY
1	47	M	50	103	13.093	drugC
2	47	M	50	103	10.114	drugC
3	28	F	80	103	7.798	drugX
4	61	F	50	103	18.043	drugY
...
195	56	F	50	103	11.567	drugC
196	16	M	50	103	12.006	drugC
197	52	M	80	103	9.894	drugX
198	23	M	80	97	14.020	drugX
199	40	F	50	97	11.349	drugX

200 rows × 6 columns

```
In [7]: convert={"Cholesterol":{"HIGH":103,"NORMAL":97}}
df=df.replace(convert)
df
```

Out[7]:

	Age	Sex	BP	Cholesterol	Na_to_K	Drug
0	23	F	129	103	25.355	drugY
1	47	M	50	103	13.093	drugC
2	47	M	50	103	10.114	drugC
3	28	F	80	103	7.798	drugX
4	61	F	50	103	18.043	drugY
...
195	56	F	50	103	11.567	drugC
196	16	M	50	103	12.006	drugC
197	52	M	80	103	9.894	drugX
198	23	M	80	97	14.020	drugX
199	40	F	50	97	11.349	drugX

200 rows × 6 columns

```
In [8]: x=["Age","BP","Cholesterol","Na_to_K"]
y=["drugY","drugC"]
all_inputs=df[x]
all_classes=df["Drug"]
```

```
In [9]: (x_train,x_test,y_train,y_test)=train_test_split(all_inputs,all_classes,test_size=0.3)
```

```
In [10]: clf=DecisionTreeClassifier(random_state=0)
```

```
In [11]: clf.fit(x_train,y_train)
```

```
Out[11]: ▾      DecisionTreeClassifier  
         DecisionTreeClassifier(random_state=0)
```

```
In [12]: score=clf.score(x_test,y_test)  
         print(score)
```

```
1.0
```

```
In [ ]:
```