In [49]:

```
import pandas as pd
import numpy as np
from sklearn.preprocessing import LabelEncoder
ds=pd.read_csv("play_tennis.csv")
ds
```

Out[49]:

	outlook	temp	humidity	wind	play
0	Sunny	Hot	High	Weak	No
1	Sunny	Hot	High	Strong	No
2	Overcast	Hot	High	Weak	Yes
3	Rain	Mild	High	Weak	Yes
4	Rain	Cool	Normal	Weak	Yes
5	Rain	Cool	Normal	Strong	No
6	Overcast	Cool	Normal	Strong	Yes
7	Sunny	Mild	High	Weak	No
8	Sunny	Cool	Normal	Weak	Yes
9	Rain	Mild	Normal	Weak	Yes
10	Sunny	Mild	Normal	Strong	Yes
11	Overcast	Mild	High	Strong	Yes
12	Overcast	Hot	Normal	Weak	Yes
13	Rain	Mild	High	Strong	No

In [50]:

```
from sklearn import preprocessing
label_encoder=preprocessing.LabelEncoder()
ds['outlook']=label_encoder.fit_transform(ds['outlook'])
ds['outlook'].values
ds['temp']=label_encoder.fit_transform(ds['temp'])
ds['temp'].values
ds['humidity']=label_encoder.fit_transform(ds['humidity'])
ds['humidity'].values
ds['wind']=label_encoder.fit_transform(ds['wind'])
ds['wind'].values
ds
```

Out[50]:

	outlook	temp	humidity	wind	play
0	2	1	0	1	No
1	2	1	0	0	No
2	0	1	0	1	Yes
3	1	2	0	1	Yes
4	1	0	1	1	Yes
5	1	0	1	0	No
6	0	0	1	0	Yes
7	2	2	0	1	No
8	2	0	1	1	Yes
9	1	2	1	1	Yes
10	2	2	1	0	Yes
11	0	2	0	0	Yes
12	0	1	1	1	Yes
13	1	2	0	0	No

In [51]:

```
x=ds.drop(['play'], axis=1)
x
```

Out[51]:

	outlook	temp	humidity	wind
0	2	1	0	1
1	2	1	0	0
2	0	1	0	1
3	1	2	0	1
4	1	0	1	1
5	1	0	1	0
6	0	0	1	0
7	2	2	0	1
8	2	0	1	1
9	1	2	1	1
10	2	2	1	0
11	0	2	0	0
12	0	1	1	1
13	1	2	0	0

In [52]:

```
y=ds['play']
```

```
y
```

Out[52]:

```
0      No
1      No
2     Yes
3     Yes
4     Yes
5      No
6     Yes
7      No
8     Yes
9     Yes
10    Yes
11    Yes
12    Yes
13     No
Name: play, dtype: object
```

In [53]:

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.2,random_state=100)
x_test.shape
```

Out[53]:

```
(3, 4)
```

In [54]:

```
x_train.shape
```

Out[54]:

```
(11, 4)
```

In [55]:

```
x_train
```

Out[55]:

	outlook	temp	humidity	wind
1	2	1	0	0
9	1	2	1	1
4	1	0	1	1
6	0	0	1	0
2	0	1	0	1
0	2	1	0	1
10	2	2	1	0
7	2	2	0	1
3	1	2	0	1
13	1	2	0	0
8	2	0	1	1

In [56]:

```
from sklearn.naive_bayes import GaussianNB
gnb = GaussianNB()
gnb.fit(x_train, y_train)
```

Out[56]:

```
GaussianNB()
```

In [58]:

```
y_pred = gnb.predict(x_test)
y_pred
```

Out[58]:

```
array(['No', 'Yes', 'Yes'], dtype='<U3')
```

In [59]:

```
from sklearn import metrics
print("Gaussian Naive Bayes model accuracy(in %):", metrics.accuracy_score(y_test, y_p
```

```
Gaussian Naive Bayes model accuracy(in %): 33.33333333333333
```

In [64]:

```
x=[[2,1,0,0]]
y_pred=gnb.predict(x)
print('palytennis',y_pred)
```

```
palytennis ['No']
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X does n
ot have valid feature names, but GaussianNB was fitted with feature names
  warnings.warn(
```

In [65]:

```
x=[[0,0,1,1]]
y_pred=gnb.predict(x)
print('palytennis',y_pred)
```

```
palytennis ['Yes']
```

```
C:\ProgramData\Anaconda3\lib\site-packages\sklearn\base.py:450: UserWarning: X does n
ot have valid feature names, but GaussianNB was fitted with feature names
  warnings.warn(
```