```
from google.colab import drive
drive.mount('/content/drive')
```

```
Mounted at /content/drive
```

```
import pandas as pd
from sklearn.preprocessing import LabelEncoder
df=pd.read_csv("/content/drive/MyDrive/Mall_Customers.csv")
df
```

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40
...	...	...	...	...	...
195	196	Female	35	120	79
196	197	Female	45	126	28
197	198	Male	32	126	74
198	199	Male	32	137	18
199	200	Male	30	137	83

200 rows × 5 columns

```
df.head()
```

	CustomerID	Genre	Age	Annual Income (k$)	Spending Score (1-100)
0	1	Male	19	15	39
1	2	Male	21	15	81
2	3	Female	20	16	6
3	4	Female	23	16	77
4	5	Female	31	17	40

```
x=df.iloc[:,1:4]
x.head()
```

	Genre	Age	Annual Income (k$)
0	Male	19	15
1	Male	21	15
2	Female	20	16
3	Female	23	16
4	Female	31	17

```
y=df['Spending Score (1-100)']
y
```

```
0      39
1      81
2       6
3      77
4      40
       ..
195    79
196    28
197    74
198    18
199    83
Name: Spending Score (1-100), Length: 200, dtype: int64
```

```
from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
```

```
x.Genre=le.fit_transform(x.Genre)
x.head()
```

	Genre	Age	Annual Income (k$)
0	1	19	15
1	1	21	15
2	0	20	16
3	0	23	16
4	0	31	17

```
from sklearn.preprocessing import MinMaxScaler
ms=MinMaxScaler()
x_scaled=pd.DataFrame(ms.fit_transform(x),columns=x.columns)
```

```
x_scaled
```

	Genre	Age	Annual Income (k$)
0	1.0	0.019231	0.000000
1	1.0	0.057692	0.000000
2	0.0	0.038462	0.008197
3	0.0	0.096154	0.008197
4	0.0	0.250000	0.016393
...	...	...	...
195	0.0	0.326923	0.860656
196	0.0	0.519231	0.909836
197	1.0	0.269231	0.909836
198	1.0	0.269231	1.000000
199	1.0	0.230769	1.000000

200 rows × 3 columns

```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x_scaled,y,test_size=0.2,random_state=0)
```

```
x_train.shape,x_test.shape,y_train.shape,y_test.shape
```

```
((160, 3), (40, 3), (160,), (40,))
```

```
from sklearn.naive_bayes import GaussianNB
model=GaussianNB()
```

```
model.fit(x_train,y_train)
```

```
GaussianNB()
```

```
pred=model.predict(x_test)
pred
```

```
array([92, 10, 10, 92, 10, 92,  6, 10, 42, 40, 92, 40, 72, 40, 46, 77, 92,
       10, 72, 46, 86, 10, 42, 78, 51, 92, 92, 92, 40, 42, 72, 92, 10, 57,
       42, 72, 10,  6, 42, 57])
```

```
y_test
```

```
18     29
170    13
107    46
98     42
177    69
182    15
5      76
146    36
12     15
152    20
61     55
```

```
125    77
180    32
154    16
80     51
7      94
33     92
130     9
37     73
74     47
183    88
145    97
45     65
159    73
60     56
123    91
179    90
185    97
122    58
44     28
16     35
55     41
150    17
111    54
22      5
189    85
129    75
4      40
83     44
106    50
Name: Spending Score (1-100), dtype: int64
```

```
model.predict(ms.transform([[3,20,74000]]))
```

```
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but MinMaxScaler was
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/sklearn/base.py:439: UserWarning: X does not have valid feature names, but GaussianNB was f
  warnings.warn(
array([79])
```