In [1]:
```
import pandas as pd
import numpy as np
import seaborn as sns
import matplotlib.pyplot as plt
```

In [2]:
```
mydata=pd.read_excel("mobile_price.csv.xlsx")
```

In [3]:
```
mydata.isnull().sum()
```

Out[3]:
```
battery_power    0
blue             0
clock_speed      0
dual_sim         0
fc               0
four_g           0
int_memory       0
m_dep            0
mobile_wt        0
n_cores          0
pc               0
px_height        0
px_width         0
ram              0
sc_h             0
sc_w             0
talk_time        0
three_g          0
touch_screen     0
wifi             0
price_range      0
dtype: int64
```

In [4]:
```
mydata
```

Out[4]:

fc	four_g	int_memory	m_dep	mobile_wt	n_cores	...	px_height	px_width	ram	sc_h	sc_w	talk_time	three_g	touch_screen	wifi	price_range
1	0	7	0.6	188	2	...	20	756	2549	9	7	19	0	0	1	1
0	1	53	0.7	136	3	...	905	1988	2631	17	3	7	1	1	0	2
2	1	41	0.9	145	5	...	1263	1716	2603	11	2	9	1	1	0	2
0	0	10	0.8	131	6	...	1216	1786	2769	16	8	11	1	0	0	2
13	1	44	0.6	141	2	...	1208	1212	1411	8	2	15	1	1	0	1
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
0	1	2	0.8	106	6	...	1222	1890	668	13	4	19	1	1	0	0
0	0	39	0.2	187	4	...	915	1965	2032	11	10	16	1	1	1	2
1	1	36	0.7	108	8	...	868	1632	3057	9	1	5	1	1	0	3
4	1	46	0.1	145	5	...	336	670	869	18	10	19	1	1	1	0
5	1	45	0.9	168	6	...	483	754	3919	19	4	2	1	1	1	3

In [5]:
```
x_ind=mydata.drop(["price_range"],axis=1)
y_dep=mydata.price_range
```

In [6]:
```
from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x_ind,y_dep,train_size=0.8,random_state=2)
```

In [7]:
```
from sklearn.naive_bayes import GaussianNB
model=GaussianNB()
y_pred=model.fit(x_train,y_train).predict(x_test)
```

In [8]:
```
from sklearn.metrics import confusion_matrix,accuracy_score
```

In [10]:
```
confusion_matrix(y_pred,y_test)
```

Out[10]:
```
array([[94, 16,  0,  0],
       [ 4, 80, 17,  0],
       [ 0, 13, 67,  4],
       [ 0,  0, 10, 95]], dtype=int64)
```

In [9]:
```
accuracy_score(y_pred,y_test)
```

Out[9]:
```
0.84
```

naive bayes gives accuracy of 84%

In [ ]:
```

```