In [1]:
```
import pandas as pd
import numpy as np
from sklearn.model_selection import train_test_split
from sklearn.naive_bayes import GaussianNB
from sklearn.metrics import accuracy_score, confusion_matrix
```

In [2]:
```
df = pd.read_csv("25_Nguyen Van Linh_Ch3_NBC.csv")
df
```

Out[2]:

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe	Type
0	1.52101	13.64	4.49	1.10	71.78	0.06	8.75	0.00	0.0	1
1	1.51761	13.89	3.60	1.36	72.73	0.48	7.83	0.00	0.0	1
2	1.51618	13.53	3.55	1.54	72.99	0.39	7.78	0.00	0.0	1
3	1.51766	13.21	3.69	1.29	72.61	0.57	8.22	0.00	0.0	1
4	1.51742	13.27	3.62	1.24	73.08	0.55	8.07	0.00	0.0	1
...	...	...	...	...	...	...	...	...	...	...
209	1.51623	14.14	0.00	2.88	72.61	0.08	9.18	1.06	0.0	7
210	1.51685	14.92	0.00	1.99	73.06	0.00	8.40	1.59	0.0	7
211	1.52065	14.36	0.00	2.02	73.42	0.00	8.44	1.64	0.0	7
212	1.51651	14.38	0.00	1.94	73.61	0.00	8.48	1.57	0.0	7
213	1.51711	14.23	0.00	2.08	73.36	0.00	8.62	1.67	0.0	7

214 rows × 10 columns

In [3]:
```
X = df.iloc[:,:-1]
y = df.iloc[:,9]
```

In [4]:
```
X
```

Out[4]:

	RI	Na	Mg	Al	Si	K	Ca	Ba	Fe
0	1.52101	13.64	4.49	1.10	71.78	0.06	8.75	0.00	0.0
1	1.51761	13.89	3.60	1.36	72.73	0.48	7.83	0.00	0.0
2	1.51618	13.53	3.55	1.54	72.99	0.39	7.78	0.00	0.0
3	1.51766	13.21	3.69	1.29	72.61	0.57	8.22	0.00	0.0
4	1.51742	13.27	3.62	1.24	73.08	0.55	8.07	0.00	0.0
...	...	...	...	...	...	...	...	...	...
209	1.51623	14.14	0.00	2.88	72.61	0.08	9.18	1.06	0.0
210	1.51685	14.92	0.00	1.99	73.06	0.00	8.40	1.59	0.0
211	1.52065	14.36	0.00	2.02	73.42	0.00	8.44	1.64	0.0
212	1.51651	14.38	0.00	1.94	73.61	0.00	8.48	1.57	0.0
213	1.51711	14.23	0.00	2.08	73.36	0.00	8.62	1.67	0.0

214 rows × 9 columns

In [5]:
```
y
```

Out[5]:
```
0      1
1      1
2      1
3      1
4      1
      ..
209    7
210    7
211    7
212    7
213    7
Name: Type, Length: 214, dtype: int64
```

In [6]:
```
X_train, X_test, y_train, y_test = train_test_split(X,y,test_size=0.3,random_state=0)
```

In [7]:
```
NBModel = GaussianNB()
```

In [ ]:
```
NBModel.fit(X_train,y_train)
```

Out[7]:
```
GaussianNB()
```

In [8]:
```
y_predicted=NBModel.predict(X_test)
y_predicted
```

Out[8]:
```
array([7, 1, 1, 6, 2, 1, 1, 1, 1, 1, 2, 1, 1, 2, 1, 7, 1, 1, 1, 1, 2, 3,
       7, 7, 1, 1, 7, 1, 1, 1, 1, 6, 1, 1, 1, 1, 1, 1, 1, 7, 5, 6, 1, 1,
       1, 1, 1, 1, 1, 1, 1, 2, 7, 1, 1, 1, 1, 1, 7, 1, 1, 1, 1, 2, 1],
      dtype=int64)
```

In [9]:
```
accuracy_score(y_test,y_predicted)*100
```

Out[9]:
```
46.15384615384615
```

In [11]:
```
import sklearn.metrics as metrics
print(metrics.classification_report(y_test,y_predicted))
print(metrics.confusion_matrix(y_test,y_predicted))
```

```
              precision    recall  f1-score   support

           1       0.39      0.86      0.54        21
           2       0.50      0.12      0.19        26
           3       0.00      0.00      0.00         7
           5       0.00      0.00      0.00         2
           6       0.67      1.00      0.80         2
           7       0.88      1.00      0.93         7

    accuracy                           0.46        65
   macro avg       0.41      0.50      0.41        65
weighted avg       0.44      0.46      0.37        65

[[18  1  0  0  1  1]
 [21  3  1  1  0  0]
 [ 7  0  0  0  0  0]
 [ 0  2  0  0  0  0]
 [ 0  0  0  0  2  0]
 [ 0  0  0  0  0  7]]
```

In [21]:
```
X_newval=np.array([1.5123,12.61,2.93,1.18,52.25,0,7.62,0.01,0])
```

In [22]:
```
y_pred=NBModel.predict([X_newval])
```

In [23]:
```
y_pred
```

Out[23]:
```
array([5], dtype=int64)
```

Với các tham số nhập vào là: [1.5123, 12.61, 2.93, 1.18, 52.25, 0, 7.62, 0.01, 0] thì thuật toán dự đoán thuộc vào Type [5]

In [ ]:
```

```