In [1]:

```
import pandas as pd
data=pd.read_csv('SMSSpamCollection',sep='\t',names=["label","message"])
import re
import nltk
nltk.download('stopwords')
from nltk.corpus import stopwords
from nltk.stem.porter import PorterStemmer
nltk.download('wordnet')
from nltk.stem import WordNetLemmatizer
cleane_corpous=[]
ps=PorterStemmer()
ls=WordNetLemmatizer()

for i in range(0,len(data)):
     review=re.sub("[^A-Za-z]",' ',data['message'][i])
     review=review.lower()
     review=review.split()
     review=[ls.lemmatize(word)for word in review if word not in stopwords.words('english')
     review=' '.join(review)
     cleane_corpous.append(review)

from sklearn.feature_extraction.text import CountVectorizer
cv=CountVectorizer(max_features=2500)
x=cv.fit_transform(cleane_corpous).toarray()

from sklearn.preprocessing import LabelEncoder
le=LabelEncoder()
data['label']=le.fit_transform(data['label'])
y=data['label']


from sklearn.model_selection import train_test_split
x_train,x_test,y_train,y_test=train_test_split(x,y,test_size=0.20, random_state=12,shuffle=

from sklearn.naive_bayes import MultinomialNB
nbmodel=MultinomialNB()
nbmodel.fit(x_train,y_train)

y_pred=nbmodel.predict(x_test)

from sklearn.metrics import accuracy_score,confusion_matrix
accuracy_score(y_test,y_pred)
confusion_matrix(y_test, y_pred)
```

```
[nltk_data] Downloading package stopwords to
[nltk_data]     C:\Users\hp\AppData\Roaming\nltk_data...
[nltk_data]   Package stopwords is already up-to-date!
[nltk_data] Downloading package wordnet to
[nltk_data]     C:\Users\hp\AppData\Roaming\nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
```

Out[1]:

```
array([[958,   8],
       [  6, 143]], dtype=int64)
```

In [3]:

```
print(accuracy_score(y_test,y_pred))
```

```
0.9874439461883409
```

In [4]:

```
print(confusion_matrix(y_test, y_pred))
```

```
[[958   8]
 [  6 143]]
```