In [3]:
```
import os
import io
import numpy
import pandas as pd
from pandas import DataFrame
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.naive_bayes import MultinomialNB

def readFiles(path):
    for root, dirnames, filenames in os.walk(path):
        for filename in filenames:
            path = os.path.join(root, filename)

            inBody = False
            lines = []
            f = io.open(path, 'r', encoding='latin1')
            for line in f:
                if inBody:
                    lines.append(line)
                elif line == '\n':
                    inBody = True
            f.close()
            message = '\n'.join(lines)
            yield path, message
def dataFrameFromDirectory(path, classification):
    rows = []
    index = []
    for filename, message in readFiles(path):
        rows.append({'message': message, 'class': classification})
        index.append(filename)

    return DataFrame(rows, index=index)

data = DataFrame({'message': [], 'class': []})

data = pd.concat([data, dataFrameFromDirectory(r'C:\Users\SAKTHI\Downloads\milestone\git
data = pd.concat([data, dataFrameFromDirectory(r'C:\Users\SAKTHI\Downloads\milestone\git
```

In [4]:
```
data.head()
```

Out[4]:

C:\Users\SAKTHI\Downloads\milestone\github\spam\00001.7848dde101aa985090474a91ec93fcf0	<!DOCTYPE HTML P
C:\Users\SAKTHI\Downloads\milestone\github\spam\00002.d94f1b97e48ed3b553b3508d116e6a09	1) Fight The
C:\Users\SAKTHI\Downloads\milestone\github\spam\00003.2ee33bc6eacdb11f38d052c44819ba6c	1) Fight The
C:\Users\SAKTHI\Downloads\milestone\github\spam\00004.eac8de8d759b7e74154f142194282724	##################
C:\Users\SAKTHI\Downloads\milestone\github\spam\00005.57696a39d7d84318ce497886896bf90d	I thought yo

In [5]:
```
vectorizer = CountVectorizer()
counts = vectorizer.fit_transform(data['message'].values)

classifier = MultinomialNB()
targets = data['class'].values
classifier.fit(counts, targets)
```

Out[5]:
```
▼ MultinomialNB
MultinomialNB()
```

In [6]:
```
examples = ['Free Viagra now!!!', "Hi Bob, how about a game of golf tomorrow?"]
example_counts = vectorizer.transform(examples)
predictions = classifier.predict(example_counts)
predictions
```

Out[6]:
```
array(['spam', 'ham'], dtype='<U4')
```

In [ ]:
```

```