 # import the necessary packages

```
import nltk
from nltk.corpus import names
```

#looks at final letter of a given name with theory that gender can be identified with last letter.
#returns last letter of name

```
def gender_features(word):
    return{'last_letter': word[-1]}
```

#prepare a list of examples and corresponding class labels

```
labeled_names = ([(name, 'male') for name in names.words('male.txt')] +
[(name, 'female') for name in names.words('female.txt')])
import random
random.shuffle(labeled_names)
```

#process the names data and divide list into a training and test set which is used in naïve Bayes #classifier

```
featuresets = [(gender_features(n), gender) for (n, gender) in labeled_names]
train_set, test_set = featuresets[500:], featuresets[:500]
classifier = nltk.NaiveBayesClassifier.train(train_set)
```

#Test Results of classifer:

```
>>> classifier.classify(gender_features('Dan'))
'male'
>>> classifier.classify(gender_features('Angela'))
'female'
```

#Evaluate classifier on larger amount of data showing almost 75% accuracy

```
>>> print(nltk.classify.accuracy(classifier, test_set))
0.746
```

#Most effective features found in distinguishing names' genders
#shows likelihood ratios (a is almost 37 times more likely to be female than male).

```
>>> classifier.show_most_informative_features(5)
Most Informative Features
             last_letter = 'a'            female : male   =     36.5 : 1.0
             last_letter = 'k'              male : female =     33.0 : 1.0
             last_letter = 'v'              male : female =     18.8 : 1.0
             last_letter = 'f'              male : female =     16.1 : 1.0
             last_letter = 'p'              male : female =     12.0 : 1.0
```