In [1]:
```
# import these modules
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize

ps = PorterStemmer()

# choose some words to be stemmed
words = ["program", "programs", "programer", "programing", "programers"]

for w in words:
    print(w, " : ", ps.stem(w))
```

```
program  :  program
programs  :  program
programer  :  program
programing  :  program
programers  :  program
```

In [2]:
```
import nltk
nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to
[nltk_data]     C:\Users\Admin\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping tokenizers\punkt.zip.
```

Out[2]:
```
True
```

In [3]:
```
# importing modules
from nltk.stem import PorterStemmer
from nltk.tokenize import word_tokenize

ps = PorterStemmer()

sentence = "Programers program with programing languages"
words = word_tokenize(sentence)

for w in words:
    print(w, " : ", ps.stem(w))
```

```
Programers  :  program
program  :  program
with  :  with
programing  :  program
languages  :  languag
```

In [4]:
```
from nltk.tokenize import word_tokenize
text = "God is Great! I won a lottery."
print(word_tokenize(text))
```

```
['God', 'is', 'Great', '!', 'I', 'won', 'a', 'lottery', '.']
```

In [5]:
```
from nltk.tokenize import sent_tokenize, word_tokenize

data = "All work and no play makes jack a dull boy, all work and no play"
print(word_tokenize(data))
```

```
['All', 'work', 'and', 'no', 'play', 'makes', 'jack', 'a', 'dull', 'boy',
',', 'all', 'work', 'and', 'no', 'play']
```

In [6]:
```
import nltk
sentence_data = "All work and no play makes jack a dull boy, all work and no p
lay"
nltk_tokens=nltk.sent_tokenize(sentence_data)
print(nltk_tokens)
```

```
['All work and no play makes jack a dull boy, all work and no play']
```

In [7]:
```
from nltk.tokenize import sent_tokenize, word_tokenize

data = "All work and no play makes jack dull boy. All work and no play makes j
ack a dull boy."
print(sent_tokenize(data))
```

```
['All work and no play makes jack dull boy.', 'All work and no play makes jac
k a dull boy.']
```

In [8]:
```
from nltk.tokenize import sent_tokenize, word_tokenize

data = "All work and no play makes jack dull boy. All work and no play makes j
ack a dull boy."

phrases = sent_tokenize(data)
words = word_tokenize(data)

print(phrases)
print(words)
```

```
['All work and no play makes jack dull boy.', 'All work and no play makes jac
k a dull boy.']
['All', 'work', 'and', 'no', 'play', 'makes', 'jack', 'dull', 'boy', '.', 'Al
l', 'work', 'and', 'no', 'play', 'makes', 'jack', 'a', 'dull', 'boy', '.']
```

In [9]:
```
import nltk
sentence_data = "The First sentence is about Python. The Second: about Django.
You can learn Python,Django and Data Ananlysis here. "
nltk_tokens = nltk.sent_tokenize(sentence_data)
print (nltk_tokens)
```

```
['The First sentence is about Python.', 'The Second: about Django.', 'You can
learn Python,Django and Data Ananlysis here.']
```

In [10]:
```
#Non english
import nltk

german_tokenizer = nltk.data.load('tokenizers/punkt/german.pickle')
german_tokens=german_tokenizer.tokenize('Wie geht es Ihnen?  Gut, danke.')
print(german_tokens)
```

```
['Wie geht es Ihnen?', 'Gut, danke.']
```

In [11]:
```
#word_tokenize
import nltk

word_data = "It originated from the idea that there are readers who prefer lea
rning new skills from the comforts of their drawing rooms"
nltk_tokens = nltk.word_tokenize(word_data)
print (nltk_tokens)
```

```
['It', 'originated', 'from', 'the', 'idea', 'that', 'there', 'are', 'reader
s', 'who', 'prefer', 'learning', 'new', 'skills', 'from', 'the', 'comforts',
'of', 'their', 'drawing', 'rooms']
```

In [12]:
```
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords

data = "All work and no play makes jack dull boy. All work and no play makes j
ack a dull boy."
stopWords = set(stopwords.words('english'))
words = word_tokenize(data)
wordsFiltered = []

for w in words:
    if w not in stopWords:
        wordsFiltered.append(w)

print(wordsFiltered)
```

```
['All', 'work', 'play', 'makes', 'jack', 'dull', 'boy', '.', 'All', 'work',
'play', 'makes', 'jack', 'dull', 'boy', '.']
```

In [13]:
```
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize

example_sent = """This is a sample sentence, showing off the stop words filtra
tion."""

stop_words = set(stopwords.words('english'))

word_tokens = word_tokenize(example_sent)

filtered_sentence = [w for w in word_tokens if not w in stop_words]

filtered_sentence = []

for w in word_tokens:
    if w not in stop_words:
        filtered_sentence.append(w)

print(word_tokens)
print(filtered_sentence)
```

```
['This', 'is', 'a', 'sample', 'sentence', ',', 'showing', 'off', 'the', 'sto
p', 'words', 'filtration', '.']
['This', 'sample', 'sentence', ',', 'showing', 'stop', 'words', 'filtration',
'.']
```

In [16]:
```
import nltk
nltk.download('averaged_perceptron_tagger')
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     C:\Users\Admin\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping taggers\averaged_perceptron_tagger.zip.
```

Out[16]:
```
True
```

In [17]:
```
from nltk.corpus import stopwords
from nltk.tokenize import word_tokenize,sent_tokenize

example_sent = """This is a sample sentence, showing off the stop words filtra
tion."""

stop_words = set(stopwords.words('english'))
tokenized = sent_tokenize(example_sent)
for i in tokenized:
    wordlist=nltk.word_tokenize(i)
    wordlist = [w for w in wordlist if not w in stop_words]
    tagged = nltk.pos_tag(wordlist)
    print(tagged)
```

```
[('This', 'DT'), ('sample', 'JJ'), ('sentence', 'NN'), (',', ','), ('showin
g', 'VBG'), ('stop', 'JJ'), ('words', 'NNS'), ('filtration', 'NN'), ('.',
'.')]
```

In [18]:
```
nltk.download('sinica_treebank')
nltk.corpus.sinica_treebank.tagged_words()
```

```
[nltk_data] Downloading package sinica_treebank to
[nltk_data]     C:\Users\Admin\AppData\Roaming\nltk_data...
[nltk_data]   Unzipping corpora\sinica_treebank.zip.
```

Out[18]:
```
[('一', 'Neu'), ('友情', 'Nad'), ('嘉珍', 'Nba'), ...]
```

In [19]:
```
nltk.download('indian')
nltk.corpus.indian.tagged_words()
```

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

Out[19]:
```
[('মহিষের', 'NN'), ('সন্তান', 'NN'), (':', 'SYM'), ...]
```

THANK YOU