In [3]:

```
import nltk
from nltk import word_tokenize
```

In [11]:

```
sentence = "I am living in Coimbatore"
```

```
split into tokens
```

In [12]:

```
tokens = word_tokenize(sentence)
print(tokens)
```

```
['I', 'am', 'living', 'in', 'Coimbatore']
```

```
Stemming
```

In [17]:

```
from nltk.stem import PorterStemmer
from nltk.tokenize import sent_tokenize, word_tokenize
stemmed = []
ps = PorterStemmer()
for w in tokens:
    stemmed.append(ps.stem(w))
print(stemmed)
```

```
['I', 'am', 'live', 'in', 'coimbator']
```

```
stop word removal
```

In [18]:

```
from nltk.corpus import stopwords
stopWords = set(stopwords.words('english'))
cleanTokens = [w for w in stemmed if w not in  stopWords]
print(cleanTokens)
```

```
['I', 'live', 'coimbator']
```

```
Part of speech tagging
```

In [19]:

```
tagged = nltk.pos_tag(cleanTokens)
print(tagged)
```

```
[('I', 'PRP'), ('live', 'VBP'), ('coimbator', 'NN')]
```

```
Named entity recognition
```

In [20]:

```
NE = nltk.ne_chunk(tagged)
print(NE)
```

```
(S I/PRP live/VBP coimbator/NN)
```