	CO2160714.5 Assignment:
14.	Implement textMining

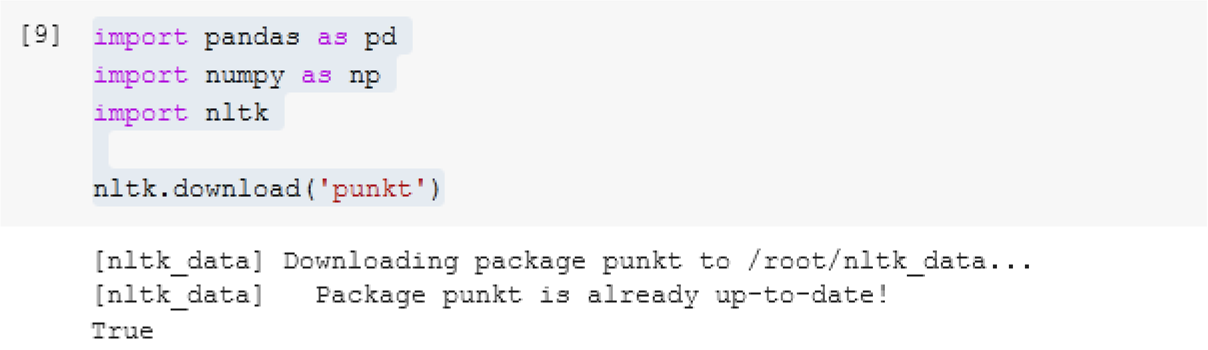
Input:

```
import pandas as pd
import numpy as np
import nltk

nltk.download('punkt')3

text = "In Brazil they drive on right hand side of the road. Brazil has a large co
from nltk.tokenize import word_tokenize
token = word_tokenize(text)
token

from nltk.probability import FreqDist
fdist =  FreqDist(token)
fdist

from nltk.stem import PorterStemmer
pst = PorterStemmer()
pst.stem("Writing")

stm = ['frozen','freezing','freezes']
for word in stm:
  print(word+" : "+pst.stem(word))

from nltk.stem import LancasterStemmer
lst = LancasterStemmer()
stm = ['take','taking','took','taken']
for word in stm:
  print(word," : ",lst.stem(word))

nltk.download('wordnet')
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
print("Rock:",lemmatizer.lemmatize("Rock"))
print("coropa:",lemmatizer.lemmatize("coropa"))

from nltk import word_tokenize
nltk.download('stopwords')
from nltk.corpus import stopwords
a = set(stopwords.words('english'))
text = "Narendra modi was bon in Vadnagar"
text1 = word_tokenize(text.lower())
print(text1)
```

```
stopwords=[x for x in text1 if x not in a]
print(stopwords)

nltk.download('averaged_perceptron_tagger')
text = "Vote to choose a particular man or a group to represent them in parliament
tex = word_tokenize(text)
for token in tex:
  print(nltk.pos_tag([token]))

text = "We saw the yellow dog"
token = word_tokenize(text)
tags = nltk.pos_tag(token)
reg = 'NP:{<DT>?<JJ>*<NN>}'
a = nltk.RegexpParser(reg)
result = a.parse(tags)
print(result)
```

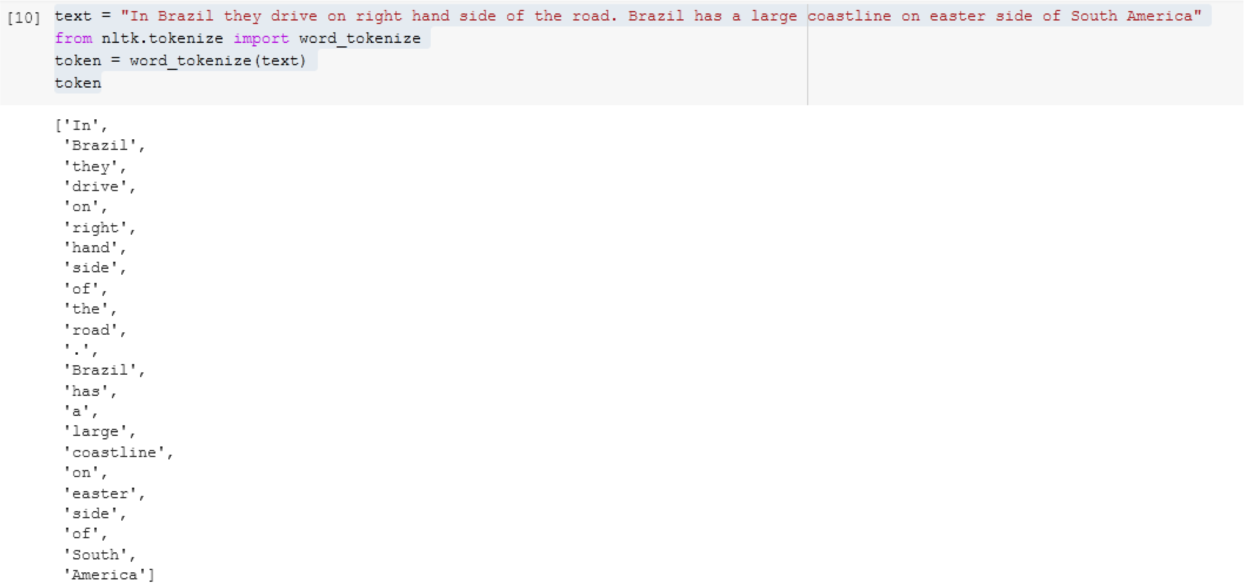
Output:

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
True

['In',
 'Brazil',
 'they',
 'drive',
 'on',
 'right',
 'hand',
 'side',
 'of',
 'the',
 'road',
 '.',
 'Brazil',
 'has',
 'a',
 'large',
 'coastline',
 'on',
 'easter',
 'side',
 'of',
 'South',
 'America']

FreqDist({'.': 1,
          'America': 1,
          'Brazil': 2,
          'In': 1,
          'South': 1,
          'a': 1,
          'coastline': 1,
          'drive': 1,
          'easter': 1,
          'hand': 1,
          'has': 1,
          'large': 1,
          'of': 2,
          'on': 2,
          'right': 1,
          'road': 1,
          'side': 2,
          'the': 1,
          'they': 1})
‘write’

frozen : frozen
freezing : freez
freezes : freez

take  :  tak
taking  :  tak
took  :  took
taken  :  tak

[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Rock: Rock
coropa: coropa

[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
['narendra', 'modi', 'was', 'bon', 'in', 'vadnagar']
['narendra', 'modi', 'bon', 'vadnagar']

[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
[('Vote', 'NN')]
[('to', 'TO')]
[('choose', 'NN')]
[('a', 'DT')]
[('particular', 'JJ')]
[('man', 'NN')]
[('or', 'CC')]
[('a', 'DT')]
[('group', 'NN')]
[('to', 'TO')]
[('represent', 'NN')]
[('them', 'PRP')]
[('in', 'IN')]
[('parliament', 'NN')]

(S We/PRP saw/VBD (NP the/DT yellow/JJ dog/NN))
```



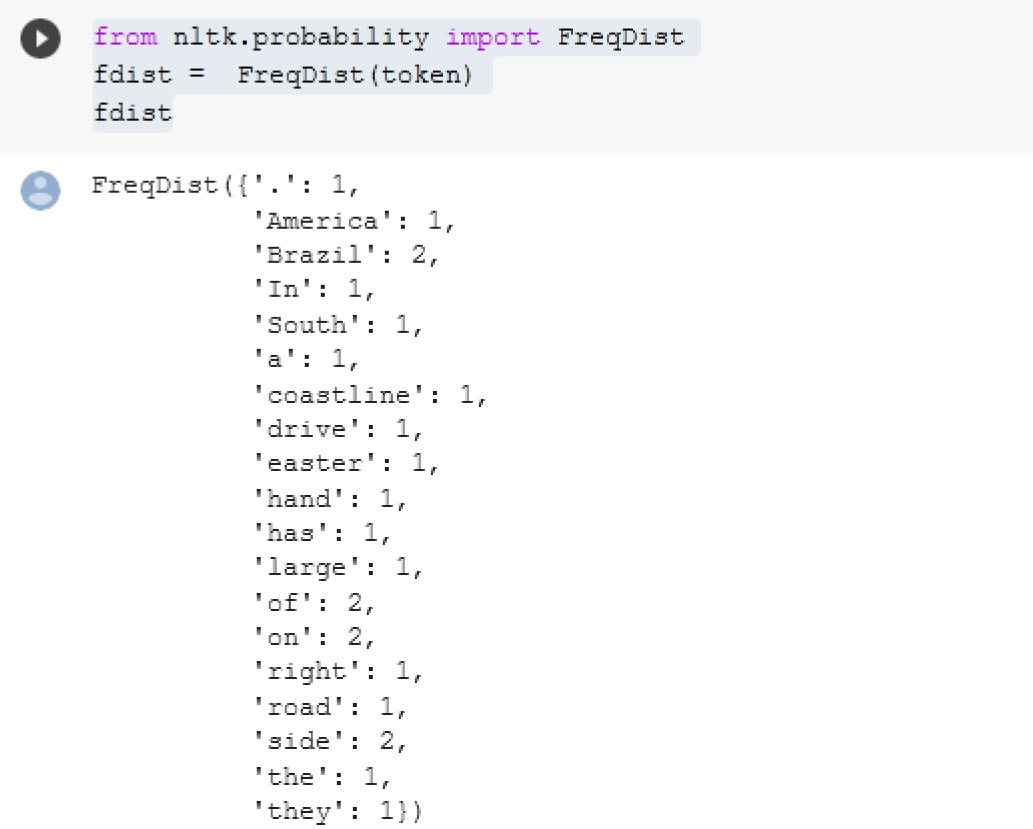
```
[9]  import pandas as pd
     import numpy as np
     import nltk

     nltk.download('punkt')
```

```
[nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data]   Package punkt is already up-to-date!
True
```



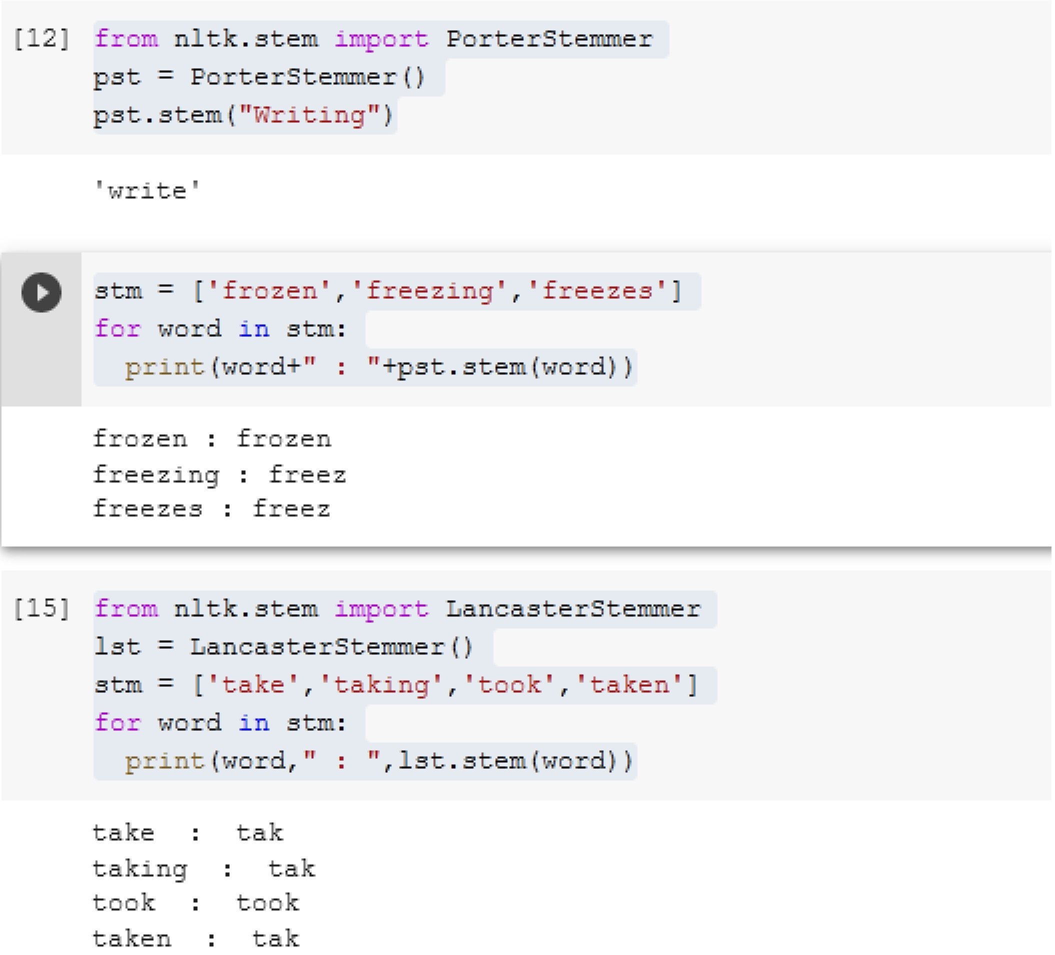
```
[10] text = "In Brazil they drive on right hand side of the road. Brazil has a large coastline on easter side of South America"
     from nltk.tokenize import word_tokenize
     token = word_tokenize(text)
     token
```

```
['In',
 'Brazil',
 'they',
 'drive',
 'on',
 'right',
 'hand',
 'side',
 'of',
 'the',
 'road',
 '.',
 'Brazil',
 'has',
 'a',
 'large',
 'coastline',
 'on',
 'easter',
 'side',
 'of',
 'South',
 'America']
```



```
from nltk.probability import FreqDist
fdist =  FreqDist(token)
fdist
```

```
FreqDist({'.': 1,
          'America': 1,
          'Brazil': 2,
          'In': 1,
          'South': 1,
          'a': 1,
          'coastline': 1,
          'drive': 1,
          'easter': 1,
          'hand': 1,
          'has': 1,
          'large': 1,
          'of': 2,
          'on': 2,
          'right': 1,
          'road': 1,
          'side': 2,
          'the': 1,
          'they': 1})
```



```
[12] from nltk.stem import PorterStemmer
     pst = PorterStemmer()
     pst.stem("Writing")
```

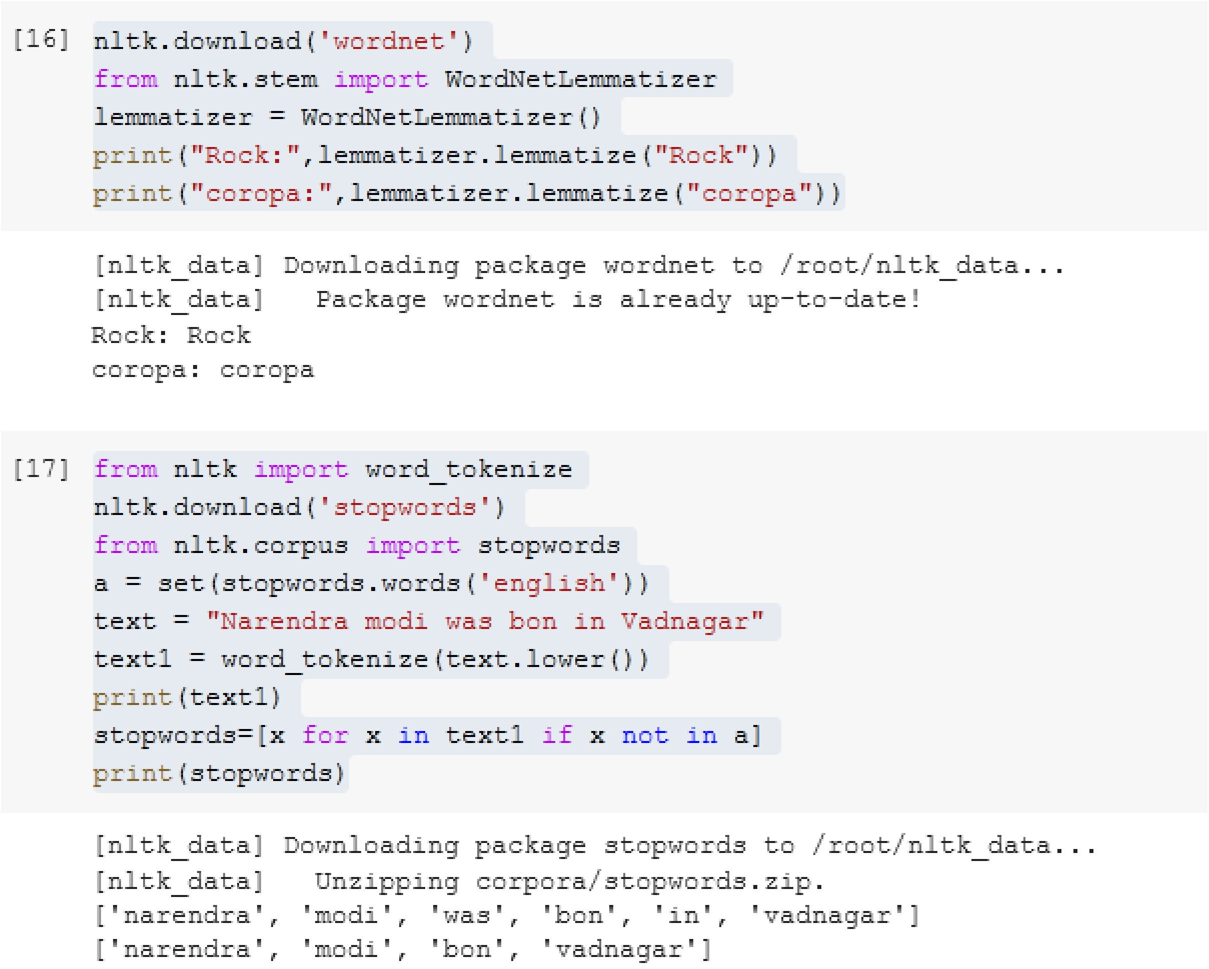
```
'write'
```

```
stm = ['frozen','freezing','freezes']
for word in stm:
  print(word+" : "+pst.stem(word))
```

```
frozen : frozen
freezing : freez
freezes : freez
```

```
[15] from nltk.stem import LancasterStemmer
     lst = LancasterStemmer()
     stm = ['take','taking','took','taken']
     for word in stm:
       print(word," : ",lst.stem(word))
```

```
take  :  tak
taking  :  tak
took  :  took
taken  :  tak
```

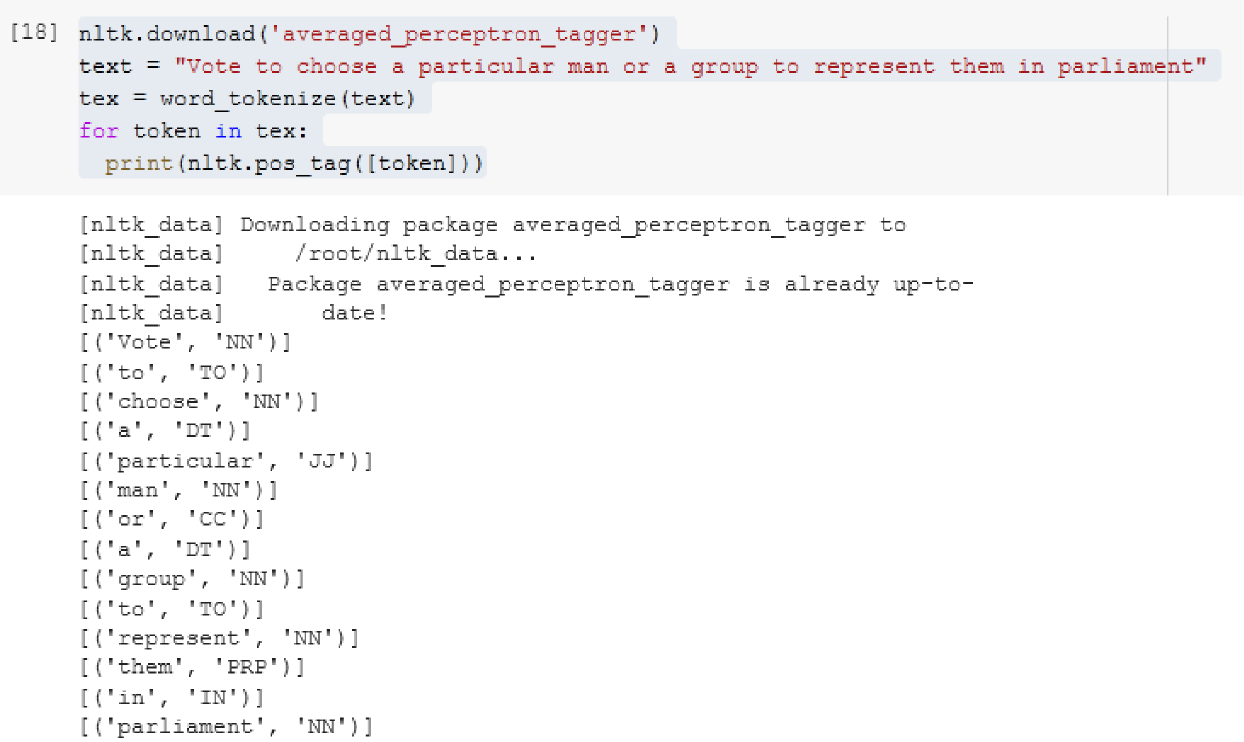


```
[16] nltk.download('wordnet')
     from nltk.stem import WordNetLemmatizer
     lemmatizer = WordNetLemmatizer()
     print("Rock:",lemmatizer.lemmatize("Rock"))
     print("coropa:",lemmatizer.lemmatize("coropa"))
```

```
[nltk_data] Downloading package wordnet to /root/nltk_data...
[nltk_data]   Package wordnet is already up-to-date!
Rock: Rock
coropa: coropa
```

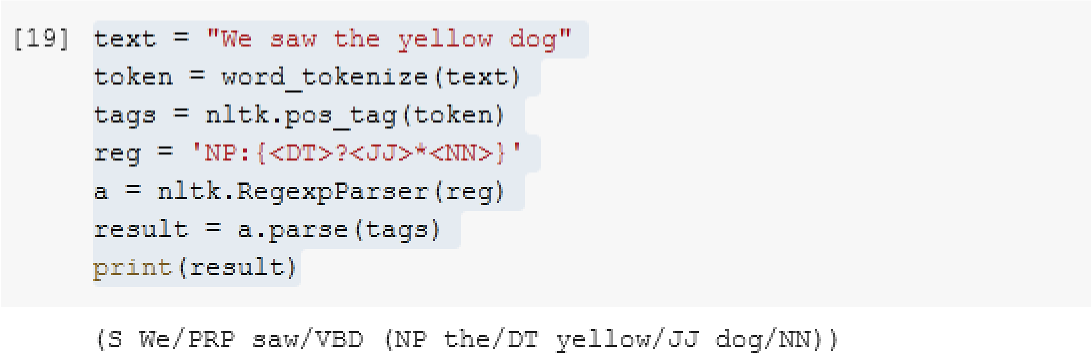
```
[17] from nltk import word_tokenize
     nltk.download('stopwords')
     from nltk.corpus import stopwords
     a = set(stopwords.words('english'))
     text = "Narendra modi was bon in Vadnagar"
     text1 = word_tokenize(text.lower())
     print(text1)
     stopwords=[x for x in text1 if x not in a]
     print(stopwords)
```

```
[nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
['narendra', 'modi', 'was', 'bon', 'in', 'vadnagar']
['narendra', 'modi', 'bon', 'vadnagar']
```



```
[18] nltk.download('averaged_perceptron_tagger')
     text = "Vote to choose a particular man or a group to represent them in parliament"
     tex = word_tokenize(text)
     for token in tex:
       print(nltk.pos_tag([token]))
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Package averaged_perceptron_tagger is already up-to-
[nltk_data]       date!
[('Vote', 'NN')]
[('to', 'TO')]
[('choose', 'NN')]
[('a', 'DT')]
[('particular', 'JJ')]
[('man', 'NN')]
[('or', 'CC')]
[('a', 'DT')]
[('group', 'NN')]
[('to', 'TO')]
[('represent', 'NN')]
[('them', 'PRP')]
[('in', 'IN')]
[('parliament', 'NN')]
```



```
[19] text = "We saw the yellow dog"
     token = word_tokenize(text)
     tags = nltk.pos_tag(token)
     reg = 'NP:{<DT>?<JJ>*<NN>}'
     a = nltk.RegexpParser(reg)
     result = a.parse(tags)
     print(result)
```

```
(S We/PRP saw/VBD (NP the/DT yellow/JJ dog/NN))
```