CSE6060

Statistical Natural Language Processing

Activity 1

Name : Kavianand G

Reg. No. : 19MAI0050

Date : 13 – June – 2020

Explore - NLTK and Corpus

In [1]:

```
#Importing necessasry packages
import nltk
from nltk.stem import PorterStemmer
from nltk.stem import LancasterStemmer
from nltk.stem import RegexpStemmer
from nltk.stem import SnowballStemmer
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk import pos_tag
```

Brown Corpus

The Brown Corpus was the first million-word electronic corpus of English, created in 1961 at Brown University. This corpus contains text from 500 sources, and the sources have been categorized by genre, such as news, editorial, and so on.

In [2]:

```
# here I (Kavianand) used brown corpus
from nltk.corpus import brown
```

In [3]:

```
#viewing raw data from brown corpus
print(brown.raw()[:10])
print("-" *100)
print(brown.raw()[:10000])
```

```
        The/at
----------------------------------------------------------------------------
--------------------------


        The/at Fulton/np-tl County/nn-tl Grand/jj-tl Jury/nn-tl said/vbd F
riday/nr an/at investigation/nn of/in Atlanta's/np$ recent/jj primary/nn e
lection/nn produced/vbd ``/`` no/at evidence/nn ''/'' that/cs any/dti irre
gularities/nns took/vbd place/nn ./.


        The/at jury/nn further/rbr said/vbd in/in term-end/nn presentment
s/nns that/cs the/at City/nn-tl Executive/jj-tl Committee/nn-tl ,/, which/
wdt had/hvd over-all/jj charge/nn of/in the/at election/nn ,/, ``/`` deser
ves/vbz the/at praise/nn and/cc thanks/nns of/in the/at City/nn-tl of/in-t
l Atlanta/np-tl ''/'' for/in the/at manner/nn in/in which/wdt the/at elect
ion/nn was/bedz conducted/vbn ./.
```

In [4]:

```
#print number of characters in Brown Corpus
print("Characters  : ",len(brown.raw()))
#print number of words in Brown Corpus
print("Words       : ",len(brown.words()))
#print the number of sentences in brown corpus
print("Sentences   : ",len(brown.sents()))
```

```
Characters  :  9964284
Words       :  1161192
Sentences   :  57340
```

In [5]:

```
print("No. of Categories : ",len(brown.categories()))
#List the categories in brown corpus
print(brown.categories())
```

```
No. of Categories :  15
['adventure', 'belles_lettres', 'editorial', 'fiction', 'government', 'hobbi
es', 'humor', 'learned', 'lore', 'mystery', 'news', 'religion', 'reviews',
'romance', 'science_fiction']
```

In [6]:

```
#print first 50 words from brown corpus
print(brown.words()[:50])
```

```
['The', 'Fulton', 'County', 'Grand', 'Jury', 'said', 'Friday', 'an', 'invest
igation', 'of', "Atlanta's", 'recent', 'primary', 'election', 'produced', '`
`', 'no', 'evidence', "''", 'that', 'any', 'irregularities', 'took', 'plac
e', '.', 'The', 'jury', 'further', 'said', 'in', 'term-end', 'presentments',
'that', 'the', 'City', 'Executive', 'Committee', ',', 'which', 'had', 'over-
all', 'charge', 'of', 'the', 'election', ',', '``', 'deserves', 'the', 'prai
se']
```

In [7]:

```
#print first 5 sentences from brown corpus
# the sentences are split into words
print(brown.sents()[:5])
```

```
[['The', 'Fulton', 'County', 'Grand', 'Jury', 'said', 'Friday', 'an', 'inves
tigation', 'of', "Atlanta's", 'recent', 'primary', 'election', 'produced',
'``', 'no', 'evidence', "''", 'that', 'any', 'irregularities', 'took', 'plac
e', '.'], ['The', 'jury', 'further', 'said', 'in', 'term-end', 'presentment
s', 'that', 'the', 'City', 'Executive', 'Committee', ',', 'which', 'had', 'o
ver-all', 'charge', 'of', 'the', 'election', ',', '``', 'deserves', 'the',
'praise', 'and', 'thanks', 'of', 'the', 'City', 'of', 'Atlanta', "''", 'fo
r', 'the', 'manner', 'in', 'which', 'the', 'election', 'was', 'conducted',
'.'], ['The', 'September-October', 'term', 'jury', 'had', 'been', 'charged',
'by', 'Fulton', 'Superior', 'Court', 'Judge', 'Durwood', 'Pye', 'to', 'inves
tigate', 'reports', 'of', 'possible', '``', 'irregularities', "''", 'in', 't
he', 'hard-fought', 'primary', 'which', 'was', 'won', 'by', 'Mayor-nominat
e', 'Ivan', 'Allen', 'Jr.', '.'], ['``', 'Only', 'a', 'relative', 'handful',
'of', 'such', 'reports', 'was', 'received', "''", ',', 'the', 'jury', 'sai
d', ',', '``', 'considering', 'the', 'widespread', 'interest', 'in', 'the',
'election', ',', 'the', 'number', 'of', 'voters', 'and', 'the', 'size', 'o
f', 'this', 'city', "''", '.'], ['The', 'jury', 'said', 'it', 'did', 'find',
'that', 'many', 'of', "Georgia's", 'registration', 'and', 'election', 'law
s', '``', 'are', 'outmoded', 'or', 'inadequate', 'and', 'often', 'ambiguou
s', "''", '.']]
```

In [8]:

```
#print 2 paragraphs from brown corpus
print(brown.paras()[:2])
```

```
[[['The', 'Fulton', 'County', 'Grand', 'Jury', 'said', 'Friday', 'an', 'inve
stigation', 'of', "Atlanta's", 'recent', 'primary', 'election', 'produced',
'``', 'no', 'evidence', "''", 'that', 'any', 'irregularities', 'took', 'plac
e', '.']], [['The', 'jury', 'further', 'said', 'in', 'term-end', 'presentmen
ts', 'that', 'the', 'City', 'Executive', 'Committee', ',', 'which', 'had',
'over-all', 'charge', 'of', 'the', 'election', ',', '``', 'deserves', 'the',
'praise', 'and', 'thanks', 'of', 'the', 'City', 'of', 'Atlanta', "''", 'fo
r', 'the', 'manner', 'in', 'which', 'the', 'election', 'was', 'conducted',
'.']]]
```

In [9]:

```
for sent in brown.sents()[:3]: # First 3 sentences.
    text = (' '.join(sent))
    print(text)
```

```
The Fulton County Grand Jury said Friday an investigation of Atlanta's recen
t primary election produced `` no evidence '' that any irregularities took p
lace .
The jury further said in term-end presentments that the City Executive Commi
ttee , which had over-all charge of the election , `` deserves the praise an
d thanks of the City of Atlanta '' for the manner in which the election was
conducted .
The September-October term jury had been charged by Fulton Superior Court Ju
dge Durwood Pye to investigate reports of possible `` irregularities '' in t
he hard-fought primary which was won by Mayor-nominate Ivan Allen Jr. .
```

In [10]:

```
#print tagged words from brown corpus
print(brown.tagged_words()[:50])
```

```
[('The', 'AT'), ('Fulton', 'NP-TL'), ('County', 'NN-TL'), ('Grand', 'JJ-T
L'), ('Jury', 'NN-TL'), ('said', 'VBD'), ('Friday', 'NR'), ('an', 'AT'), ('i
nvestigation', 'NN'), ('of', 'IN'), ("Atlanta's", 'NP$'), ('recent', 'JJ'),
('primary', 'NN'), ('election', 'NN'), ('produced', 'VBD'), ('``', '``'),
('no', 'AT'), ('evidence', 'NN'), ("''", "''"), ('that', 'CS'), ('any', 'DT
I'), ('irregularities', 'NNS'), ('took', 'VBD'), ('place', 'NN'), ('.',
'.'), ('The', 'AT'), ('jury', 'NN'), ('further', 'RBR'), ('said', 'VBD'),
('in', 'IN'), ('term-end', 'NN'), ('presentments', 'NNS'), ('that', 'CS'),
('the', 'AT'), ('City', 'NN-TL'), ('Executive', 'JJ-TL'), ('Committee', 'NN-
TL'), (',', ','), ('which', 'WDT'), ('had', 'HVD'), ('over-all', 'JJ'), ('ch
arge', 'NN'), ('of', 'IN'), ('the', 'AT'), ('election', 'NN'), (',', ','),
('``', '``'), ('deserves', 'VBZ'), ('the', 'AT'), ('praise', 'NN')]
```

In [11]:

```
#print tagged sentences from brown corpus
print(brown.tagged_sents()[:50])

```

```
[[('The', 'AT'), ('Fulton', 'NP-TL'), ('County', 'NN-TL'), ('Grand', 'JJ-T
L'), ('Jury', 'NN-TL'), ('said', 'VBD'), ('Friday', 'NR'), ('an', 'AT'),
('investigation', 'NN'), ('of', 'IN'), ("Atlanta's", 'NP$'), ('recent', 'J
J'), ('primary', 'NN'), ('election', 'NN'), ('produced', 'VBD'), ('``', '`
`'), ('no', 'AT'), ('evidence', 'NN'), ("''", "''"), ('that', 'CS'), ('an
y', 'DTI'), ('irregularities', 'NNS'), ('took', 'VBD'), ('place', 'NN'),
('.', '.')], [('The', 'AT'), ('jury', 'NN'), ('further', 'RBR'), ('said',
'VBD'), ('in', 'IN'), ('term-end', 'NN'), ('presentments', 'NNS'), ('tha
t', 'CS'), ('the', 'AT'), ('City', 'NN-TL'), ('Executive', 'JJ-TL'), ('Com
mittee', 'NN-TL'), (',', ','), ('which', 'WDT'), ('had', 'HVD'), ('over-al
l', 'JJ'), ('charge', 'NN'), ('of', 'IN'), ('the', 'AT'), ('election', 'N
N'), (',', ','), ('``', '``'), ('deserves', 'VBZ'), ('the', 'AT'), ('prais
e', 'NN'), ('and', 'CC'), ('thanks', 'NNS'), ('of', 'IN'), ('the', 'AT'),
('City', 'NN-TL'), ('of', 'IN-TL'), ('Atlanta', 'NP-TL'), ("''", "''"),
('for', 'IN'), ('the', 'AT'), ('manner', 'NN'), ('in', 'IN'), ('which', 'W
DT'), ('the', 'AT'), ('election', 'NN'), ('was', 'BEDZ'), ('conducted', 'V
BN'), ('.', '.')], [('The', 'AT'), ('September-October', 'NP'), ('term',
'NN'), ('jury', 'NN'), ('had', 'HVD'), ('been', 'BEN'), ('charged', 'VB
N'), ('by', 'IN'), ('Fulton', 'NP-TL'), ('Superior', 'JJ-TL'), ('Court',
```

Frequency Distribution

In [12]:

```
text = brown.words(categories='reviews')
fdist = nltk.FreqDist(w.lower() for w in text)
modals = [ 'good', 'bad', 'average',
          'can', 'could', 'may', 'might', 'must', 'will']

for m in modals:
    print(m + ':', fdist[m], end=' ')
    print("\n")
```

```
good: 44

bad: 5

average: 1

can: 45

could: 40

may: 47

might: 26

must: 19

will: 61
```

Conditional Frequency Distribution

In [13]:

```
cfd = nltk.ConditionalFreqDist((genre, word)
           for genre in brown.categories()
           for word in brown.words(categories=genre))
genres = ['news','reviews', 'religion', 'hobbies', 'science_fiction', 'romance', 'humor
modals = ['Good','can', 'could', 'may', 'might', 'must', 'will']
cfd.tabulate(conditions=genres, samples=modals)
```

	Good	can	could	may	might	must	will
news	1	93	86	66	38	50	389
reviews	2	45	40	45	26	19	58
religion	1	82	59	78	12	54	71
hobbies	7	268	58	131	22	83	264
science_fiction	1	16	49	4	12	8	16
romance	4	74	193	11	51	45	43
humor	1	16	30	8	8	9	13

In [ ]:

```

```

Gutenberg Corpus

In [14]:

```
from nltk.corpus import gutenberg
```

In [15]:

```
#List of files in Gutenberg corpus
gutenberg.fileids()
```

Out[15]:

```
['austen-emma.txt',
 'austen-persuasion.txt',
 'austen-sense.txt',
 'bible-kjv.txt',
 'blake-poems.txt',
 'bryant-stories.txt',
 'burgess-busterbrown.txt',
 'carroll-alice.txt',
 'chesterton-ball.txt',
 'chesterton-brown.txt',
 'chesterton-thursday.txt',
 'edgeworth-parents.txt',
 'melville-moby_dick.txt',
 'milton-paradise.txt',
 'shakespeare-caesar.txt',
 'shakespeare-hamlet.txt',
 'shakespeare-macbeth.txt',
 'whitman-leaves.txt']
```

In [16]:

```
print("No. of Words :" ,len(gutenberg.words('shakespeare-caesar.txt')))
print(gutenberg.words(fileids='shakespeare-caesar.txt')[:100])
```

```
No. of Words : 25833
['[', 'The', 'Tragedie', 'of', 'Julius', 'Caesar', 'by', 'William', 'Shakesp
eare', '1599', ']', 'Actus', 'Primus', '.', 'Scoena', 'Prima', '.', 'Enter',
'Flauius', ',', 'Murellus', ',', 'and', 'certaine', 'Commoners', 'ouer', 'th
e', 'Stage', '.', 'Flauius', '.', 'Hence', ':', 'home', 'you', 'idle', 'Crea
tures', ',', 'get', 'you', 'home', ':', 'Is', 'this', 'a', 'Holiday', '?',
'What', ',', 'know', 'you', 'not', '(', 'Being', 'Mechanicall', ')', 'you',
'ought', 'not', 'walke', 'Vpon', 'a', 'labouring', 'day', ',', 'without', 't
he', 'signe', 'Of', 'your', 'Profession', '?', 'Speake', ',', 'what', 'Trad
e', 'art', 'thou', '?', 'Car', '.', 'Why', 'Sir', ',', 'a', 'Carpenter', 'Mu
r', '.', 'Where', 'is', 'thy', 'Leather', 'Apron', ',', 'and', 'thy', 'Rul
e', '?', 'What', 'dost']
```

In [17]:

```
for fileid in gutenberg.fileids():
    print(gutenberg.raw(fileids='shakespeare-caesar.txt')[:])
```

```
[The Tragedie of Julius Caesar by William Shakespeare 1599]


Actus Primus. Scoena Prima.

Enter Flauius, Murellus, and certaine Commoners ouer the Stage.

  Flauius. Hence: home you idle Creatures, get you home:
Is this a Holiday? What, know you not
(Being Mechanicall) you ought not walke
Vpon a labouring day, without the signe
Of your Profession? Speake, what Trade art thou?
  Car. Why Sir, a Carpenter

   Mur. Where is thy Leather Apron, and thy Rule?
What dost thou with thy best Apparrell on?
You sir, what Trade are you?
  Cobl. Truely Sir, in respect of a fine Workman, I am
but as you would say, a Cobler
```

Frequency Distribution

In [18]:

```
text = gutenberg.words('shakespeare-caesar.txt')
fdist = nltk.FreqDist(w.lower() for w in text)
modals = [ 'caesar', 'julius', 'cassius',
          'what', 'could', 'may', 'might', 'must', 'will']

for m in modals:
    print(m + ':', fdist[m], end=' ')
    print("\n")
```

```
caesar: 190

julius: 1

cassius: 85

what: 129

could: 18

may: 38

might: 13

must: 36

will: 163
```

Lexicons

In [19]:

```
from nltk.corpus import names, stopwords, words
```

In [20]:

```
words.fileids()
```

Out[20]:

```
['en', 'en-basic']
```

In [21]:

```
print("No. of Words :" ,len(words.words('en')))
print(words.words('en')[:100])
```

```
No. of Words : 235886
['A', 'a', 'aa', 'aal', 'aalii', 'aam', 'Aani', 'aardvark', 'aardwolf', 'Aar
on', 'Aaronic', 'Aaronical', 'Aaronite', 'Aaronitic', 'Aaru', 'Ab', 'aba',
'Ababdeh', 'Ababua', 'abac', 'abaca', 'abacate', 'abacay', 'abacinate', 'aba
cination', 'abaciscus', 'abacist', 'aback', 'abactinal', 'abactinally', 'aba
ction', 'abactor', 'abaculus', 'abacus', 'Abadite', 'abaff', 'abaft', 'abais
ance', 'abaiser', 'abaissed', 'abalienate', 'abalienation', 'abalone', 'Abam
a', 'abampere', 'abandon', 'abandonable', 'abandoned', 'abandonedly', 'aband
onee', 'abandoner', 'abandonment', 'Abanic', 'Abantes', 'abaptiston', 'Abara
mbo', 'Abaris', 'abarthrosis', 'abarticular', 'abarticulation', 'abas', 'aba
se', 'abased', 'abasedly', 'abasedness', 'abasement', 'abaser', 'Abasgi', 'a
bash', 'abashed', 'abashedly', 'abashedness', 'abashless', 'abashlessly', 'a
bashment', 'abasia', 'abasic', 'abask', 'Abassin', 'abastardize', 'abatabl
e', 'abate', 'abatement', 'abater', 'abatis', 'abatised', 'abaton', 'abato
r', 'abattoir', 'Abatua', 'abature', 'abave', 'abaxial', 'abaxile', 'abaze',
'abb', 'Abba', 'abbacomes', 'abbacy', 'Abbadide']
```

In [22]:

```
stopwords.fileids()
```

Out[22]:

```
['arabic',
 'azerbaijani',
 'danish',
 'dutch',
 'english',
 'finnish',
 'french',
 'german',
 'greek',
 'hungarian',
 'indonesian',
 'italian',
 'kazakh',
 'nepali',
 'norwegian',
 'portuguese',
 'romanian',
 'russian',
```

In [23]:

```
print("No. of Words :" ,len(stopwords.words('german')))
print(stopwords.words('german'))
```

```
No. of Words : 232
['aber', 'alle', 'allem', 'allen', 'aller', 'alles', 'als', 'also', 'am', 'a
n', 'ander', 'andere', 'anderem', 'anderen', 'anderer', 'anderes', 'anderm',
'andern', 'anderr', 'anders', 'auch', 'auf', 'aus', 'bei', 'bin', 'bis', 'bi
st', 'da', 'damit', 'dann', 'der', 'den', 'des', 'dem', 'die', 'das', 'das
s', 'daß', 'derselbe', 'derselben', 'denselben', 'desselben', 'demselben',
'dieselbe', 'dieselben', 'dasselbe', 'dazu', 'dein', 'deine', 'deinem', 'dei
nen', 'deiner', 'deines', 'denn', 'derer', 'dessen', 'dich', 'dir', 'du', 'd
ies', 'diese', 'diesem', 'diesen', 'dieser', 'dieses', 'doch', 'dort', 'durc
h', 'ein', 'eine', 'einem', 'einen', 'einer', 'eines', 'einig', 'einige', 'e
inigem', 'einigen', 'einiger', 'einiges', 'einmal', 'er', 'ihn', 'ihm', 'e
s', 'etwas', 'euer', 'eure', 'eurem', 'euren', 'eurer', 'eures', 'für', 'geg
en', 'gewesen', 'hab', 'habe', 'haben', 'hat', 'hatte', 'hatten', 'hier', 'h
in', 'hinter', 'ich', 'mich', 'mir', 'ihr', 'ihre', 'ihrem', 'ihren', 'ihre
r', 'ihres', 'euch', 'im', 'in', 'indem', 'ins', 'ist', 'jede', 'jedem', 'je
den', 'jeder', 'jedes', 'jene', 'jenem', 'jenen', 'jener', 'jenes', 'jetzt',
'kann', 'kein', 'keine', 'keinem', 'keinen', 'keiner', 'keines', 'können',
'könnte', 'machen', 'man', 'manche', 'manchem', 'manchen', 'mancher', 'manch
es', 'mein', 'meine', 'meinem', 'meinen', 'meiner', 'meines', 'mit', 'muss',
'musste', 'nach', 'nicht', 'nichts', 'noch', 'nun', 'nur', 'ob', 'oder', 'oh
ne', 'sehr', 'sein', 'seine', 'seinem', 'seinen', 'seiner', 'seines', 'selbs
t', 'sich', 'sie', 'ihnen', 'sind', 'so', 'solche', 'solchem', 'solchen', 's
olcher', 'solches', 'soll', 'sollte', 'sondern', 'sonst', 'über', 'um', 'un
d', 'uns', 'unsere', 'unserem', 'unseren', 'unser', 'unseres', 'unter', 'vie
l', 'vom', 'von', 'vor', 'während', 'war', 'waren', 'warst', 'was', 'weg',
'weil', 'weiter', 'welche', 'welchem', 'welchen', 'welcher', 'welches', 'wen
n', 'werde', 'werden', 'wie', 'wieder', 'will', 'wir', 'wird', 'wirst', 'w
o', 'wollen', 'wollte', 'würde', 'würden', 'zu', 'zum', 'zur', 'zwar', 'zwis
chen']
```

In [24]:

```
print("No. of Words :" ,len(stopwords.words('english')))
print(stopwords.words('english'))
```

```
No. of Words : 179
['i', 'me', 'my', 'myself', 'we', 'our', 'ours', 'ourselves', 'you', "you're", "you've", "you'll", "you'd", 'your', 'yours', 'yourself', 'yourselves', 'he', 'him', 'his', 'himself', 'she', "she's", 'her', 'hers', 'herself', 'it', "it's", 'its', 'itself', 'they', 'them', 'their', 'theirs', 'themselves', 'what', 'which', 'who', 'whom', 'this', 'that', "that'll", 'these', 'those', 'am', 'is', 'are', 'was', 'were', 'be', 'been', 'being', 'have', 'has', 'had', 'having', 'do', 'does', 'did', 'doing', 'a', 'an', 'the', 'and', 'but', 'if', 'or', 'because', 'as', 'until', 'while', 'of', 'at', 'by', 'for', 'with', 'about', 'against', 'between', 'into', 'through', 'during', 'before', 'after', 'above', 'below', 'to', 'from', 'up', 'down', 'in', 'out', 'on', 'off', 'over', 'under', 'again', 'further', 'then', 'once', 'here', 'there', 'when', 'where', 'why', 'how', 'all', 'any', 'both', 'each', 'few', 'more', 'most', 'other', 'some', 'such', 'no', 'nor', 'not', 'only', 'own', 'same', 'so', 'than', 'too', 'very', 's', 't', 'can', 'will', 'just', 'don', "don't", 'should', "should've", 'now', 'd', 'll', 'm', 'o', 're', 've', 'y', 'ain', 'aren', "aren't", 'couldn', "couldn't", 'didn', "didn't", 'doesn', "doesn't", 'hadn', "hadn't", 'hasn', "hasn't", 'haven', "haven't", 'isn', "isn't", 'ma', 'mightn', "mightn't", 'mustn', "mustn't", 'needn', "needn't", 'shan', "shan't", 'shouldn', "shouldn't", 'wasn', "wasn't", 'weren', "weren't", 'won', "won't", 'wouldn', "wouldn't"]
```

In [25]:

```
names.fileids()
```

Out[25]:

```
['female.txt', 'male.txt']
```

In [26]:

```
print("No. of Words :" ,len(names.words('male.txt')))
print(names.words('male.txt')[:100])

```

```
No. of Words : 2943
['Aamir', 'Aaron', 'Abbey', 'Abbie', 'Abbot', 'Abbott', 'Abby', 'Abdel', 'Abdul', 'Abdulkarim', 'Abdullah', 'Abe', 'Abel', 'Abelard', 'Abner', 'Abraham', 'Abram', 'Ace', 'Adair', 'Adam', 'Adams', 'Addie', 'Adger', 'Aditya', 'Adlai', 'Adnan', 'Adolf', 'Adolfo', 'Adolph', 'Adolphe', 'Adolpho', 'Adolphus', 'Adrian', 'Adrick', 'Adrien', 'Agamemnon', 'Aguinaldo', 'Aguste', 'Agustin', 'Aharon', 'Ahmad', 'Ahmed', 'Ahmet', 'Ajai', 'Ajay', 'Al', 'Alaa', 'Alain', 'Alan', 'Alasdair', 'Alastair', 'Albatros', 'Albert', 'Alberto', 'Albrecht', 'Alden', 'Aldis', 'Aldo', 'Aldric', 'Aldrich', 'Aldus', 'Aldwin', 'Alec', 'Aleck', 'Alejandro', 'Aleks', 'Aleksandrs', 'Alessandro', 'Alex', 'Alexander', 'Alexei', 'Alexis', 'Alf', 'Alfie', 'Alfonse', 'Alfonso', 'Alfonzo', 'Alford', 'Alfred', 'Alfredo', 'Algernon', 'Ali', 'Alic', 'Alister', 'Alix', 'Allah', 'Allan', 'Allen', 'Alley', 'Allie', 'Allin', 'Allyn', 'Alonso', 'Alonzo', 'Aloysius', 'Alphonse', 'Alphonso', 'Alston', 'Alton', 'Alvin']
```

In [27]:

```
if "George" in names.words('male.txt'):
    print("True")
else:
    print("False")
```

```
True
```

---End of Documentation---