Experiment 1

NAME	**Shreya Shetty**
UID	**2019140059**
CLASS	**TE IT**
BATCH	**B**
SUBJECT	**NLP Lab**

AIM:

1. Install NLTK and perform basic Corpus analysis using NLTK such as frequency distribution
2. Learn about morphological features of a word by analysing it.

THEORY:

What is NLP?

Natural language processing (NLP) refers to the branch of computer science concerned with giving computers the ability to understand text and spoken words in much the same way human beings can. Together, these technologies enable computers to process human language in the form of text or voice data and to 'understand' its full meaning, complete with the speaker or writer's intent and sentiment. NLP drives computer programs that translate text from one language to another, respond to spoken commands, and summarize large volumes of text rapidly—even in real time. There's a good chance you've interacted with NLP in the form of voice-operated GPS systems, digital assistants, speech-to-text dictation software, customer service chatbots, and other consumer conveniences.

What is NLTK?

NLTK (Natural Language Toolkit) Library is a suite that contains libraries and programs for statistical language processing. It is one of the most powerful NLP libraries, which contains packages to make machines understand human language and reply to it with an appropriate response. Learning Natural Language Toolkit will help you add an extra skill and also enhance your knowledge of NLP. Learning the NLTK library is also beneficial for enhancing careers in AI and Natural Language Processing with Python.

Installation(Windows):

1. Installing Jupyter –

    a. ***pip install jupyterlab***
    b. ***pip install jupyter notebook***

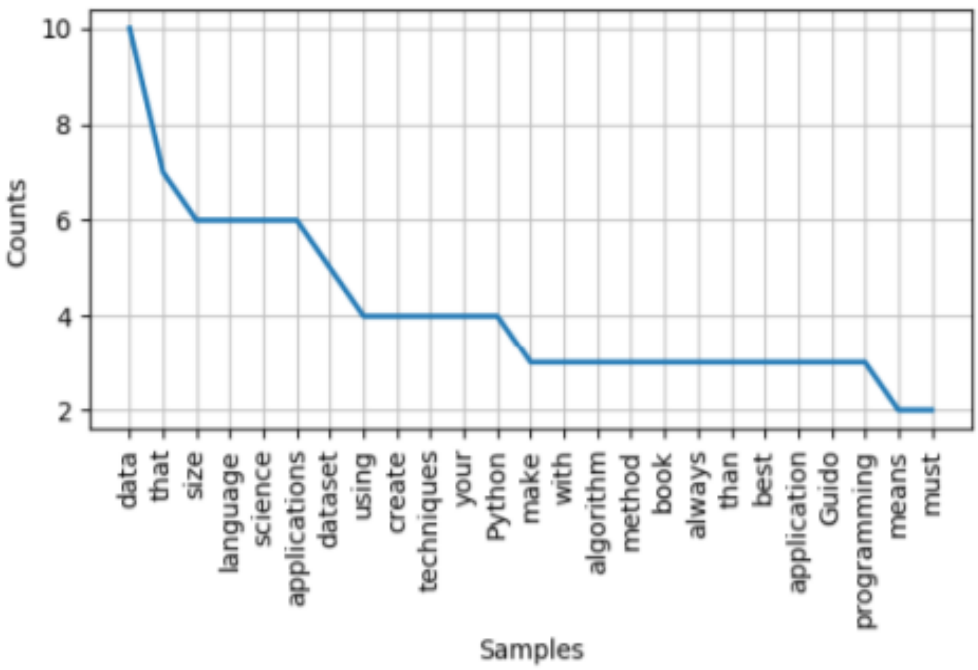
2. Installing NLTK toolkit –

    a. ***pip install nltk***

Frequency distribution:

A frequency distribution records the number of times each outcome of an experiment has occurred. For example, a frequency distribution could be used to record the frequency of each word type in a document. Formally, a frequency distribution can be defined as a function mapping from each sample to the number of times that sample occurred as an outcome.

Frequency distributions are generally constructed by running a number of experiments, and incrementing the count for a sample every time it is an outcome of an experiment.



What is Morphological Analysis of a word?

The morphological level of linguistic processing deals with the study of word structures and word formation, focusing on the analysis of the individual components of words. The most important unit of morphology, defined as having the “minimal unit of meaning”, is referred to as the morpheme.

Take, for example, the word: “unhappiness”. It can be broken down into three morphemes (prefix, stem, and suffix), with each conveying some form of meaning: the prefix un- refers to “not being”, while the suffix -ness refers to “a state of being”. The stem happy is considered as a free morpheme since it is a “word” in its own right.

IDE USED: Jupyter Notebook

LIBRARIES USED:

nltk

PROCEDURE:

Installation :

Installing jupyter lab :

```
C:\Users\AIM>pip install jupyterlab
Collecting jupyterlab
  Downloading jupyterlab-3.2.8-py3-none-any.whl (8.5 MB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 8.5 MB 257 kB/s
Collecting nbclassic~=0.2
  Downloading nbclassic-0.3.5-py3-none-any.whl (25 kB)
Requirement already satisfied: packaging in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyterlab) (21.3)
Requirement already satisfied: jupyter-core in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyterlab) (4.7.1)
Collecting jupyterlab-server~=2.3
  Downloading jupyterlab_server-2.10.3-py3-none-any.whl (61 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 61 kB 298 kB/s
Collecting jupyter-server~=1.4
  Downloading jupyter_server-1.13.4-py3-none-any.whl (395 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 395 kB 344 kB/s
Requirement already satisfied: jinja2>=2.1 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyterlab) (2.11.3)
Requirement already satisfied: ipython in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyterlab) (7.30.1)
Requirement already satisfied: tornado>=6.1.0 in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyterlab) (6.1)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jinja2>=2.1->jupyterlab) (1.1.1)
Requirement already satisfied: pyzmq>=17 in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyter-server~=1.4->jupyterlab) (22.0.3)
Collecting websocket-client
  Downloading websocket_client-1.2.3-py3-none-any.whl (53 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 53 kB 129 kB/s
Collecting argon2-cffi
  Downloading argon2_cffi-21.3.0-py3-none-any.whl (14 kB)
Collecting Send2Trash
  Downloading Send2Trash-1.8.0-py3-none-any.whl (18 kB)
Requirement already satisfied: ipython-genutils in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyter-server~=1.4->jupyterlab) (0.2.0)
Requirement already satisfied: traitlets>=5 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyter-server~=1.4->jupyterlab) (5.1.1)
Collecting nbformat
  Downloading nbformat-5.1.3-py3-none-any.whl (178 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 178 kB 168 kB/s
Collecting terminado>=0.8.3
  Downloading terminado-0.13.1-py3-none-any.whl (14 kB)
Collecting anyio<4,>=3.1.0
  Downloading anyio-3.5.0-py3-none-any.whl (79 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 79 kB 144 kB/s
Requirement already satisfied: jupyter-client>=6.1.1 in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyter-server~=1.4->jupyterlab) (6.1.12)
Collecting nbconvert
  Downloading nbconvert-6.4.1-py3-none-any.whl (557 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 557 kB 344 kB/s
Collecting prometheus-client
  Downloading prometheus_client-0.13.1-py3-none-any.whl (57 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 57 kB 236 kB/s
Requirement already satisfied: pywin32>=1.0 in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyter-core->jupyterlab) (300)
Collecting entrypoints>=0.2.2
  Downloading entrypoints-0.3-py2.py3-none-any.whl (11 kB)
Collecting jsonschema>=3.0.1
  Downloading jsonschema-4.4.0-py3-none-any.whl (72 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 72 kB 46 kB/s
```

Installing jupyter notebook :

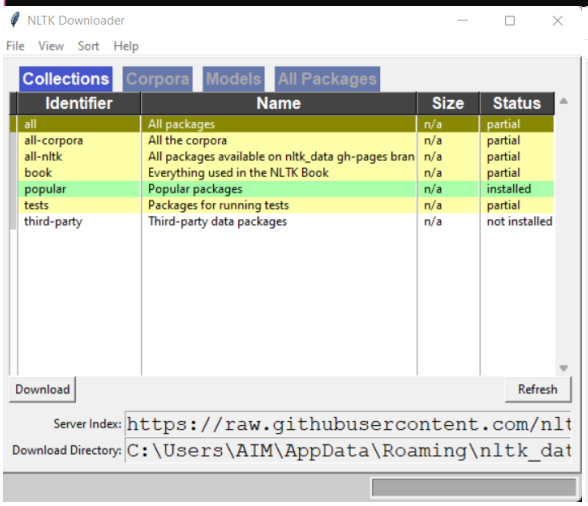
```
C:\Users\AIM>pip install jupyter notebook
Collecting jupyter
  Downloading jupyter-1.0.0-py2.py3-none-any.whl (2.7 kB)
Requirement already satisfied: notebook in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (6.4.8)
Requirement already satisfied: nbconvert in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyter) (6.4.1)
Requirement already satisfied: ipykernel in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyter) (5.5.0)
Collecting jupyter-console
  Downloading jupyter_console-6.4.0-py3-none-any.whl (22 kB)
Collecting qtconsole
  Downloading qtconsole-5.2.2-py3-none-any.whl (120 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 120 kB 364 kB/s
Collecting ipywidgets
  Downloading ipywidgets-7.6.5-py2.py3-none-any.whl (121 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 121 kB 344 kB/s
Requirement already satisfied: jupyter-client>=5.3.4 in c:\users\aim\appdata\roaming\python\python38\site-packages (from notebook) (6.1.12)
Requirement already satisfied: tornado>=6.1 in c:\users\aim\appdata\roaming\python\python38\site-packages (from notebook) (6.1)
Requirement already satisfied: jupyter-core>=4.6.1 in c:\users\aim\appdata\roaming\python\python38\site-packages (from notebook) (4.7.1)
Requirement already satisfied: pyzmq>=17 in c:\users\aim\appdata\roaming\python\python38\site-packages (from notebook) (22.0.3)
Requirement already satisfied: prometheus-client in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (0.13.1)
Requirement already satisfied: nest-asyncio>=1.5 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (1.5.4)
Requirement already satisfied: ipython-genutils in c:\users\aim\appdata\roaming\python\python38\site-packages (from notebook) (0.2.0)
Requirement already satisfied: nbformat in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (5.1.3)
Requirement already satisfied: terminado>=0.8.3 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (0.13.1)
Requirement already satisfied: jinja2 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (2.11.3)
Requirement already satisfied: argon2-cffi in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (21.3.0)
Requirement already satisfied: Send2Trash>=1.8.0 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (1.8.0)
Requirement already satisfied: traitlets>=4.2.1 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from notebook) (5.1.1)
Requirement already satisfied: python-dateutil>=2.1 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyter-client>=5.3.4->notebook) (2.8.1)
Requirement already satisfied: pywin32>=1.0 in c:\users\aim\appdata\roaming\python\python38\site-packages (from jupyter-core>=4.6.1->notebook) (300)
Requirement already satisfied: pywinpty>=1.1.0 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from terminado>=0.8.3->notebook) (2.0.1)
Requirement already satisfied: argon2-cffi-bindings in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from argon2-cffi->notebook) (21.2.0)
Requirement already satisfied: ipython>=5.0.0 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from ipykernel->jupyter) (7.30.1)
Collecting widgetsnbextension~=3.5.0
  Downloading widgetsnbextension-3.5.2-py2.py3-none-any.whl (1.6 MB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 1.6 MB 218 kB/s
Collecting jupyterlab-widgets>=1.0.0
  Downloading jupyterlab_widgets-1.0.2-py3-none-any.whl (243 kB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 243 kB 312 kB/s
Requirement already satisfied: jsonschema!=2.5.0,>=2.4 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbformat->notebook) (4.4.0)
Requirement already satisfied: MarkupSafe>=0.23 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jinja2->notebook) (1.1.1)
Requirement already satisfied: prompt-toolkit!=3.0.0,!=3.0.1,<3.1.0,>=2.0.0 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyter-console->jupyter) (3.0.23)
Requirement already satisfied: pygments in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from jupyter-console->jupyter) (2.10.0)
Requirement already satisfied: entrypoints>=0.2.2 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (0.3)
Requirement already satisfied: defusedxml in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (0.7.1)
Requirement already satisfied: jupyterlab-pygments in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (0.1.2)
Requirement already satisfied: bleach in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (4.1.0)
Requirement already satisfied: nbclient<0.6.0,>=0.5.0 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (0.5.10)
Requirement already satisfied: pandocfilters>=1.4.1 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (1.5.0)
Requirement already satisfied: mistune<2,>=0.8.1 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nbconvert->jupyter) (0.8.4)
```

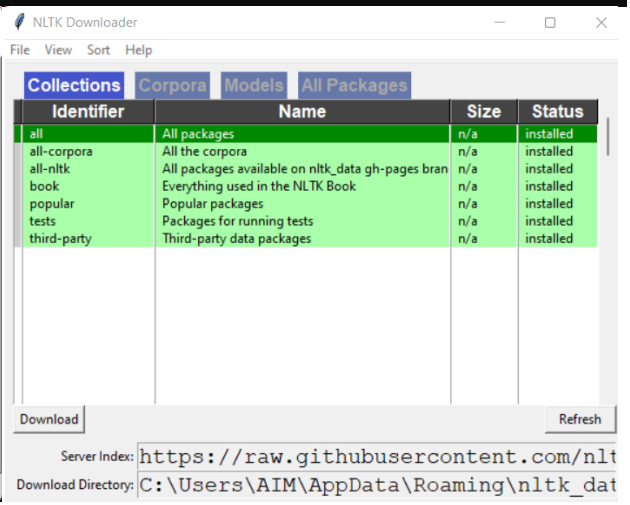
Installing nltk :

```
C:\Users\AIM>pip install nltk
Collecting nltk
  Downloading nltk-3.6.7-py3-none-any.whl (1.5 MB)
     |ƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒƒ| 1.5 MB 64 kB/s
Requirement already satisfied: click in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from n
ltk) (7.1.2)
Requirement already satisfied: joblib in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from
nltk) (1.0.0)
Requirement already satisfied: tqdm in c:\users\aim\appdata\local\programs\python\python38\lib\site-packages (from nl
tk) (4.59.0)
Requirement already satisfied: regex>=2021.8.3 in c:\users\aim\appdata\local\programs\python\python38\lib\site-packag
es (from nltk) (2021.10.8)
Installing collected packages: nltk
Successfully installed nltk-3.6.7
```

Installing nltk packages :

```
D:\PROJECT_AND_CODES\Jupyter>py
Python 3.8.1 (tags/v3.8.1:1b293b6, Dec 18 2019, 23:11:46) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> nltk.download()
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
NameError: name 'nltk' is not defined
>>> import nltk
>>> nltk.download()
showing info https://raw.githubusercontent.com/nltk/nltk_data/gh-pages/index.xml
True
```





Starting Jupyter notebook :

```
D:\PROJECT_AND_CODES\Jupyter>jupyter notebook
[I 2022-01-31 09:31:42.148 LabApp] JupyterLab extension loaded from C:\users\aim\appdata\local\programs\python\python
38\lib\site-packages\jupyterlab
[I 2022-01-31 09:31:42.149 LabApp] JupyterLab application directory is C:\Users\AIM\AppData\Local\Programs\Python\Pyt
hon38\share\jupyter\lab
[I 09:31:42.168 NotebookApp] Serving notebooks from local directory: D:\PROJECT_AND_CODES\Jupyter
[I 09:31:42.168 NotebookApp] Jupyter Notebook 6.4.8 is running at:
[I 09:31:42.169 NotebookApp] http://localhost:8888/?token=d8929f4e0be1c7a37eafcc157e532a372de20b5ea9ae4305
[I 09:31:42.169 NotebookApp]  or http://127.0.0.1:8888/?token=d8929f4e0be1c7a37eafcc157e532a372de20b5ea9ae4305
[I 09:31:42.169 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation)
.
[C 09:31:42.373 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///C:/Users/AIM/AppData/Roaming/jupyter/runtime/nbserver-16092-open.html
    Or copy and paste one of these URLs:
        http://localhost:8888/?token=d8929f4e0be1c7a37eafcc157e532a372de20b5ea9ae4305
     or http://127.0.0.1:8888/?token=d8929f4e0be1c7a37eafcc157e532a372de20b5ea9ae4305
[I 09:32:03.315 NotebookApp] 302 GET /?token=d8929f4e0be1c7a37eafcc157e532a372de20b5ea9ae4305 (127.0.0.1) 1.990000ms
```

CODE:

```
#Access corpus
import nltk
from nltk.corpus import genesis
from nltk.probability import FreqDist
#  Accessing Fileids of corpus genesis
genesis.fileids()
```

```
# Word from fileid english-web.txt
genesis.words('english-web.txt')
g_words = genesis.words('english-web.txt')
# Printing length of genesis words of file english-web.txt
print(len(g_words))
fd = nltk.FreqDist(g_words)
# Taking the specific words only if their frequency is greater than 4.
filter_words = dict([(m, n) for m, n in fd.items() if len(m) > 4])
len(filter_words)
# Printing the sorted filtered words
for key in sorted(filter_words):
    print("%s: %s" % (key, filter_words[key]))
fd = nltk.FreqDist(filter_words)
# Plotting graph for 20 words
fd.plot(20, cumulative=False)
genesis.raw('english-web.txt')[:1000]
# Lemmatization
from nltk.stem import WordNetLemmatizer
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords
from string import punctuation
punctuation=list(punctuation)
stopwords.words('english')
#Stemming with lemmatization to get proper meaning words after stemming
#Create obj of Lemmatizer
lemmmatizer=WordNetLemmatizer()
#Sentence Tokenizer
sentences=sent_tokenize(genesis.raw('english-web.txt')[:1000])
# print(sentences)
for i in range(len(sentences)):
    words=word_tokenize(sentences[i])
    #List comprehension
    words = [lemmmatizer.lemmatize(word.lower()) for word in words if word not in
set(stopwords.words('english')) and word not in punctuation]
    sentences[i]=' '.join(words)
print(sentences)
```

INPUT: Text Corpus of genesis

In [1]:
```
#Access corpus
import nltk
from nltk.corpus import genesis
from nltk.probability import FreqDist
```

In [2]:
```
#  Accessing Fileids of corpus genesis
genesis.fileids()
```

Out[2]:
```
['english-kjv.txt',
 'english-web.txt',
 'finnish.txt',
 'french.txt',
 'german.txt',
 'lolcat.txt',
 'portuguese.txt',
 'swedish.txt']
```

In [3]:
```
# Word from fileid english-web.txt
genesis.words('english-web.txt')
```

Out[3]:
```
['In', 'the', 'beginning', 'God', 'created', 'the', ...]
```

In [4]:
```
g_words = genesis.words('english-web.txt')
```

In [5]:
```
# Printing length of genesis words of file english-web.txt
print(len(g_words))
```

```
44054
```

In [12]:
```
genesis.raw('english-web.txt')[:1000]
```

Out[12]:
```
'In the beginning God created the heavens and the earth.\nNow the earth was formless and empty.  Darkness was on the surface\no
f the deep.  God\'s Spirit was hovering over the surface\nof the waters.\nGod said, "Let there be light," and there was ligh
t.\nGod saw the light, and saw that it was good.  God divided\nthe light from the darkness.\nGod called the light Day, and the
darkness he called Night.\nThere was evening and there was morning, one day.\nGod said, "Let there be an expanse in the middle
of the waters,\nand let it divide the waters from the waters."\nGod made the expanse, and divided the waters which were under\n
the expanse from the waters which were above the expanse;\nand it was so.\nGod called the expanse sky.  There was evening and t
here\nwas morning, a second day.\nGod said, "Let the waters under the sky be gathered together\nto one place, and let the dry l
and appear;" and it was so.\nGod called the dry land Earth, and the gathering together\nof the waters he called Seas.  God saw
that it '
```

OUTPUT:

In [6]:
```
fd = nltk.FreqDist(g_words)
```

In [7]:
```
# Taking the specific words only if their frequency is greater than 4.
filter_words = dict([(m, n) for m, n in fd.items() if len(m) > 4])
```

In [8]:
```
len(filter_words)
```

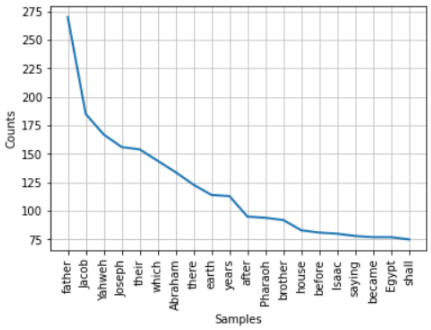
Out[8]:
```
1928
```

In [9]:
```
# Printing the sorted filtered words
for key in sorted(filter_words):
    print("%s: %s" % (key, filter_words[key]))
```

```
Abida: 1
Abimael: 1
Abimelech: 24
About: 1
Abraham: 134
Abram: 61
Accad: 1
According: 1
Achbor: 2
Adbeel: 1
Admah: 3
Adullamite: 3
After: 10
Afterward: 3
Afterwards: 1
Again: 1
Ahuzzath: 1
Allon: 1
Almighty: 6
```

```
In [10]: fd = nltk.FreqDist(filter_words)
```

```
In [11]: # Plotting graph for 20 words
         fd.plot(20, cumulative=False)
```




```
Out[11]: <matplotlib.axes._subplots.AxesSubplot at 0x19242b3f910>
```

```
In [13]: # Lemmatization
         from nltk.stem import WordNetLemmatizer
         from nltk.tokenize import sent_tokenize, word_tokenize
         from nltk.corpus import stopwords
         from string import punctuation
         punctuation=list(punctuation)
         stopwords.words('english')
         #Stemming with lemmatization to get proper meaning words after stemming
         #Create obj of Lemmatizer
         lemmmatizer=WordNetLemmatizer()
         #Sentence Tokenizer
         sentences=sent_tokenize(genesis.raw('english-web.txt')[:1000])
         # print(sentences)

         for i in range(len(sentences)):
             words=word_tokenize(sentences[i])
             #List comprehension
             words = [lemmmatizer.lemmatize(word.lower()) for word in words if word not in set(stopwords.words('english')) and word not i
             sentences[i]=' '.join(words)
         print(sentences)
```

```
['in beginning god created heaven earth', 'now earth formless empty', 'darkness surface deep', "god 's spirit hovering surface
water", "god said `` let light '' light", 'god saw light saw good', 'god divided light darkness', 'god called light day darknes
s called night', 'there evening morning one day', "god said `` let expanse middle water let divide water water '", 'god made e
xpanse divided water expanse water expanse', 'god called expanse sky', 'there evening morning second day', "god said `` let wat
er sky gathered together one place let dry land appear '", 'god called dry land earth gathering together water called sea', 'g
od saw']
```

REFERENCES:

1. https://jupyter.org/install
2. https://www.nltk.org/install.html
3. https://www.youtube.com/watch?v=Qu8pob9RX64
4. https://www.mygreatlearning.com/blog/nltk-tutorial-with-python/
5. https://www.nltk.org/howto/corpus.html
6. https://medium.com/@jeevanchavan143/nlp-tokenization-stemming-lemmatization-bag-of-words-tf-idf-pos-7650f83c60be
7. https://www.pythonprogramming.in/find-frequency-of-each-word-from-a-text-file-using-nltk.html
8. https://medium.com/@CKEspanol/what-are-the-different-levels-of-nlp-how-do-these-integrate-with-information-retrieval-c0de6b9ebf61