

```
import nltk
nltk.download('punkt')
from nltk.tokenize import word_tokenize
```

```
↳ [nltk_data] Downloading package punkt to /root/nltk_data...
[nltk_data] Unzipping tokenizers/punkt.zip.
```

```
text="In summary, even though many rocks may look alike, there are three types of rocks w
```

```
words=word_tokenize(text)
print(words)
```

```
↳ ['In', 'summary', ',', 'even', 'though', 'many', 'rocks', 'may', 'look', 'alike', '']
```

```
from nltk import FreqDist
fd = FreqDist(words)
print(fd)
```

```
↳ <FreqDist with 89 samples and 147 outcomes>
```

```
print(fd.most_common(15))
```

```
↳ [('.', 11), (',', 9), ('of', 6), ('and', 6), ('are', 5), ('rock', 4), ('rocks', 3),
```

```
print(fd.hapaxes())
```

```
↳ ['In', 'summary', 'even', 'though', 'look', 'alike', 'there', 'with', 'different',
```

```
text1="A tree is a wooden stick trying hard to reach the sky. It wants to reach the sun."
```

```
words2=word_tokenize(text1)
print(words2)
```

```
↳ ['A', 'tree', 'is', 'a', 'wooden', 'stick', 'trying', 'hard', 'to', 'reach', 'the',
```

```
from nltk.corpus import stopwords
nltk.download('stopwords')
```

```
↳ [nltk_data] Downloading package stopwords to /root/nltk_data...
[nltk_data]   Unzipping corpora/stopwords.zip.
True
```

```
stop_words=set(stopwords.words("english"))
print(stop_words)
```

```
↳ {'will', 'doing', 'we', 'i', 'and', 'didn', 'down', 'only', 'does', 'won', "you're"
```

```
for w in words2:
    if w not in stop_words:
        print(w)
```

```
↳ A
tree
wooden
stick
trying
hard
reach
sky
.
It
wants
reach
sunlight
needs
life
.
The
stick
called
trunk
.
Raising
tall
also
keeps
leaves
farther
away
insects
animals
.
There
two
```

```

main
types
trees
:
conifers
broad-leaved.Broad-leaved
trees
usually
rounded
.
Conifers
grow
triangular
shape
.
To
called
tree
plant
must
twenty
feet
tall
.
It

```

```
stopwords_in_para=[]
```

```
filtered_words_in_para=[]
```

```

for w in words2:
    if w in stop_words:
        stopwords_in_para.append(w)
print (stopwords_in_para)

```

➞ ['is', 'a', 'to', 'the', 'to', 'the', 'which', 'it', 'for', 'is', 'a', 'itself', 'tl

```

for w in words2:
    if w not in stop_words:
        filtered_words_in_para.append(w)
print (filtered_words_in_para)

```

➞ ['A', 'tree', 'wooden', 'stick', 'trying', 'hard', 'reach', 'sky', '.', 'It', 'want

```

fd0para=FreqDist(filtered_words_in_para)
print("Total tokens=" + str(fd0para.N()))

```

```
print("Total unique tokens=" + str(fd0para.B()))
print("Top 10 tokens")
for token, freq in fd0para.most_common(20):
    print(token + "\t" + str(freq))
```

```
⇒ Total tokens=306
   Total unique tokens=90
   Top 10 tokens
   .          42
   tree       14
   The        12
   water      8
   bark       8
   called     6
   trunk      6
   trees      6
   grow       6
   feet       6
   food       6
   A          4
   stick      4
   reach      4
   It         4
   sunlight           4
   tall       4
   leaves     4
   twenty    4
   make       4
```

Start coding or [generate](#) with AI.

```
import nltk
nltk.download('gutenberg')
```

```
⇒ [nltk_data] Downloading package gutenberg to /root/nltk_data...
   [nltk_data]   Unzipping corpora/gutenberg.zip.
   True
```

```
from nltk.corpus import gutenberg
list_of_words=gutenberg.words('austen-persuasion.txt')
```

```
list_of_words=gutenberg.words('austen-persuasion.txt')
```

```
fd=FreqDist(list_of_words)
print("Total tokens=" + str(fd.N()))
print("Total unique tokens=" + str(fd.B()))
print("Top 10 tokens")
```

```
for token, freq in fd.most_common(10):
    print(token + " " + str(freq))
```

```

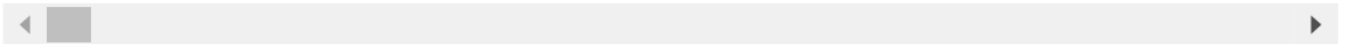
⇒ Total tokens=98171
Total unique tokens=6132
Top 10 tokens
,      6750
the    3120
to     2775
.      2741
and    2739
of     2564
a      1529
in     1346
was    1330
;      1290

```

```
content=file.read("")
print(content)
```

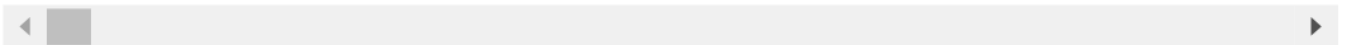
```
file=open("Salonifile1.txt","r")
content=file.read()
print(content)
file.close()
```

```
⇒ The force that makes everything fall to Earth is called gravity. It is a mysterious
```



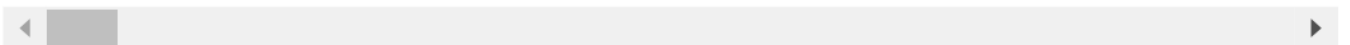
```
wordsNew=word_tokenize(content)
print(wordsNew)
```

```
⇒ ['The', 'force', 'that', 'makes', 'everything', 'fall', 'to', 'Earth', 'is', 'called', 'gravity', 'It', 'is', 'a', 'mysterious', 'force']
```



```
stop_words=set(stopwords.words("english"))
print(stop_words)
```

```
⇒ {'will', 'doing', 'we', 'i', 'and', 'didn', 'down', 'only', 'does', 'won', 'you're'}
```



```
filtered_words=[]
```

```
for w in wordsNew:
    if w not in stop_words:
        filtered_words.append(w)
print(filtered_words)
```

➞ ['The', 'force', 'makes', 'everything', 'fall', 'Earth', 'called', 'gravity', '.',

```
fdnew = FreqDist(wordsNew)
print(fdnew)
```

➞ <FreqDist with 266 samples and 660 outcomes>

```
print(fdnew.most_common(15))
```

➞ [('the', 56), ('of', 29), ('.', 27), ('.', 26), ('gravity', 23), ('is', 19), ('to',

```
print(fdnew.hapaxes())
```

➞ ['makes', 'everything', 'mysterious', 'been', 'studied', 'since', 'first', 'mathema

```
fd0=FreqDist(wordsNew)
print("Total tokens=" + str(fd0.N()))
print("Total unique tokens=" + str(fd0.B()))
print("Top 10 tokens")
for token, freq in fd0.most_common(20):
    print(token + "====>" + str(freq))
```

➞ Total tokens=660
Total unique tokens=266
Top 10 tokens
the====>56
of====>29
,====>27
.====>26
gravity====>23
is====>19
to====>14
on====>13
a====>12
person====>10
Earth====>9
it====>9

```

and==>9
The==>7
's==>6
in==>6
fall==>5
Newton==>5
be==>5
objects==>5

```

```

fdnew2 = FreqDist(filtered_words)
print(fdnew2)

```

```

↵ <FreqDist with 209 samples and 391 outcomes>

```

```

print(fdnew2.most_common(15))

```

```

↵ [(',', 27), ('.', 26), ('gravity', 23), ('person', 10), ('Earth', 9), ('The', 7), (

```

◀ ▶

```

print(fdnew2.hapaxes())

```

```

↵ ['makes', 'everything', 'mysterious', 'studied', 'since', 'first', 'mathematically'

```

◀ ▶

```

fd01=FreqDist(filtered_words)
print("Total tokens=" + str(fd01.N()))
print("Total unique tokens=" + str(fd01.B()))
print("Top 10 tokens")
for token, freq in fd01.most_common(20):
    print(token + "\t" + str(freq))

```

```

↵ Total tokens=391
Total unique tokens=209
Top 10 tokens
,      27
.      26
gravity 23
person 10
Earth   9
The     7
's      6
fall    5
Newton  5
objects 5
much    5
object  5

```

force	4	
theory	4	
important		4
Sun	4	
would	4	
moon	4	
different		4
called	3	