```
import nltk
nltk.download('punkt')
nltk.download('wordnet')
```

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

```
from nltk.tokenize import word_tokenize
tokenized_words = word_tokenize("Educated fool with money on my mind")
print(tokenized_words)
```

```
['Educated', 'fool', 'with', 'money', 'on', 'my', 'mind']
```

```
nltk.word_tokenize("Shadows are our own reflection")
```

```
['Shadows', 'are', 'our', 'own', 'reflection']
```

```
nltk.download('stopwords')
```

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

```
from nltk.corpus import stopwords
stop_words = set(stopwords.words('english'))
print(stop_words)
```

```
{'some', 'themselves', 'and', "needn't", 'here', 'y', 'me', 'isn', 'shouldn', 'any', 'this', "she's", 'than', 'it', 'in', 'but', "s
```

```
print(stop_words)
```

```
{'some', 'themselves', 'and', "needn't", 'here', 'y', 'me', 'isn', 'shouldn', 'any', 'this', "she's", 'than', 'it', 'in', 'but', "s
```

```
filtered_sentence = []
input_text = "I like the weather of this city."
input_text = nltk.word_tokenize(input_text)
for w in input_text:
    if w not in stop_words:
        filtered_sentence.append(w)
print("Filtered Sentence: ")
print(filtered_sentence)
```

```
Filtered Sentence:
['I', 'like', 'weather', 'city', '.']
```

```
from nltk.stem import PorterStemmer
stemmer = PorterStemmer()
input_string = "I am studying without lights."
input_string = nltk.word_tokenize(input_string)
for word in input_string:
  print(stemmer.stem(word))
```

```
i
am
studi
without
light
.
```

```
from nltk.stem import WordNetLemmatizer
lemmatizer = WordNetLemmatizer()
input_string = "Eating grains is a bad habit in cities by mice."
input_string = nltk.word_tokenize(input_string)
for word in input_string:
  print(lemmatizer.lemmatize(word))
```

```
Eating
grain
is
a
bad
habit
```

```
in
city
by
mouse
.
```

```
nltk.download('averaged_perceptron_tagger')
```

```
[nltk_data] Downloading package averaged_perceptron_tagger to
[nltk_data]     /root/nltk_data...
[nltk_data]   Unzipping taggers/averaged_perceptron_tagger.zip.
True
```

```
text = "Astronomy is a humbling and character-building experience."
```

```
result = nltk.pos_tag(word_tokenize(text))
```

```
print(result)
```

```
[('Astronomy', 'NNP'), ('is', 'VBZ'), ('a', 'DT'), ('humbling', 'NN'), ('and', 'CC'), ('character-building', 'JJ'), ('experience',
```

```
nltk.download('tagsets')
```

```
[nltk_data] Downloading package tagsets to /root/nltk_data...
[nltk_data]   Unzipping help/tagsets.zip.
True
```

```
nltk.help.upenn_tagset('NNP')
```

```
NNP: noun, proper, singular
    Motown Venneboerger Czestochwa Ranzer Conchita Trumplane Christos
    Oceanside Escobar Kreisler Sawyer Cougar Yvette Ervin ODI Darryl CTCA
    Shannon A.K.C. Meltex Liverpool ...
```

```
nltk.help.upenn_tagset('VBZ')
```

```
VBZ: verb, present tense, 3rd person singular
    bases reconstructs marks mixes displeases seals carps weaves snatches
    slumps stretches authorizes smolders pictures emerges stockpiles
    seduces fizzes uses bolsters slaps speaks pleads ...
```

```
text_data = ['One day, someone will think about you for the last time in eternity. You will be forgotten by the world and the universe.']
```

```
from sklearn.feature_extraction.text import TfidfVectorizer
```

```
# creating object
tfidf = TfidfVectorizer()

# get tf-df values
result = tfidf.fit_transform(text_data)
```

```
print(result)
```

```
  (0, 15)       0.17407765595569785
  (0, 1)        0.17407765595569785
  (0, 17)       0.17407765595569785
  (0, 3)        0.17407765595569785
  (0, 7)        0.17407765595569785
  (0, 2)        0.17407765595569785
  (0, 5)        0.17407765595569785
  (0, 8)        0.17407765595569785
  (0, 14)       0.17407765595569785
  (0, 9)        0.17407765595569785
  (0, 12)       0.5222329678670935
  (0, 6)        0.17407765595569785
  (0, 18)       0.3481553119113957
  (0, 0)        0.17407765595569785
  (0, 13)       0.17407765595569785
  (0, 16)       0.3481553119113957
  (0, 11)       0.17407765595569785
  (0, 4)        0.17407765595569785
  (0, 10)       0.17407765595569785
```

```
# indexing of terms
tfidf.vocabulary_
```

```
{'one': 10,
 'day': 4,
 'someone': 11,
 'will': 16,
 'think': 13,
 'about': 0,
 'you': 18,
 'for': 6,
 'the': 12,
 'last': 9,
 'time': 14,
 'in': 8,
 'eternity': 5,
 'be': 2,
 'forgotten': 7,
 'by': 3,
 'world': 17,
 'and': 1,
 'universe': 15}
```

```
# tf-idf values in matrix form
print(result.toarray())
```

```
[[0.17407766 0.17407766 0.17407766 0.17407766 0.17407766 0.17407766
  0.17407766 0.17407766 0.17407766 0.17407766 0.17407766 0.17407766
  0.52223297 0.17407766 0.17407766 0.17407766 0.34815531 0.17407766
  0.34815531]]
```