```
pip install nltk
```

```
Looking in indexes: https://pypi.org/simple, https://us-python.pkg.dev/colab-wheels/public/simple/
Requirement already satisfied: nltk in /usr/local/lib/python3.10/dist-packages (3.8.1)
Requirement already satisfied: click in /usr/local/lib/python3.10/dist-packages (from nltk) (8.1.3)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.2.0)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2022.10.31)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.65.0)
```

Downloading NLTK Package

```
nltk.download()
```

```
NLTK Downloader
---------------------------------------------------------------------------
    d) Download   l) List    u) Update   c) Config   h) Help   q) Quit
---------------------------------------------------------------------------
Downloader> d

Download which package (l=list; x=cancel)?
  Identifier> l
Packages:
  [ ] abc..................... Australian Broadcasting Commission 2006
  [ ] alpino.................. Alpino Dutch Treebank
  [ ] averaged_perceptron_tagger Averaged Perceptron Tagger
  [ ] averaged_perceptron_tagger_ru Averaged Perceptron Tagger (Russian)
  [ ] basque_grammars..... Grammars for Basque
  [ ] bcp47................... BCP-47 Language Tags
  [ ] biocreative_ppi..... BioCreAtIvE (Critical Assessment of Information
                           Extraction Systems in Biology)
  [ ] bllip_wsj_no_aux.... BLLIP Parser: WSJ Model
  [ ] book_grammars....... Grammars from NLTK Book
  [ ] brown................... Brown Corpus
  [ ] brown_tei........... Brown Corpus (TEI XML Version)
  [ ] cess_cat............ CESS-CAT Treebank
  [ ] cess_esp............ CESS-ESP Treebank
  [ ] chat80.............. Chat-80 Data Files
  [ ] city_database....... City Database
  [ ] cmudict............. The Carnegie Mellon Pronouncing Dictionary (0.6)
  [ ] comparative_sentences Comparative Sentence Dataset
  [ ] comtrans............ ComTrans Corpus Sample
  [ ] conll2000........... CONLL 2000 Chunking Corpus
Hit Enter to continue:
  [ ] conll2002........... CONLL 2002 Named Entity Recognition Corpus
  [ ] conll2007........... Dependency Treebanks from CoNLL 2007 (Catalan
                           and Basque Subset)
  [ ] crubadan............ Crubadan Corpus
  [ ] dependency_treebank. Dependency Parsed Treebank
  [ ] dolch............... Dolch Word List
  [ ] europarl_raw........ Sample European Parliament Proceedings Parallel
                           Corpus
  [ ] extended_omw........ Extended Open Multilingual WordNet
  [ ] floresta............ Portuguese Treebank
  [ ] framenet_v15........ FrameNet 1.5
  [ ] framenet_v17........ FrameNet 1.7
  [ ] gazetteers.......... Gazeteer Lists
  [ ] genesis............. Genesis Corpus
  [ ] gutenberg........... Project Gutenberg Selections
  [ ] ieer................ NIST IE-ER DATA SAMPLE
  [ ] inaugural........... C-Span Inaugural Address Corpus
  [ ] indian.............. Indian Language POS-Tagged Corpus
  [ ] jeita............... JEITA Public Morphologically Tagged Corpus (in
                           ChaSen format)
  [ ] kimmo............... PC-KIMMO Data Files
Hit Enter to continue:
  [ ] knbc................ KNB Corpus (Annotated blog corpus)
  [ ] large_grammars...... Large context-free and feature-based grammars
                           for parser comparison
  [ ] lin_thesaurus....... Lin's Dependency Thesaurus
  [ ] mac_morpho.......... MAC-MORPHO: Brazilian Portuguese news text with
                           part-of-speech tags
```

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

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

Text Tokenization

```
from nltk.tokenize.punkt import PunktToken
```

```
import nltk
import os
import nltk.corpus
```

```
from nltk import word_tokenize, sent_tokenize
sent = "Twinkle twinkle little star how I wonder what you are"
print(word_tokenize(sent))
print(sent_tokenize(sent))
```

```
['Twinkle', 'twinkle', 'little', 'star', 'how', 'I', 'wonder', 'what', 'you', 'are']
['Twinkle twinkle little star how I wonder what you are']
```

```
from nltk.tokenize import word_tokenize
NLP_tokens = word_tokenize(sent)
NLP_tokens
```

```
['Twinkle',
 'twinkle',
 'little',
 'star',
 'how',
 'I',
 'wonder',
 'what',
 'you',
 'are']
```

Count Word frequency

```
from nltk.probability import FreqDist
fdist = FreqDist()
for words in NLP_tokens:
 fdist[words.lower()]+=1
fdist
```

```
FreqDist({'twinkle': 2, 'little': 1, 'star': 1, 'how': 1, 'i': 1, 'wonder': 1, 'what': 1, 'you': 1, 'are': 1})
```

```
fdist_top5 = fdist.most_common(5)
fdist_top5
```

```
[('twinkle', 2), ('little', 1), ('star', 1), ('how', 1), ('i', 1)]
```

```
fdist.items()
```

```
dict_items([('twinkle', 2), ('little', 1), ('star', 1), ('how', 1), ('i', 1), ('wonder', 1), ('what', 1), ('you', 1), ('are', 1)])
```

Remove stopwords

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

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

```
from nltk.corpus import stopwords
```

```
from nltk.corpus import stopwords
stop_list = stopwords.words('english')

token = word_tokenize(sent)
cleaned_token=[]
for word in token:
  if word not in stop_list:
    cleaned_token.append(word)

print("this is the uncleaned version:" ,token)
print("this is the cleaned version:", cleaned_token)
```

```
this is the uncleaned version: ['Twinkle', 'twinkle', 'little', 'star', 'how', 'I', 'wonder', 'what', 'you', 'are']
this is the cleaned version: ['Twinkle', 'twinkle', 'little', 'star', 'I', 'wonder']
```

POS Tagging

```
from nltk import pos_tag


sent2 = "Hello everyone welcome to my channel"


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
```

```
token = word_tokenize(sent) + word_tokenize(sent2)
tagged = pos_tag(cleaned_token)
print(tagged)
```

```
[('Twinkle', 'NNP'), ('twinkle', 'VBD'), ('little', 'JJ'), ('star', 'NN'), ('I', 'PRP'), ('wonder', 'VBP')]
```