```
pip install nltk
```

```
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) (8.1.7)
Requirement already satisfied: joblib in /usr/local/lib/python3.10/dist-packages (from nltk) (1.3.2)
Requirement already satisfied: regex>=2021.8.3 in /usr/local/lib/python3.10/dist-packages (from nltk) (2023.12.25)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from nltk) (4.66.2)
```

```
import nltk
```

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

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

```
text_data = """
Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence
concerned with the interactions between computers and human language, in particular how to program computers to
process and analyze large amounts of natural language data. Challenges in natural language processing frequently
involve speech recognition, natural language understanding, and natural language generation."""
```

```
from nltk.tokenize import RegexpTokenizer
```

```
from nltk.stem import PorterStemmer, WordNetLemmatizer
from nltk.corpus import stopwords
from nltk.corpus import wordnet
```

```
tokenizer = RegexpTokenizer(r'\w+')
tokens = tokenizer.tokenize(text_data)
```

```
stemmer = PorterStemmer()
stemmed_words = [stemmer.stem(word) for word in tokens]
```

```
lemmatizer = WordNetLemmatizer()
lemmatized_words = [lemmatizer.lemmatize(word, wordnet.VERB) for word in tokens]
```

```
stop_words = set(stopwords.words('english'))
filtered_tokens = [word for word in tokens if word.lower() not in stop_words]
```

```
print("Original Text:")
print(text_data)
```

```
Original Text:

Natural language processing (NLP) is a subfield of linguistics, computer science, and artificial intelligence
concerned with the interactions between computers and human language, in particular how to program computers to
process and analyze large amounts of natural language data. Challenges in natural language processing frequently
involve speech recognition, natural language understanding, and natural language generation.
```

```
print("\nTokenized Text:")
print(tokens)
```

```
Tokenized Text:
['Natural', 'language', 'processing', 'NLP', 'is', 'a', 'subfield', 'of', 'linguistics', 'computer', 'science', 'and', 'arti
```

```
print("\nStemmed Text:")
print(stemmed_words)
```

```
Stemmed Text:
['natur', 'languag', 'process', 'nlp', 'is', 'a', 'subfield', 'of', 'linguist', 'comput', 'scienc', 'and', 'artifici', 'inte
```

```
print("\nLemmatized Text:")
print(lemmatized_words)
```

```
Lemmatized Text:
['Natural', 'language', 'process', 'NLP', 'be', 'a', 'subfield', 'of', 'linguistics', 'computer', 'science', 'and', 'artific
```

```
print("\nFiltered Text (Removing stopwords):")
print(filtered_tokens)
```

```
Filtered Text (Removing stopwords):
['Natural', 'language', 'processing', 'NLP', 'subfield', 'linguistics', 'computer', 'science', 'artificial', 'intelligence',
```

an other example

```
#new data
data="""Assignment 3: NLP Task with NLTK

Task:
Preprocess a text dataset using NLTK.
Perform stemming and lemmatization.
Tokenize the text using regexp tokenizer.
"""
```

```
tokenizer = RegexpTokenizer(r'\w+')
tokens = tokenizer.tokenize(text_data)
print(tokens)
```

```
['Natural', 'language', 'processing', 'NLP', 'is', 'a', 'subfield', 'of', 'linguistics', 'computer', 'science', 'and', 'arti
```

```
stemmer = PorterStemmer()
stemmed_words = [stemmer.stem(word) for word in tokens]
print(stemmed_words)
```

```
['natur', 'languag', 'process', 'nlp', 'is', 'a', 'subfield', 'of', 'linguist', 'comput', 'scienc', 'and', 'artifici', 'inte
```

```
lemmatizer = WordNetLemmatizer()
lemmatized_words = [lemmatizer.lemmatize(word, wordnet.VERB) for word in tokens]
print(lemmatized_words)
```

```
['Natural', 'language', 'process', 'NLP', 'be', 'a', 'subfield', 'of', 'linguistics', 'computer', 'science', 'and', 'artific
```

Start coding or generate with AI.

```
stop_words = set(stopwords.words('english'))
filtered_tokens = [word for word in tokens if word.lower() not in stop_words]
print(filtered_tokens)
```

```
['Natural', 'language', 'processing', 'NLP', 'subfield', 'linguistics', 'computer', 'science', 'artificial', 'intelligence',
```