```
#Import libraries
import nltk
from nltk import word_tokenize
from nltk.util import ngrams
from collections import Counter
```

```
# Download tokenizer (only once needed)
nltk.download('punkt')
nltk.download('punkt_tab')
```

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

```
# Sample text
text = "I love NLP and I love AI"

# Tokenize the text into words
tokens = word_tokenize(text)

# ---- Unigrams ----
unigrams = list(ngrams(tokens, 1))
unigram_freq = Counter(unigrams)

# ---- Bigrams ----
bigrams = list(ngrams(tokens, 2))
bigram_freq = Counter(bigrams)

# ---- Trigrams ----
trigrams = list(ngrams(tokens, 3))
trigram_freq = Counter(trigrams)

# Display results
print("Tokens:", tokens)
print("\nUnigrams:", unigrams)
print("Unigram Frequencies:", unigram_freq)

print("\nBigrams:", bigrams)
print("Bigram Frequencies:", bigram_freq)

print("\nTrigrams:", trigrams)
print("Trigram Frequencies:", trigram_freq)
```

```
Tokens: ['I', 'love', 'NLP', 'and', 'I', 'love', 'AI']

Unigrams: [('I',), ('love',), ('NLP',), ('and',), ('I',), ('love',), ('AI',)]
Unigram Frequencies: Counter({('I',): 2, ('love',): 2, ('NLP',): 1, ('and',): 1, ('AI',): 1})

Bigrams: [('I', 'love'), ('love', 'NLP'), ('NLP', 'and'), ('and', 'I'), ('I', 'love'), ('love', 'AI')]
Bigram Frequencies: Counter({('I', 'love'): 2, ('love', 'NLP'): 1, ('NLP', 'and'): 1, ('and', 'I'): 1, ('l

Trigrams: [('I', 'love', 'NLP'), ('love', 'NLP', 'and'), ('NLP', 'and', 'I'), ('and', 'I', 'love'), ('I',
Trigram Frequencies: Counter({('I', 'love', 'NLP'): 1, ('love', 'NLP', 'and'): 1, ('NLP', 'and', 'I'): 1,
```