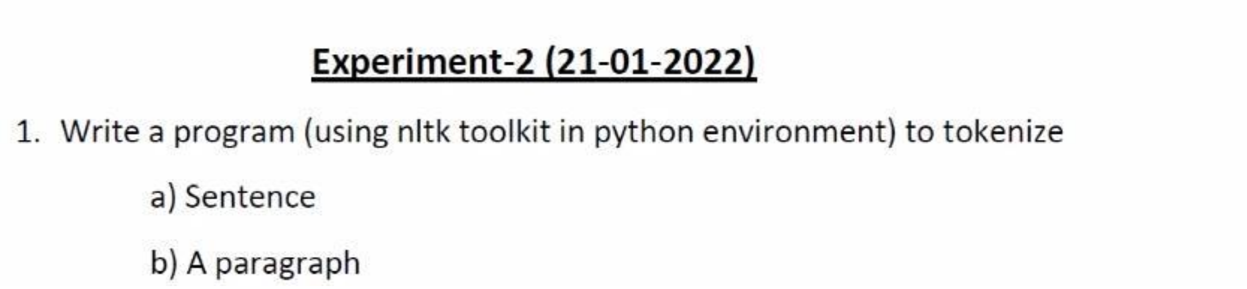
CSE-3024 Web Mining

Lab Assignment 2

Alokam Nikhitha

19BCE2555

Question



Experiment-2 (21-01-2022)

1. Write a program (using nltk toolkit in python environment) to tokenize

   a) Sentence

   b) A paragraph

Problem statement:

To write a program to tokenize using nltk toolkit in Python Environment.

Procedure:

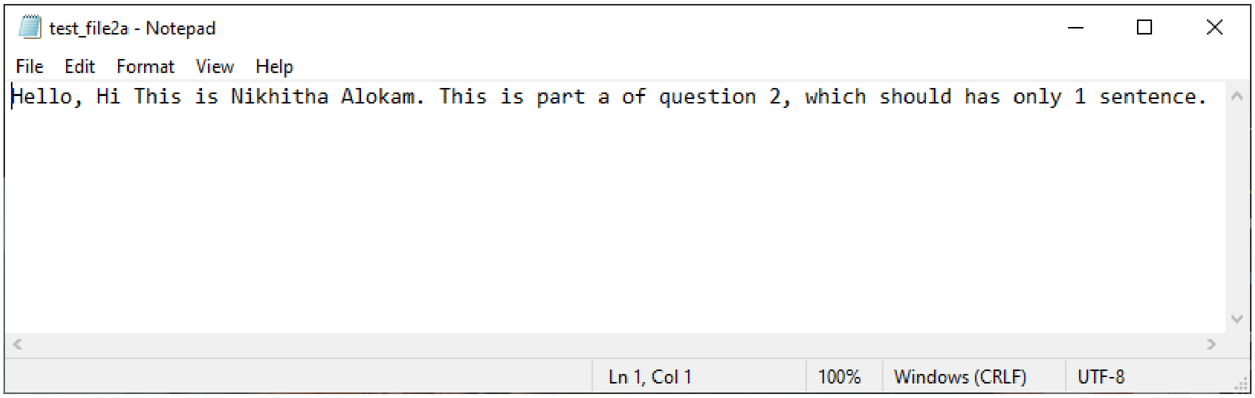
- We firstly import our text file into our workspace. To **do that we are able to** use open **method** of python which will read our **text file** to the workspace.
- **Next, we will import the necessary NLTK libraries including stopwords, sent_tokenize and word_tokenize.**
- **Using word_tokenize we tokenize each word and store it in a variable list named tokens**
- Then, we will **read every word in input** file as a string **input**. This **may be carried out using** nested for loop **in which** we **split every word whenever** we **come across** a space.
- Then , **we use** regex to **remove** the punctuations from the **input** string. This will render token a **higher** syntactic **shape** and **break** the sentence bonds.
- Then ,we **split** and **store every** token **right into a list with** nltk's sentence_tokenize **method**.
- Then **subsequently** we **remove stop words** in the same process as in **preceding** assignment.
- **Finally, we print our list that contains the resultant tokens post removal of stop words as well.**

a) Sentence

Code:

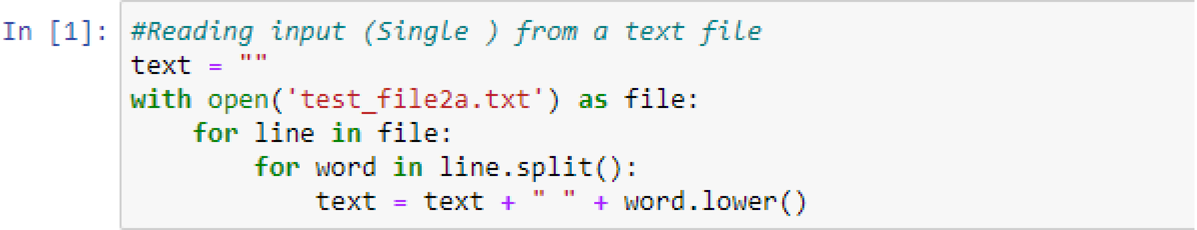
```
#Reading input (Single ) from a text file
text = ""
with open('test_file2a.txt') as file:
    for line in file:
        for word in line.split():
            text = text + " " + word.lower()

#Importing libraries
import re
import nltk
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords

#Removing punctuations from our input .
text = re.sub(r'[^\w\s]', '', text)
text

#Printing each token
tokens = word_tokenize(text)
print(tokens)

#Printing unique tokens
import numpy as np
tokens = np.unique(tokens)
print(tokens)

#Removing Stopwords
res = []
for x in tokens:
    if x not in set(stopwords.words('english')):
        res.append(x)

#Printing cleaned tokens in our input text file
print(res)
```

Text File Taken as Input:



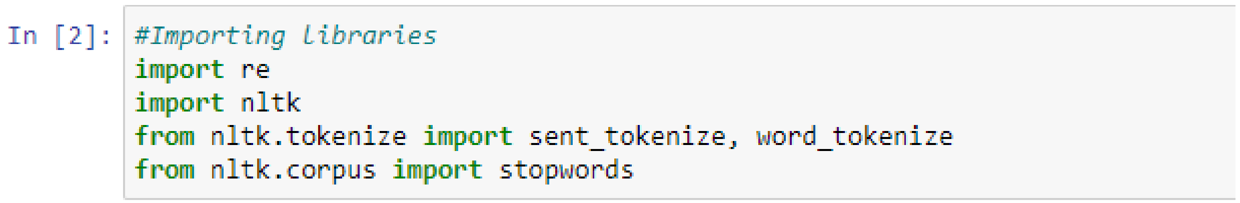
```
Hello, Hi This is Nikhitha Alokam. This is part a of question 2, which should has only 1 sentence.
```

Code Snippets and Outputs:



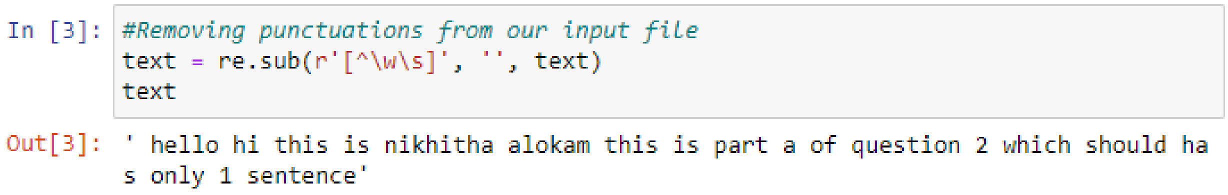
```
#Reading input (Single ) from a text file
text = ""
with open('test_file2a.txt') as file:
    for line in file:
        for word in line.split():
            text = text + " " + word.lower()
```

Here **we're analyzing** the **text file by using** open **approach** in python. Then **analyzing every** line we **split every word** and append it to a string variable with a **space** in between.



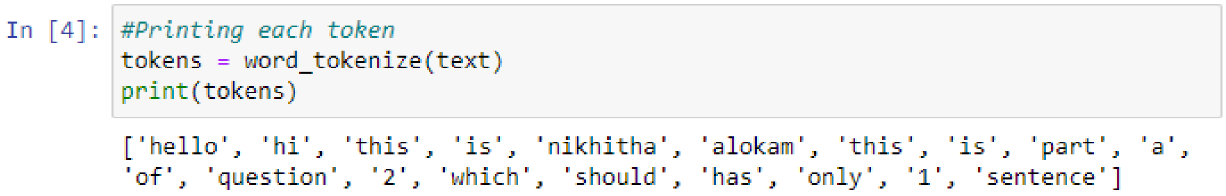
```
In [2]: #Importing libraries
        import re
        import nltk
        from nltk.tokenize import sent_tokenize, word_tokenize
        from nltk.corpus import stopwords
```

Here we're importing the necessary libraries which incorporates our library of challenge this is nltk. We additionally import stopwords and word_tokenize, sentence_tokenize from nltk. To eliminate punctuations we import the regex library.



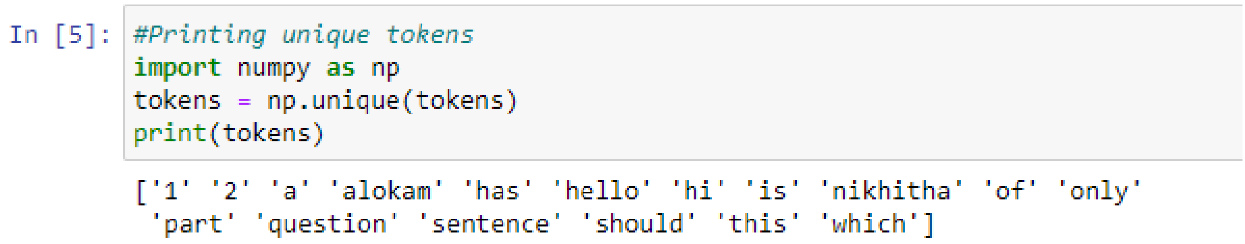
```
In [3]: #Removing punctuations from our input file
        text = re.sub(r'[^\w\s]', '', text)
        text

Out[3]: ' hello hi this is nikhitha alokam this is part a of question 2 which should ha
        s only 1 sentence'
```

Here we're removing punctuations from the input taken from input file. This is carried out by using regex, wherein we keep only alphanumeric inputs in our text string. We can see all of the intervals and commas from original input file are eliminated here.



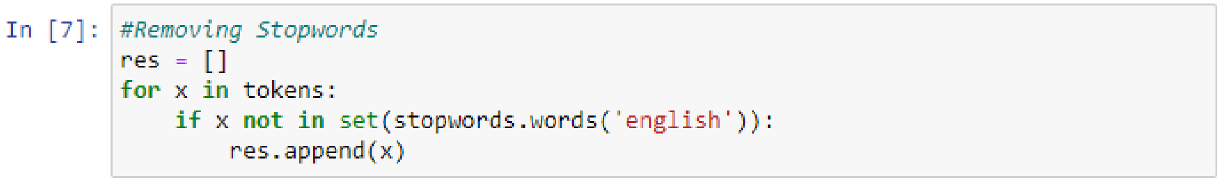
```
In [4]: #Printing each token
        tokens = word_tokenize(text)
        print(tokens)

        ['hello', 'hi', 'this', 'is', 'nikhitha', 'alokam', 'this', 'is', 'part', 'a',
        'of', 'question', '2', 'which', 'should', 'has', 'only', '1', 'sentence']
```

Since we have to tokenize every word, we have used the word_tokenize and as we can see each token in identified here and we print them.

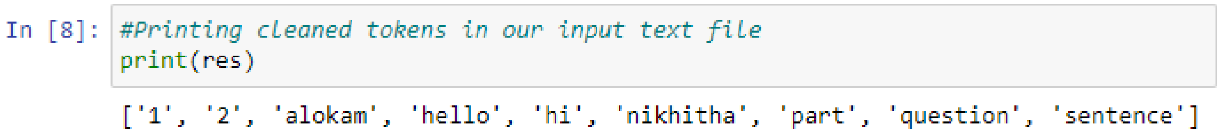


```
In [5]: #Printing unique tokens
        import numpy as np
        tokens = np.unique(tokens)
        print(tokens)

        ['1' '2' 'a' 'alokam' 'has' 'hello' 'hi' 'is' 'nikhitha' 'of' 'only'
         'part' 'question' 'sentence' 'should' 'this' 'which']
```

Here we print all the unique tokens in after tokenizing. And print them.



```
In [7]: #Removing Stopwords
        res = []
        for x in tokens:
            if x not in set(stopwords.words('english')):
                res.append(x)
```

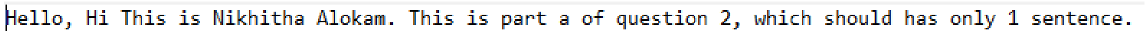


```
In [8]: #Printing cleaned tokens in our input text file
        print(res)

        ['1', '2', 'alokam', 'hello', 'hi', 'nikhitha', 'part', 'question', 'sentence']
```

Here we remove all the stop words from a self-defined list of stop words. We use nested loop to check if given token belongs to both tokens list and stopwords list. If it does, we don’t add it to our result else we add it to our results.

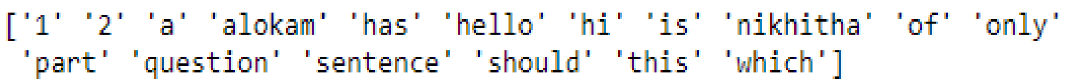
Results and Output

1. Input Sentence



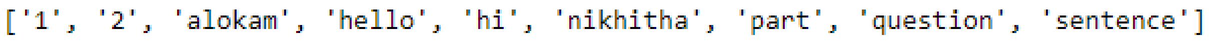
```
Hello, Hi This is Nikhitha Alokam. This is part a of question 2, which should has only 1 sentence.
```

2. Tokens with out removing Stopwords.



```
['1' '2' 'a' 'alokam' 'has' 'hello' 'hi' 'is' 'nikhitha' 'of' 'only'
 'part' 'question' 'sentence' 'should' 'this' 'which']
```

3. Tokens after removing Stopwords



```
['1', '2', 'alokam', 'hello', 'hi', 'nikhitha', 'part', 'question', 'sentence']
```

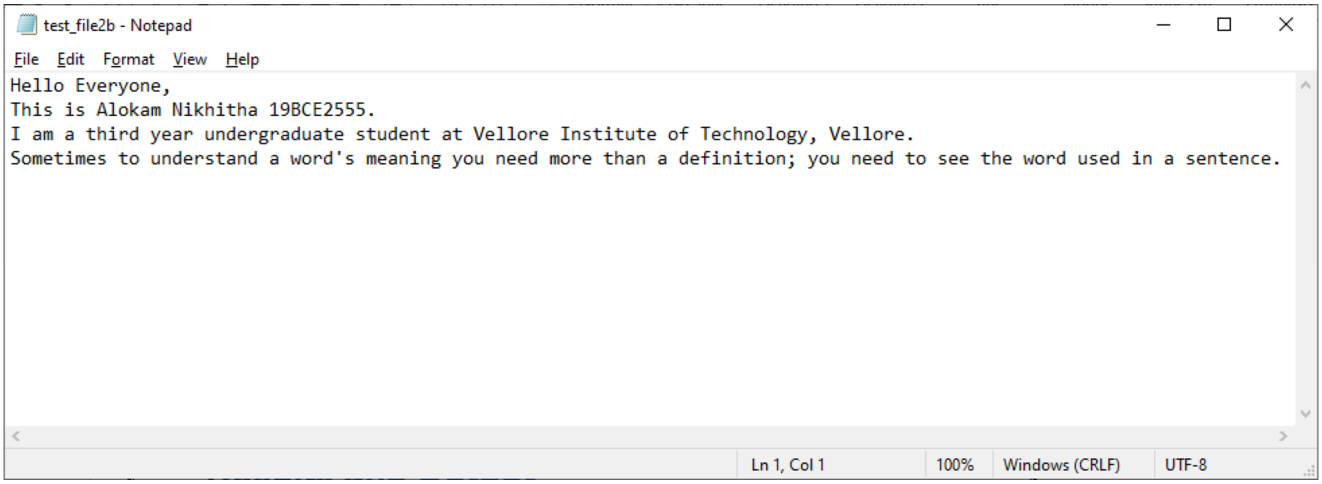
b) A Paragraph

Code:

```
#Reading input (Single ) from a text file
text = ""
with open('test_file2b.txt') as file:
    for line in file:
        for word in line.split():
            text = text + " " + word.lower()

#Importing libraries
import re
import nltk
from nltk.tokenize import sent_tokenize, word_tokenize
from nltk.corpus import stopwords

#Removing punctuations from our input .
text = re.sub(r'[^\w\s]', '', text)
text

#Printing each token
tokens = word_tokenize(text)
print(tokens)

#Printing unique tokens
import numpy as np
tokens = np.unique(tokens)
print(tokens)

#Removing Stopwords
res = []
for x in tokens:
    if x not in set(stopwords.words('english')):
```

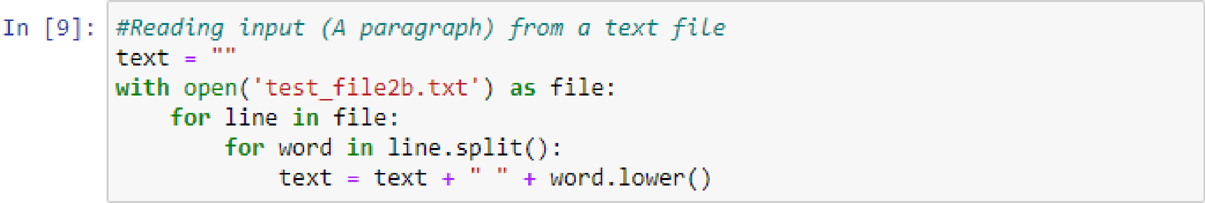
```
    res.append(x)

#Printing cleaned tokens in our input text file
print(res)
```

Text File Taken as Input:



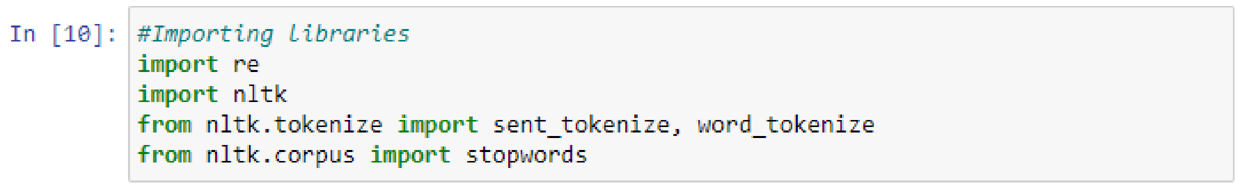
```
Hello Everyone,
This is Alokam Nikhitha 19BCE2555.
I am a third year undergraduate student at Vellore Institute of Technology, Vellore.
Sometimes to understand a word's meaning you need more than a definition; you need to see the word used in a sentence.
```

Code Snippets and Outputs:



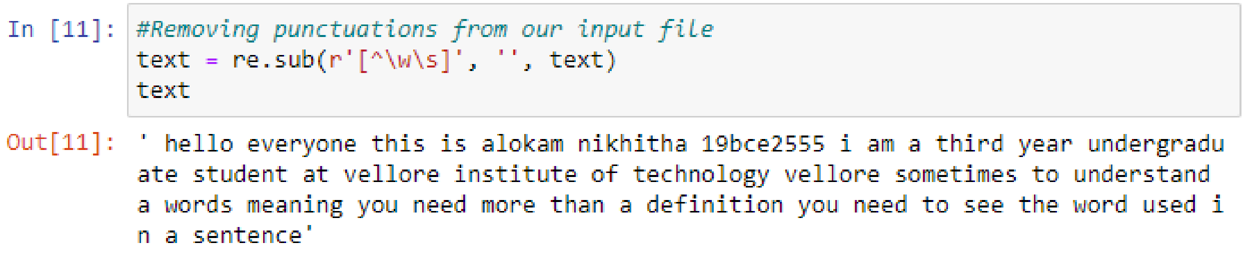
```
In [9]: #Reading input (A paragraph) from a text file
        text = ""
        with open('test_file2b.txt') as file:
            for line in file:
                for word in line.split():
                    text = text + " " + word.lower()
```

Here we're analyzing the text file by using open approach in python. Then analyzing every line we split every word and append it to a string variable with a space in between.



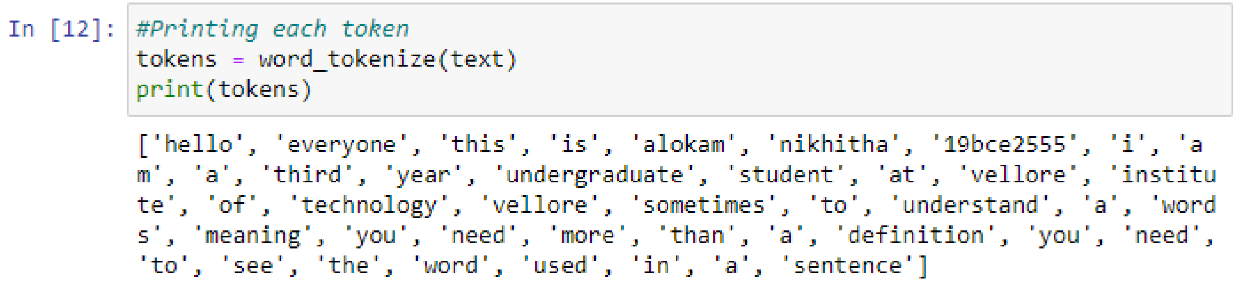
```
In [10]: #Importing libraries
         import re
         import nltk
         from nltk.tokenize import sent_tokenize, word_tokenize
         from nltk.corpus import stopwords
```

Here we're importing the necessary libraries which incorporates our library of challenge this is nltk. We additionally import stopwords and word_tokenize, sentence_tokenize from nltk. To eliminate punctuations we import the regex library



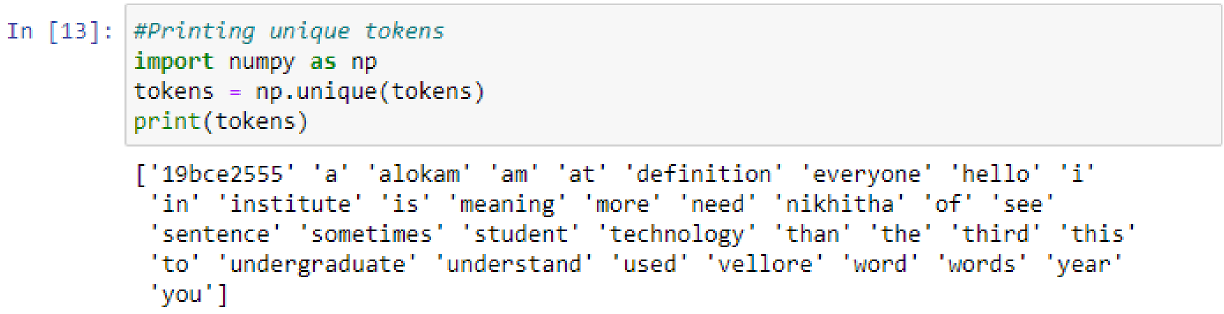
```
In [11]: #Removing punctuations from our input file
         text = re.sub(r'[^\w\s]', '', text)
         text

Out[11]: ' hello everyone this is alokam nikhitha 19bce2555 i am a third year undergradu
         ate student at vellore institute of technology vellore sometimes to understand
         a words meaning you need more than a definition you need to see the word used i
         n a sentence'
```

Here we're removing punctuations from the input taken from input file. This is carried out by using regex, wherein we keep only alphanumeric inputs in our text string. We can see all of the intervals and commas from original input file are eliminated here.



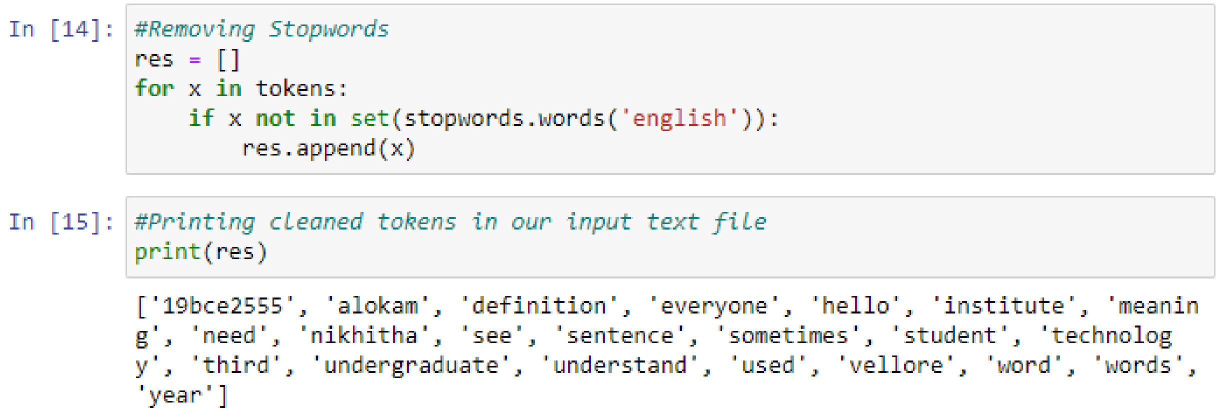
```
In [12]: #Printing each token
         tokens = word_tokenize(text)
         print(tokens)

         ['hello', 'everyone', 'this', 'is', 'alokam', 'nikhitha', '19bce2555', 'i', 'a
         m', 'a', 'third', 'year', 'undergraduate', 'student', 'at', 'vellore', 'institu
         te', 'of', 'technology', 'vellore', 'sometimes', 'to', 'understand', 'a', 'word
         s', 'meaning', 'you', 'need', 'more', 'than', 'a', 'definition', 'you', 'need',
         'to', 'see', 'the', 'word', 'used', 'in', 'a', 'sentence']
```

Since we have to tokenize every word, we have used the word_tokenize and as we can see each token in identified here and we print them.



```
In [13]: #Printing unique tokens
         import numpy as np
         tokens = np.unique(tokens)
         print(tokens)

         ['19bce2555' 'a' 'alokam' 'am' 'at' 'definition' 'everyone' 'hello' 'i'
          'in' 'institute' 'is' 'meaning' 'more' 'need' 'nikhitha' 'of' 'see'
          'sentence' 'sometimes' 'student' 'technology' 'than' 'the' 'third' 'this'
          'to' 'undergraduate' 'understand' 'used' 'vellore' 'word' 'words' 'year'
          'you']
```

Here we print all the unique tokens in after tokenizing. And print them.

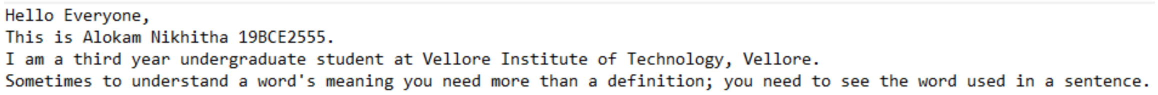


```
In [14]: #Removing Stopwords
         res = []
         for x in tokens:
             if x not in set(stopwords.words('english')):
                 res.append(x)

In [15]: #Printing cleaned tokens in our input text file
         print(res)

         ['19bce2555', 'alokam', 'definition', 'everyone', 'hello', 'institute', 'meanin
         g', 'need', 'nikhitha', 'see', 'sentence', 'sometimes', 'student', 'technolog
         y', 'third', 'undergraduate', 'understand', 'used', 'vellore', 'word', 'words',
         'year']
```

Here we remove all the stop words from a self-defined list of stop words. We use nested loop to check if given token belongs to both tokens list and stopwords list. If it does, we don’t add it to our result else we add it to our results.

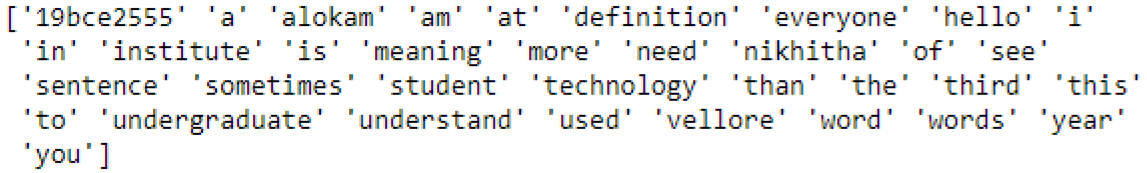
Results and Output

1.Input Sentence



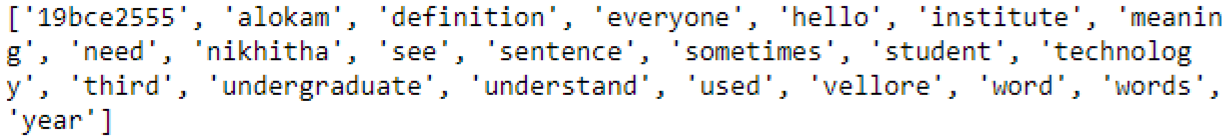
```
Hello Everyone,
This is Alokam Nikhitha 19BCE2555.
I am a third year undergraduate student at Vellore Institute of Technology, Vellore.
Sometimes to understand a word's meaning you need more than a definition; you need to see the word used in a sentence.
```

2.Tokens with out removing Stopwords.



```
['19bce2555' 'a' 'alokam' 'am' 'at' 'definition' 'everyone' 'hello' 'i'
 'in' 'institute' 'is' 'meaning' 'more' 'need' 'nikhitha' 'of' 'see'
 'sentence' 'sometimes' 'student' 'technology' 'than' 'the' 'third' 'this'
 'to' 'undergraduate' 'understand' 'used' 'vellore' 'word' 'words' 'year'
 'you']
```

3.Tokens after removing Stopwords



```
['19bce2555', 'alokam', 'definition', 'everyone', 'hello', 'institute', 'meanin
g', 'need', 'nikhitha', 'see', 'sentence', 'sometimes', 'student', 'technolog
y', 'third', 'undergraduate', 'understand', 'used', 'vellore', 'word', 'words',
'year']
```