**EXPERIMENT NO.2**

**AIM:** Stop word removal.

**RESOURCES REQUIRED:**

Python 3, NLTK toolkit, Text editor, 4 GB RAM and above, i5 processor and above

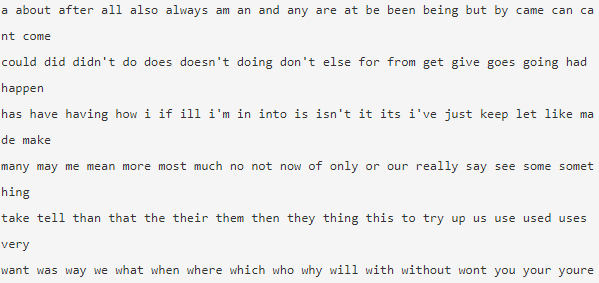
**THEORY:**

**STOP WORD REMOVAL:**

Stop words are the most common words in any natural language. For the purpose of analyzing text data and building NLP models, these stop words might not add much value to the meaning of the document.

Consider this text string – “There is a pen on the table”. Now, the words “is”, “a”, “on” and “the” add no meaning to the statement while parsing it. Whereas words like “there”, “book”, and “table” are the keywords and tell us what the statement is all about.

A basic list of stop words is given below:



Removing stop words is not a hard and fast rule in NLP. It depends upon the task that we are working on. For tasks like text classification, where the text is to be classified into different categories, stop words are removed or excluded from the given text so that more focus can be given to those words which define the meaning of the text.

A few key benefits of removing stop words:

* On removing stop words, dataset size decreases and the time to train the model also decreases
* Removing stop words can potentially help improve the performance as there are fewer and only meaningful tokens left. Thus, it could increase classification accuracy
* Even search engines like Google remove stop words for fast and relevant retrieval of data from the database.

We can remove stop words while performing the following tasks:

Text Classification

Spam Filtering

Language Classification

Genre Classification

Caption Generation

Auto-Tag Generation

**CONCLUSION:**

Stop word removal is a pre-processing task in natural language processing. Stop word removal is necessary to improve analysis of the corpora in use. Stop word removal helps to understand relationships between the elements of the text and extract features. Stop word removal has been performed on a simple text corpus.

**CODE:**

from random import choice

import nltk

from nltk.corpus import stopwords, brown

samples = choice(brown.paras(categories="fiction"))

corpus = " ".join([" ".join(sample) for sample in samples])

print(f"Original corpus :\n{corpus}\n")

tokens = nltk.word\_tokenize(corpus)

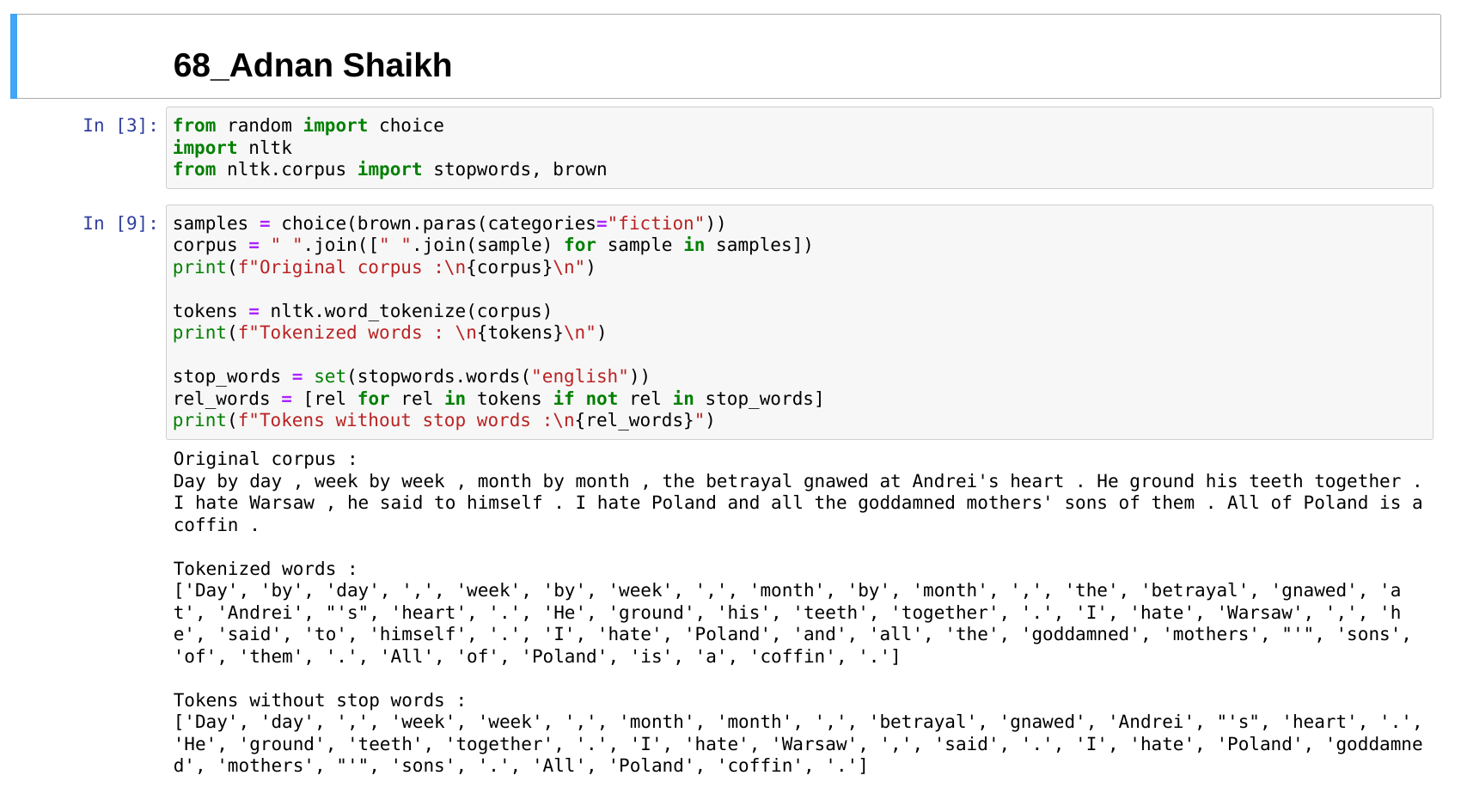
print(f"Tokenized words : \n{tokens}\n")

stop\_words = set(stopwords.words("english"))

rel\_words = [rel for rel in tokens if not rel in stop\_words]

print(f"Tokens without stop words :\n{rel\_words}")

**OUTPUT:**

****