Problem Statement -:

1. Extract Sample document and apply following document preprocessing methods:
   Tokenization, POS Tagging, stopwords removal, stemming and Lemmatization.
2. Create representation of document by calculating term frequency and Inverse Document frequency

objectives -:

- To protect sensitive data while preserving its business utility.

- labelling each word in a sentance with its appropriate part of speech.

Theory -:

i] Tokenization -:

- Tokenization is the process of dividing text into a set of meaningful pieces. these pieces are called tokens.
  Ex-:- we can divide a chunk of text into words, or we can divide it into sentences.

- Depending on the task at hand, we can define our own conditions to divide the input text into meaningful tokens.

```
From nltk.tokenize import word_tokenize

sentence = "Books are on the table"
words = word_tokenize (sentence)
print(words)
```

output : ['Books', 'are', 'on', 'the', 'table']

ii] POS Tagging -:

- POS Tagging (Part of speech) is a process to make the words in text format for a particular part of speech based on it's defining and context
- It is responsible for text reading in a language and assuming some specific word token.
- Let's learn with a NLTK POS example :

Input:

Everything to permit us

output:

[('Everything', NN), ('to', TO), ('Permit', VB, ('us', PRP)].

iii] Stopwords removal :-

- Stop Words : A stop word is a commonly used word (such as "the", "a", "an", "in") that a search engine has been programmed to ignore.
- When indexing entries for searching and when retrieving them as the result of a Search quer

iv] Stemming :-

- Much of natural language machine learning is about sentiment of the text.
- Stemming is a process where words are reduced to a root by removing inflection through dropping unnecessary characters, usually a suffix. there are several stemming models, including Porter & snowball

Ex:-

IN: ["It never once occured."]

OUT : ['it', 'never', 'onc', 'occur']

v] Lemmazation :-

- Lemmazation is an alternative approach from stemming to removing inflection.
- By determining the part of speech and utilizing WordNet's lexical database of English, lemmazation can get better results.

Ex-:

The stemmed form of leafs is : leaf
The lemmatized form of leafs is : leaf.

- How to calculating term frequency and inverse Document frequency.

TF-IDF for a word in a document is calculated by multiplying two different metrics:

- The term frequency of a word in a document. there are several ways of calculating this frequency, with the simplest being a raw count of instances a word appears in a document.
- Then, there are ways to adjust the frequency, by length of a document, or by the raw frequency of the most frequency word in a document.
- The inverse document frequency of the word across a set of documents. this means, how common or rare a word is in the entire document set.
- The clear it is to '0', the more common a word is. this metric can be calculated by taking the total number of documents, dividing it by the number of documents that contain a word, and calculating the logarithm.

To put it in more formal mathematical terms, the TF - IDF score of the word t in the document d from the document set D is calculated as follows :

$$tf\ idf(t, d, D) = tf(t, d) . idf(t, D)$$

Where :

$$tf(t, d) = \log(1 + freq(t, d))$$

$$idf(t, D) = \log\left(\frac{N}{count(d \in D : t \in d)}\right)$$

Result -: In this way, we can study about preprocessing methods and representation of document by calculating TF & IDF.