**Pre Processing Steps for text**

1. Importing the necessary libraries:
   * The **pandas** library is imported for data manipulation.
   * The **word\_tokenize** and **sent\_tokenize** functions are imported from the NLTK library for text tokenization.
2. Reading the CSV file into a DataFrame:
   * The code reads the data from a CSV file named **'nature.csv'** into a pandas DataFrame called **df**.
3. Tokenizing the DataFrame:
   * The **word\_tokenize** function is applied to the DataFrame, converting it into a list of words.
   * The resulting list of words is stored in a variable called **x**.
4. Stemming the tokens:
   * The code imports the **PorterStemmer** class from the NLTK library, which is a widely used stemming algorithm.
   * An instance of the **PorterStemmer** class is created.
   * The stemmer's **stem** method is applied to each word in the list of tokens (**x**), resulting in a new list of stemmed words.
   * The list of stemmed words is stored in a variable called **v**.
5. Lemmatizing the tokens:
   * The code imports the **WordNetLemmatizer** class from the NLTK library.
   * An instance of the **WordNetLemmatizer** class is created.
   * The lemmatizer's **lemmatize** method is applied to each word in the list of stemmed words (**v**).
   * Lemmatization attempts to reduce each word to its base or dictionary form.
   * The resulting list of lemmas is stored in a variable called **w**.
6. Removing stopwords:
   * The code imports the **stopwords** corpus from the NLTK library, which contains common words that often carry little meaning.
   * A list comprehension is used to filter out any words from the lemmatized list (**w**) that appear in the NLTK stopwords list.
   * The resulting list of words without stopwords is stored in a variable called **sw**.