API Documentation

Function	Function Description	Inputs / Outputs
def read_file(filename):	This function is responsible for reading the contents of the user specified file	**Input**: filename (str) - Name of the input file specified by the user. **Output:** chars (str) - Content of the file.
def write_file(output_Filename, output_data):	This function is responsible for writing the results of this program to a new file, the name of the file will be specified by the user.	**Input:** output_Filename (str) - Name of the output file specified by the user. output_data (str) - Results of the program to be written to the output file. **Output:** None / returns the results of the program in the output file
def wordCount(chars):	This function counts the number of total words in the user-specified file.	**Input:** chars (str) - Content of the input file. **Output:** wordCount (int) - The total count of words in the file.
def num_sentences(chars):	This function counts the total number of sentences in the file, considering periods and question marks as sentence delimiters.	**Input:** chars (str) - Content of the input file. **Output:** num_of_sent (int) - The total number of sentences in the input file.
def frequency(chars):	This function counts the total number of times each word appears in the input text file, considering word frequencies and removing stopwords and punctuation.	**Input:** chars (str) - Content of the input file. **Output:** word_freq (dict) - The total number of times each word appears in the input file.
def user_word_frequency(chars):	This function counts the total number of times a specific word, which is specified by the user, appears in the file.	**Input:** chars (str) - Content of the input file. **Output:** freq_of_word (str) - The frequency of the word specified by the user.
def frequent_words(chars):	This function counts the total number of times a word appears in the file, sorts them in descending order, and outputs the top	**Input:** chars (str) - Content of the input file. **Output:** df (DataFrame) - A DataFrame consisting of the top 10 words that occurred

	10 most occurring words.	most frequently in the file.
def termFrequency_idf(chars):	This function calculates the TF-IDF value for each word in the document and displays it as a DataFrame.	**Input:** chars (str) - Content of the input file. **Output**: tfidf_df (DataFrame) - A DataFrame containing the Term Frequency-Inverse Document Frequency (TF-IDF) of each word in the document. Also, a new file called **TermFrequcny.csv** will be created and it will contain the Term Frequency - Inverse Document Frequency (TF-IDF) of each word in the document.
def avg_words_in_sent(chars):	This function calculates the average number of words in a sentence in the input file, returning 0 if the file is empty.	**Input:** chars (str) - Content of the input file. **Output**: avg_word_count (float) - The average number of words in a sentence in the entire file.
def longest_word(chars):	This function returns the longest word in the text file.	**Input**: chars (str) - Content of the input file. **Output:** longest (str) - The longest word in the input file.
def length_longest_word(chars):	This function returns the length of the longest word in the text file.	**Input:** chars (str) - Content of the input file. **Output**: len(longest) (int) - The length of the longest word in the input file.
def shortest_word(chars):	This function returns the shortest word in the text file.	**Input:** chars (str) - Content of the input file. **Output:** shortest_word (str) - The shortest word in the input file.
def count_chars(chars):	This function calculates the frequency of each character in the file and saves it in a DataFrame.	**Input:** chars (str) - Content of the input file. **Output:** table (Pandas DataFrame) - A DataFrame containing the frequency of each character in the file and saves it in a DataFrame. Also, a new file called **alphabet_frequcny.csv** will be created and it will contain the frequency of each "character" in the file.