Rushabh Shah
Andrew ID: rushabhs

DOCUMENTATION

This program has 3 tasks, and the 2nd and 3rd tasks are built on the 1st one. So the function that will be used in the 1st task will be used in 2nd and 3rd as well, similarly functions used in task 2 will be used in task 3.

- In Task 1, the program performs web-scraping on a url and extracts all the urls from the master website using looping. Then performs scraping on each url and extracts abstract for each of the website and stores in an external text file and prints the abstract on the monitor. There are 3 functions, 1st is **parse_soup()** which works on the web scraping done in the program on the U.S. patent website. This function mainly finds all the urls on the soup response and loops through the content between table attribute to find all urls and adds all the unique urls to the list. The 2nd function is **parse_all()** takes the list of all urls and scrapes each url to only extract abstracts from the web pages. The main function calls these functions and prints the abstract on the console.
  'URL_Abstract.txt' is the file generated from this program.
- In Task 2, the program takes the string which contains all the abstracts of 15 urls and performs scrubbing on the data. The program 1st preserves compound words, then performs normalization to replace all synonyms with just one word on the result of compound words function, later it also stems the result of normalization operation to standardize verbs to
  present tense and nouns to singular forms, and at last removes punctuations and stopwords and any other cleansing required and saves the data to a text file. There are 5 functions in the program, 1st **make_dictionary()** to create a dictionary of compound words and help preserve compound words, 2nd function **perform_compoundwords()** preserves the compound words, 3rd function **perform_normalization()** replaces the synonyms for words by one word, **perform_stemming()** converts the words to their root word and standardizes, 4th function **removestopwords()** removes the noise and stopwords and the 5th function eliminates all the punctuations and 1-2 character long strings for better analysis, 6th function **print_tokens()** copies the final scrubbed data to an external file and prints the output on the console and main() function which calls all the functions and controls the flow.
  'Final_Scrubbed.txt' is the text file generated from this task and will have all the concept words

- In Task 3, the program reads the final scrubbed file created in the 2nd task , there is one function and the function **final_analysis()** mainly identifies the concept words and their frequencies and prints as well as stores the concepts and their frequencies in a text file and on the console in sorted order with the highest frequency concept as the 1st word.
  'Concept_Frequency.txt' is the text file generated from this task that contains all the unique concept words and their frequencies and is sorted in the descending order of frequency.

This is how the program holistically performs, and all the functions used have been explained.

- Now, this program can be used in a lot of different scenarios. Since, this program performs web scraping on U.S. patent website and then performs different operations on it and does scrubbing on the response, this program can be used to do web scraping on any other website and can be used to extract not only abstract but any other details from the url. So the same program can be used in lot of different situations and scrape any website needed and can be used to perform analysis of unstructured data on any website.
- Also, the code for tasks 2 and 3 can be used to do scrubbing like stemming, normalization, removal of noise and count the frequencies of words in the result for text analysis.