

Spell Checker Problem

In this exercise, you will create a simple spell checker program. Your program will have the following characteristics:

- Your program will be capable of reading multiple files to be spell checked from command line arguments.
- When your program starts up, you will read in the dictionary.txt file which contains a list of known words. Only read in this file once. Store these words in a HashSet data structure.
- Create a TreeSet to store any miss-spelled words that you find.
- For each file in the command line parameter list, print out the name of the file before reading in any words.
- For each file parse out each of the words in the file and check to see if the word is in the HashSet you created with the dictionary.txt words or its already in your list of miss-spelled words.
- If the word is in the dictionary HashSet or the miss-spelled word TreeSet then proceed to the next word.
- If the word is not in the dictionary or the miss-spelled word list, give your user 2 choices: Add the word to the HashSet dictionary, or add the word to the list of miss-spelled words.
- At the end of each file processed, dump out the list of miss-spelled words.
- When you then start to read in the next file, make sure you clear the list of miss-spelled words. Keep any words added to the dictionary HashSet.

More details:

- In parsing the words in these text files, you will need to filter out punctuation characters. I used the StringTokenizer, and I called the following constructor:

```
StringTokenizer st = new StringTokenizer(line, " \t,.;:-%'\");
```
- before checking the word against the dictionary, make sure you lower case the word
- if the "word" doesn't start out with a letter (i.e. \geq 'a' And \leq 'z'), then skip it because it is likely a number
- If the word couldn't be found in either list and ends with an 's', try removing the 's' to create a singular version of the word and try again to find the word in your dictionary or miss-spelled word list using the singular version of the word.
- Only print out lines that contain a word needing a user response. See the Sample output below from the published answer.

Use the following dictionary file

[dictionary.txt](#)

You can find the input files to test from the following links:

[Russia.txt](#)

[Ukraine.txt](#)

Your command line arguments will be:

Russia.txt Ukraine.txt

When you execute your program to publish your output, answer "n" to the question of whether to add the word to the dictionary for all words except the for the word "christianity" .

See the sample output below:

Some of the output from running this program should look something like:

[First part of the output from running the published answer](#)

For those of you who like to see a template of the published answer, here it is:

```
public class SpellCheck {  
  
    // We could have also used TreeSet for the dictionary  
    private HashSet<String> dictionary = new HashSet<String>();  
  
    private TreeSet<String> miss_spelled_words = new  
    TreeSet<String>();  
  
    public SpellCheck() throws FileNotFoundException  
    {  
        // Add all of the words from "dictionary.txt" to the  
        dictionary HashSet    }  
}
```

```

    public void checkSpelling(String fileName) throws
FileNotFoundException
    {
        System.out.println("=====  

+ " =====");

        // Clear miss_spelled_words
        miss_spelled_words.clear();

        // Read in each line from "fileName"
        // For each line, break the line into words using the
following StringTokenizer
        StringTokenizer st = new StringTokenizer(line, " \t,.;:-
%'\");

        // lower case each word obtained from the StringTokenizer
        // skip word if the first character is not a letter

        // Skip over word if it can be found in either
dictionary, or miss_spelled_words
        // If word ends with 's', then try the singular version
of the word in the dictionary and miss_spelled_words ... skip if
found

        // Print line containing miss-spelled word(make sure you
only print it once if multiple miss-spelled words are found on
this line)

        // Ask the user if he wants the word added to the
dictionary or the miss-spelled words and add word as specified
by the user

    }

    public void dump_miss_spelled_words()
    {
        // Print out the miss-spelled words
    }

    public static void main(String[] args) {

        try {
            SpellCheck spellCheck = new SpellCheck();

```

```
        for (int i=0; i < args.length; i++)
        {
            spellCheck.checkSpelling(args[i]);
            spellCheck.dump_miss_spelled_words();
        }
    }
    catch (FileNotFoundException e)
    {
        System.out.println(e);
    }
}
}
```