Assignment 11: File Input and Output Processing

Learning Objectives

Upon completion of this assignment, the student shall be able to:

- Use the Scanner class to read data from a text file.
- Use the PrintWriter class to write data to a text file.
- Use a 2D array.

Evaluation

This assignment shall be graded.

- A rubric has been created in Canvas that identifies the criteria that shall be used to evaluate your assignment submission.

Description

Follow the steps below to complete this assignment.

1. Develop a solution that adheres to the following requirements.

    a. Open the file `LinesOfTokens.txt` using the Scanner class. This file is formatted as follows:
        i. Each text line may contain from zero (0) to twelve (12) tokens. These tokens are delineated with one or more spaces.
        ii. There are no more than 20 text lines in the input file.
        iii. An example of what a text line in this file might look like is below.

```
10   twenty 30.5    star-fish   -5
1   two 3
```

    b. Store the tokens found on each text line in a 2D array.
        i. This 2D array must store String objects.
        ii. Using the example text lines shown above. The first text line would result in the five tokens being stored in locations [0][0], [0][1] , [0][2] , [0][3], and [0][4]. The second text line would result in the three tokens being stored in locations [1][0], [1][1], and [1][2].
    c. Once tokens in the input file have been stored in the 2D array, do the following:
        i. For each row in the 2D array, sum the length of tokens found in the row of tokens.
        ii. Multiply this sum by the number of tokens in the row.
    d. Create an output file named `TokensAndProducts.txt` using the PrintWriter class. Write tokens to this output file as follows.
        i. Write tokens in each row of the 2D array so they are formatted with one space between each token.
        ii. At the end of each text line, write the sum computed in step 1.c.i followed by the product computed in step 1.c.ii.
        iii. Be sure you put a space before the sum value and before the product value.
        iv. Using the two sample input text lines from above, an example of what this output file looks like is below.

```
10 twenty 30.5 star-fish -5 23 115
1 two 3 5 15
```

2. Submit all of your Java source code files in a **zip file**.