CSGE601021 Dasar-Dasar Pemrograman 2
Tutorial Lab 02
Counting Characters with Loops and Switch Statement

Build a Java project using your favorite IDE for this tutorial.

The file *CountChars.java* at the end of this document contains a template of a program to read in a string (a sentence or phrase) and count the number of blank spaces, each vowel (a, e, i, o, u in lower case or upper case), consonants (lower case or upper case) and the rest in the string. It needs a loop to go through the string character by character and count the characters separately. Since we know how many characters there are (the *length* of the string) we can use a **for** loop controlled by a counter.

1. Add a loop statement to the program. Inside the loop you need to access each individual character; the *charAt* method of the String class lets you do that. The assignment statement

```
ch = phrase.charAt(i);
```

   assigns the variable *ch* (type char) the character that is in index i of the String *phrase*. In your loop you can use an assignment similar to this.

2. You need to declare and initialize eight counting variables ***countA, countE, countI, countO, countU, countSpace, countConsonant, countOther***. You have to use the new version of the ***switch*** statement. The cases will be based on the value of the *ch* variable. The switch starts as follows; complete it.

```
switch (ch) {
   case 'a', 'A' -> countA++;

   case ……….

   default -> {
       if (Character.isLetter(ch)) {
           countConsonant++;
       } else {
           ..................
       }
   }
}
```

   Note that no **break** statement is needed. If *ch* is an 'a' or 'A' the first case matches and the **countA** variable would be incremented etc.

3. Add statements to print out all of the counts.

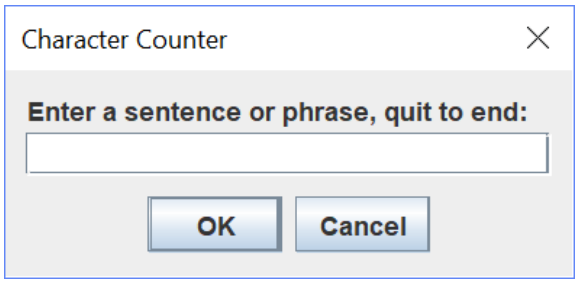
4. Study the usage of the method showInputDialog and the method showMessageDialog from the class javax.swing.JOptionPane. See Slide 06.

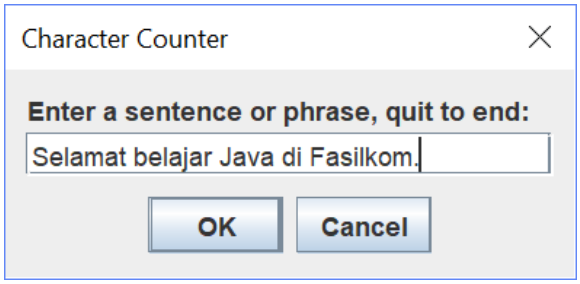
5. Test your program on several phrases to make sure it is correct.

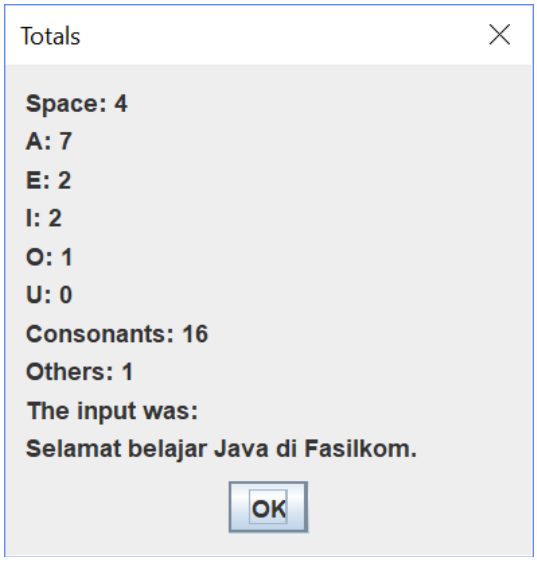
6. It would be nice to have the program let the user keep entering phrases rather than having to restart it every time. To do this we need another loop surrounding the current code. That is, the current loop will be nested inside the new loop. Add an outer while loop that will continue to execute as long as the user does NOT enter the phrase *quit*. Note that all of the initializations for the counts should be inside the while loop (that is we want the counts to start over for each new phrase entered by the user). Be sure to go through the program and properly indent your code; with nested loops the inner loop should be indented.

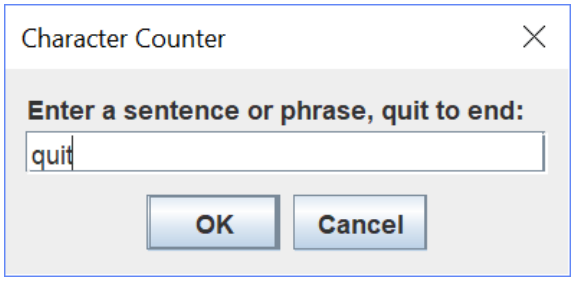
```
// *********************************************************
//   CountChars.java
//
//   This program reads in strings (phrases) and counts the
//   number of blank characters, vowels, consonants,and other
//   characters in the phrase.
// *********************************************************

package ddp2.lab02;

import javax.swing.JOptionPane;

public class CountChars {

    public static void main(String[] args) {
        String phrase;    // a string of characters

        ……….          // other variables


        // Read in a string and find its length

        phrase = JOptionPane.showInputDialog(null,
                "Enter a sentence or phrase, quit to end:\n",
                "Character Counter", JOptionPane.PLAIN_MESSAGE);

        while (……) { //while the phrase is not equal to “quit”
            length = phrase.length();

            // Initialize counts
            countSpace = 0;
            ……

            // for loop to go through the string character by character
            // and count them by category using the new switch statement

                ……


            // Print the results
            JOptionPane.showMessageDialog(null,
                    "Space: " + countSpace + "\nA: " + countA + "\nE: " +
                    countE + "\nI: " + countI + "\nO: " + countO +
                    "\nU: " + countU + "\nConsonants: " + countConsonants +
                    "\nOthers: " + countOther + "\nThe input was: \n" +
                    phrase, "Totals", JOptionPane.PLAIN_MESSAGE);


            //Get phrase for next time
            phrase = JOptionPane.showInputDialog(null,
                "Enter a sentence or phrase, quit to end:\n",
                "Character Counter", JOptionPane.PLAIN_MESSAGE);
        }
    }
}
```

Examples of program execution:









Through the link at SCeLE, submit all your project files (1 project folder), zipped into a file: lab02_<class>_<TACode>_<YourName>_<YourNPM>.zip

Marking components:

Code correctness	90%
Clear comments	10%

Selamat Mengerjakan!

'Met Ngoding! ☺

L.Y.Stefanus & the Asdos Team