THE UNIVERSITY COLLEGE
OF THE CARIBOO
Computing Science

COMP 113 & 152 - Computer Programming I (3,1,1)
Fall 2004
Lab Exercise 3

Due: *Beginning of lab session in the week of October 4 to 8.*

Self-Review Questions: *(answers are provided at the end of the chapter)*

1. page 78, questions: #18, #19
2. page 78, questions: #22, #23
3. page 87, question: #26
4. page 88, question: #32 *(**interesting)*

Preparation:

Read Section 2.2 in the text (pages **68-78**). This section covers the **String** class.
As an example of applying the **String** class methods, use the program in **Display 2.8** (page **76**).

Problems:

Create program solutions for both of the following problems. Only one will be marked.

1. Input a sentence that has the words *energetic* and *lazy* in it. Your program will exchange the locations of the 2 words as it displays the sentence.

Example 1 (*user input is in **bold***),

```
Please input the sentence:
The energetic dog jumped over the lazy cat.

The new sentence is:
The lazy dog jumped over the energetic cat.
```

Example 2 (*user input is in **bold***),

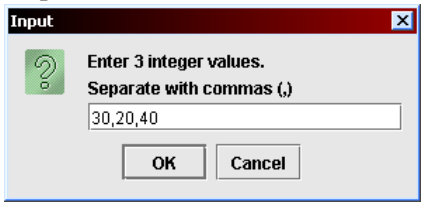
```
Please input the sentence:
a lazy energetic a

The new sentence is:
a energetic lazy a
```

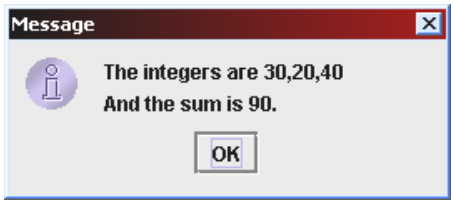
Your program must work for any input sentence that has these two words (you can assume that neither words starts the sentence). Further, the input sentence has only one occurrence of each word.

2. Using the **Integer.parseInt()** and **Double.parseDouble()** methods to translate a String to an value, GUI input almost demands the input of one value per input dialog window. By using the concepts from the problem above, from a single large input String, individual Strings can be extracted and translated to numeric values.

   Write a program that asks the user for exactly 3 integers from a single GUI input dialog, separating the numbers with a comma (,). Display sum of the numbers in an output dialog. (*hint: Use the input String from the input dialog in the output dialog.*)

   Example:





Results:

Submit a print of both programs' source code and samples of each one's output; provide output of at least two executions of each program *(use the screen capture method from the first lab).*