ISM3230 In-class lab Week 4, Module 3 – Working with Strings Fall 2015

FOCUS TOPICS

- String manipulation

TASK

Your task is to build a palindrome from an input string. A palindrome is a word that reads the same backward or forward. Your code will take the first 5 characters of the user input, and create a 9-character palindrome from it.

Words shorter than 5 characters will result in a runtime error when you run your code. This is ok for this exercise – we will cover input validation in a later class.

Some examples of input words and the resulting palindromes:

Input	Palindrome
sixty	sixtytxis
tomorrow	tomoromot
craziness	crazizarc
two	Error – too short

As input, let the user enter a word, which should be at least 5 characters or longer.

As output, report the original word, the length of the original word, and the palindrome you created based on the original word.

In your code, use only the following String class methods:

- concat(stringToAdd)
- length()
- substring(startIndex, endIndex)

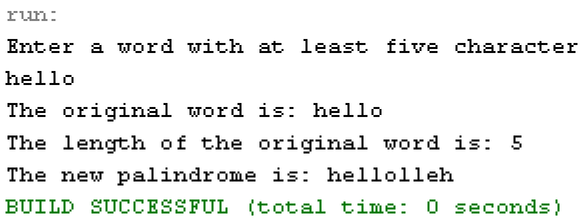
For this exercise, you may NOT use the '+' operator to concatenate Strings together.

As you work through the instructions below, refer to slides 23, 24, and 25 from today's class material for help with character indexing and the specifics on how the String methods work.

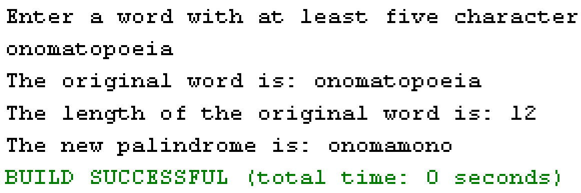
1. Create a new NetBeans project.
2. Prompt the user to enter a word that is at least 5 characters long.
3. Read the input from the keyboard, and store it in a String variable.
    - **Hint**: Use the nextLine() method of the Scanner class.
    - **Hint**: Don't forget the import statement!

4. Create a new String variable to represent the base of the output palindrome. Use the substring method to get the first five characters of the input word, and store this String value to the base palindrome variable.
5. Create four new String variables to represent the first four letters in the input word. Use the substring method to get a single-character String from the input word for each of these four letters, and store these Strings to the variables you created.
6. Create a new String variable to store the final palindrome. Concatenate the base palindrome and the letters together to form the final palindrome, and store it to your final palindrome variable.
7. Following the sample output below, report the following information, in the order below:
    - The original input word
    - The length of the original input word
    - The new palindrome
8. When your code is working, upload your .java file to the Module 3, week 4 in-class lab dropbox.

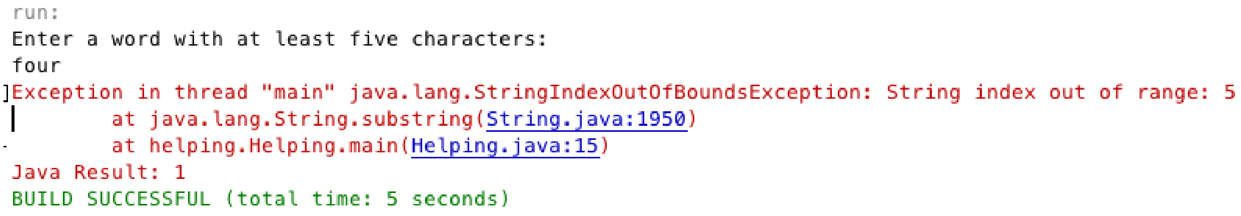
SAMPLE OUTPUT



```
run:
Enter a word with at least five characters:
hello
The original word is: hello
The length of the original word is: 5
The new palindrome is: hellolleh
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
Enter a word with at least five characters:
onomatopoeia
The original word is: onomatopoeia
The length of the original word is: 12
The new palindrome is: onomamono
BUILD SUCCESSFUL (total time: 0 seconds)
```



```
run:
Enter a word with at least five characters:
four
]Exception in thread "main" java.lang.StringIndexOutOfBoundsException: String index out of range: 5
        at java.lang.String.substring(String.java:1950)
        at helping.Helping.main(Helping.java:15)
Java Result: 1
BUILD SUCCESSFUL (total time: 5 seconds)
```