

Java Lab 2

Due: Friday, September 1, 11:00 AM EDT

In this lab, you will practice with a few of the String and StringBuilder methods.

1. Create a project named Lab2. Download the file StringStuff.java from Canvas and copy it to the src directory.
2. Run the program; enter "gopher" when prompted for a string. What do the boolean values tell you?
3. Create a new String variable s3 and set it equal to the phrase "The name of my pet ". Then:
 - display s3
 - display s3 in all upper case
 - display s3 again: has it changed? Why or why not?
 - set s2 to s3 concatenated to " " and to s1 (that is, s3 first) and display it
 - concatenate the string " is Fluffy Face" (notice the extra leading space) to s2. Display it.
 - display s2 in all lower case.
 - display s2 with all "e" characters replaced by "XYZ". Display s2 again – has it changed?
4. Create a new String variable s4 and set it equal to "Barrett,Sales,#44132,8/22/2018". Then:
 - display s4
 - create an array of strings named slist (like this: `String[] slist`) and set it equal to the result of splitting s4 on the comma character. Then display slist with this code:

```
for (String str: slist) {  
    System.out.println(str);  
}
```

How many lines were printed?

- create a new `String[]` date and set it equal to `slist[3]` split on the "/" character. Display the parts of date with similar code to the above loop.
- create a new String s5 and set it equal to the parts of slist concatenated together – the parts are accessed by `slist[0]`, `slist[1]`, etc. Display s5.

5. This problem tests the relative performance of String concatenation versus StringBuilder appending. Use this code to time 10,000 String concatenations:

```
long startTime = System.currentTimeMillis();  
for (int i = 0; i<10000; i++) { s1 += s2; }  
long endTime = System.currentTimeMillis();  
System.out.println(endTime-startTime);
```

This will show the number of milliseconds it takes to do the concatenations. Note the use of the long integer type here; it's required by the timer. Comment out the prompt for entering a string and the `scanner.next()` line so it won't annoy you. Run the program.

Create a new `StringBuilder` object sb with s1 as a parameter. Then copy the lines above and paste them back in below the `StringBuilder`; remove the keyword `long` (because you've already declared those variables); replace the concatenation of s2 to s1 with appending s2 to the `StringBuilder` object like this:

```
sb.append(s2);
```

Run the program and compare the times.

Deliverable: Add your name and Andrew id to the comment at the top of the file. Upload the file to Canvas.