

Model 1 Common Mistakes

Use *JShell* to run each of the code snippets. For Program B, do not provide any user input (just press Enter). Record the output of the last statement in the space below the table.

Program A	Program B
<pre>String greeting = "hello world"; greeting.toUpperCase(); System.out.println(greeting);</pre>	<pre>Scanner in = new Scanner(System.in); String line = in.nextLine(); char letter = line.charAt(1);</pre>
hello world	StringIndexOutOfBoundsException

Questions (15 min)

Start time:

1. What is the logic error you see when you run Program A?

The "hello world" greeting is still lowercase.

2. In Program A, what is returned by the string method? What happens to the return value?

A new string is returned, but its value is not saved anywhere.

3. Describe two different ways you can fix the logic error in #2.

```
greeting = greeting.toUpperCase(); // or just combine the two lines  
System.out.println(greeting);      System.out.println(greeting.toUpperCase());
```

4. In what cases will Program B throw an exception? What is the error message displayed?

If the user enters a blank line, then there will be no character at position 1. It will show the out of bounds exception and display the stack trace.

5. Describe two different ways you can fix the run-time error in #4.

```
1. Check if the string is long enough: if (line.length() >= 2)  
2. Keep asking for input until valid: do { ... } while (line.length() < 2);
```

6. To compare strings, you must use either the `String.equals` or `String.compareTo` method. Predict the output of the following code.

```
String name1 = new String("Mark");
String name2 = new String("Mark");

// compare name1 and name2
if (name1 == name2) {
    System.out.println("name1 and name2 are identical");
} else {
    System.out.println("name1 and name2 are NOT identical");
}
```

```
name1 and name2 are NOT identical
```

```
// compare "Mark" and "Mark"
if (name1.equals(name2)) {
    System.out.println("name1 and name2 are equal");
} else {
    System.out.println("name1 and name2 are NOT equal");
}
```

```
name1 and name2 are equal
```

7. What is the difference between *identical* and *equal* in the previous question?

“Identical” means the variables refer to the same `String` object. “Equal” means the strings have the same characters, in the `Arrays.equals` sense.