CSCA08H lab – week 5

This document contains the instructions for the week 5 CSCA08H lab.

1 Objectives

1. Learn a bit about how `JFrame` contents are organized.
2. Learn about buttons and text areas.
3. Write and use constructors.
4. Work with `String` methods.
5. Work with code someone else wrote.

2 Starting up

Sit down with your partner. The rest of these instructions call you two `s1` and `s2`. Pick which one is which. `s1` should log in and start up DrJava, and be the first driver.

Read each section with your partner before you start working on it so that you understand what is required. If there are areas that do not make sense to you, ask your TA!

3 The task

A `javax.swing.JButton` is a clickable button. A `javax.swing.JTextArea` is a typing area. Both those things are called *components* of a GUI (Graphical User Interface). Most GUI components go in the *content pane* of a `JFrame`.

The content pane of a `JFrame` is called a *container*, because it can contain components. The default content pane for a `JFrame` has five areas: north, south, east, west, and center.

Download `BorderJFrame.java` from the week 5 row of this page:

`http://www.cs.toronto.edu/~campbell/108/05w/labs.shtml`

Open it in DrJava, compile it, and in the Interactions Pane instantiate a `BorderJFrame` and `show()` it. Notice that the button names don't make sense: everything is in the wrong place.

In the `BorderJFrame` constructor, figure out why and fix it. Hint: the problems happen in the calls to `c.add(...)`.

Compile it, test it, and fix any errors. You can test it with this Interactions code:

```
new BorderJFrame().show();
```

4 Another constructor

Switch roles: `s2` drives and `s1` navigates.

Now add a `BorderJFrame` constructor that takes 4 `String`s as parameters, and uses them for the `JButton` names in the north, south, east, and west (in that order). For example, if you type this in the Interactions Pane:

```
new BorderJFrame("bn", "bs", "be", "bw").show();
```

then in the window that appears the north button will be called `bn`, the south one `bs`, and so on. In this constructor, make the text area with an empty string: `""`, rather than `"Center"`.

Compile it, test it, and fix any errors.

5 Yet another constructor

Switch roles: `s1` drives and `s2` navigates. Don't forget what we said about reading and understanding the section thoroughly before you start

For this part, you'll need to rename the instance variables in order to keep the variable names from being confusing. Search for `northButton` and replace it with `button1`, `southButton` with `button2`, `eastButton` with `button3`, `westButton` with `button4`, and `centerArea` with `textArea`.

Every call to method `add` has the item to add and the location to add it. As you have seen, the second argument is a `String`.

Add a `BorderJFrame` constructor that takes a single `String` as a parameter. This `String` has the following format, where `b1` means "button 1" (and so on), and the commas and dollar signs are separating the pieces of information:

```
b1-name,b1-loc$b2-name,b2-loc$b3-name,b3-loc$b4-name,b4-loc$text,text-loc
```

For example, with this code in the Interactions Pane (exactly as shown)

```
new BorderJFrame("Danny,North$Vasiliki,Center$Hojjat,West$Wei,East$Dong,South").show()
```

the north button would be called `Danny`, the center button `Vasiliki`, the west button `Hojjat`, the east button `Wei`, and the south text area `Dong`.

This code should produce the identical result:

```
new BorderJFrame("Dong,South$Danny,North$Hojjat,West$Wei,East$Vasiliki,Center").show()
```

Here is a list of `String` methods that you will find useful:

- `indexOf(String s)`: return the index of s in this `String`. Use to get indices of the dollars and commas.
- `substring(int i)`: return the substring from `i` to the end. Use to extract the names and locations.
- `substring(int i, int j)`: return the substring from `i` to `j`. Use to extract the names and locations.

You must use the following process:

Try to extract the first button name and location, add it to the `BorderJFrame`, and *don't write any more code.* Instead, compile it, test it, and fix any errors. There should be only one button in the window.

Switch roles: `s2` drives and `s1` navigates.

Using a `substring` call in the constructor, remove the initial button information from the parameter. In our example, you should now have this:

```
Vasiliki,Center$Hojjat,West$Wei,East$Dong,South
```

Now try to extract the second button name and location, add it to the `BorderJFrame`, and *don't write any more code.* Instead, compile it, test it, and fix any errors. There should be two buttons in the window.

Repeat this process (switching roles frequently) until you have it working.

Show your TA.