LAB 11

The purpose of this lab is to create a simple application with a Graphical User Interface.

We will create a simple GUI around a Counter: the interface will have a button to increment the counter, and a little label showing the current value of the counter.

1. Create the counter and the GUI:

Create a class CounterFrame, with the following attributes:

- graphic elements: a JFrame with the title "Counter", a JLabel, and a JButton that says "increment counter"
- a Counter (Use the basic Counter provided)

For the JFrame, you can set the layout to a FlowLayout, which will arrange the objects inside in a line.

Compile and test. At this point your interface doesn't do anything, but everything should display nicely, so that you can concentrate on the interaction with the Counter.

2. Connect the button to the counter, using the Event Model. JButtons generate ActionEvents each time the user presses the button. So we need to create a listener that will listen for these events and actually increment the counter when they occur.

- First you need to define a class ButtonHandler that implements the interface ActionListener. The ButtonHandler must also have a reference to the Counter, and when the button is pressed, the ButtonHandler must increment the counter by calling its method increment()
- The CounterFrame must create a ButtonHandler and register it to the button

Compile and test. At this point pressing the button should increment the counter, which you can observe by inspecting the Counter object, but the value does not yet display in the GUI.

3. We now want to connect the counter to the label so that the label displays the counter value. We will do this using the Event Model, except this time the counter is the one that generates events, and we will create a handler class to listen for these events and modify the label.

3.1. Create the class CounterEvent, that describes a change of state of a Counter, an interface CounterListener that defines a method for listeners of such events.

3.2. Modify the Counter so that it maintains a list of CounterListeners and notifies them when it changes state (using a CounterEvent). It should also offer a method to add new listeners.

3.3. Create a class CounterHandler that implements the listener interface. The CounterFrame must create a CounterHandler, register it with the Counter. The CounterHandler must also have a reference to the GUI JLabel so that when the counter changes value, the new value is displayed in the JLabel.

Compile and test: now the GUI should be fully functional: when you press the button, the counter should increment, and the value should be visible in the JLabel.