David Katz CM2270
William D'Attilio CM2158
Frank Roetker CM2419

CSSE221-01
Dr. Defoe
September 20,2011

Capsule Project: Event Listeners

Source: Horstmann Section 9.6

What are Event Listeners?

Sometimes you want to run instructions after a certain event has happened. An event listener is an object that has a method that contains these instructions that you want to have executed when an event occurs. An event listener requires an event source. This event source is an object that can be linked to any number of event listeners.

Example:

```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
public class ClickLIstener implements ActionListener {
     public void actionPerformed(ActionEvent event){
          System.out.println("I was clicked.");
     }
}
```

What are some examples of when an event listener should be used?

1. ActionListener
    - This event is activated when a component's specific action occurs
2. ChangeListener
    - Checks to see when the state of an event has changed
3. FocusListener
    - Observes whether the input focus has increased or decreased in two different levels
4. ItemListener
    - Looks to see when an item is selected or deselected
5. KeyListener
    - Receives keyboard strokes
6. WindowListener
    - Processes window events such as opening and closing

Handling Mouse Events

The MouseListener interface has five methods: mouseClicked, mouseEntered, mouseExited, mousePressed, mouseReleased.

If you want to have an object handle mouse events:

- mouseClicked() checks for mouse clicks
- mouseEntered() checks if the mouse cursor enters a component
- mouseExited() checks if mouse cursor leaves a component
- mousePressed() checks if a mouse button has been pressed on a component
- mouseReleased() checks if a mouse button has been released on a component