Event Handlings

"Understanding Events, Handling events in Applets"
Advanced in Programming

Shakirullah Waseeb

shakir.waseeb@gmail.com

Nangarhar University

April 26, 2017

Agenda

1. Introduction
   - What is event handling?
   - Event Handling Approach

2. Event Handling in Practice
   - Events, Source, and Listener
   - Mouse click event example

3. Questions and Discussion

Agenda

Applets Overview

- As we studied Applets are event-driven programs in last lecture
- Hence, event handling is at the core of successful applet programming
- Most events to which applets responds are generated by the user
- Events can be passed into applets in a variety of ways, depending upon the actual event
- Most common handled events are generated by:
  - Mouse
  - Keyboard
  - Various controls such as button, text box, drop down, check box etc.
  - Run as a part of web document
- Events are supported by java.awt.event package

Agenda

Event Handling Approach

- Modern approach to event handling is based on **delegation event model**
- This approach defines standards and consistent mechanism to handle and process events
- In this model *source* generates an event and sends it to one or more *listeners*
  - The listener simply waits until it receives an event
  - Once received, the listener processes the event and then returns
  - Listeners must register with a source in order to receive an event notification

Agenda

Events

- In **delegation-model**, an event is an object that describes a *state change* in a source
- It can be generated as a result of users interacting with the elements in a graphical user interface
- Some of the activities that cause events to be generated are
  - pressing a button
  - entering a character via the keyboard
  - selecting an item in a list
  - clicking the mouse
- Events may also occur that are not directly caused by interactions with a user interface
  - an event may be generated when a timer expires
  - a counter exceeds a value
  - a software or hardware failure occurs
  - an operation is completed

Event Sources

- A **source** is an object that generates an event
- Occurs when the internal state of that object changes in some way
- May generate more than one type of event
- A source must register listeners in order for the listeners to receive notifications about a specific type of event
- Each type of event has its own registration method
  - general form
    public void *addTypeListener*(TypeListener el)
  - registering a keyboard listener
    public void *addKeyListener*(KeyEvent el)

Event Listeners

- A **listener** is an object that is notified when an event occurs
- Has two major requirements
    - must have been registered with one or more sources to receive notifications about specific types of events
    - must implement methods to receive and process these notifications
- The methods that receive and process events are defined in a set of interfaces found in java.awt.event
- MouseMotionListener interface defines two methods to receive notifications when the mouse is dragged or moved

Event Classes

- The classes that represent events are at the core of Java's event handling mechanism
- Some of these classes are listed below

Event Class	Description
ActionEvent	Generated when a button is pressed, a list item is double-clicked, or a menu item is selected.
AdjustmentEvent	Generated when a scroll bar is manipulated.
ComponentEvent	Generated when a component is hidden, moved, resized, or becomes visible.
ContainerEvent	Generated when a component is added to or removed from a container.
FocusEvent	Generated when a component gains or loses keyboard focus.
InputEvent	Abstract super class for all component input event classes.
ItemEvent	Generated when a check box or list item is clicked; also occurs when a choice selection is made or a checkable menu item is selected or deselected.

Figure: Some Event Classes [3, Page 657]

Agenda

A simple Applet Example

Simple Applet Handling a Mouse Click Event

```
import java.awt.*;
import java.applet.*;
/< /*applet code="SimpleMouseClickedApplet" width=200 height=60></applet> */
    public class SimpleMouseClickedApplet extends Applet implements MouseListener {
        public void init() {
            addMouseListener(this);
        }
        public void mouseClicked(MouseEvent me) {
            showStatus("Mouse Clicked");
            repaint();
        }
        public void paint(Graphics g) {
            g.drawString("A Simple Applet", 20, 20);
        }
    }
```

Your Turn: Time to hear from you!

1

[1]https://fensafitters.files.wordpress.com/2013/07/3d095.jpg

References

📕 P.J. Deitel, H.M. Deitel
Java How to program, 10th Edition .
Prentice Hall, 2015.

📕 P.J. Deitel, H.M. Deitel
Java How to program, 9th Edition .
Prentice Hall, 2012.

📕 Herbert Schildt
The complete reference Java2, 5th Edition .
McGraw-Hill/Osborne, 2002.