

JavaSwing中三种事件处理方法之比较 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022_JavaSwing_E4_c104_145006.htm

Swing是目前Java中不可缺少的窗口工具组，是用户建立图形化用户界面（GUI）程序的强大工具。Java Swing组件自动产生各种事件来响应用户行为。如当用户点击按钮或选择菜单项目时，Swing组件会产生一个ActionEvent。Swing组件会产生许多事件，如ActionEvents,ChangeEvents,ItemEvents等，来响应用户的鼠标点击行为，列表框中值的改变，计时器的开始计时等行为。在Java Swing编程中，通过注册监听器，我们可以监听事件源产生的事件，从而在事件处理程序中处理我们所需要处理的用户行为。Java Swing中处理各组件事件的一般步骤是：1. 新建一个组件（如JButton）。2. 将该组件添加到相应的面板（如JPanel）。3. 注册监听器以监听事件源产生的事件（如通过ActionListener来响应用户点击按钮）。4. 定义处理事件的方法（如在ActionListener中的actionPerformed中定义相应方法）。以上步骤我们可以用多种方法实现。但人们通常用二种方法。第一种方法是只利用一个监听器以及多个if语句来决定是哪个组件产生的事件；第二种方法是使用多个内部类来响应不同组件产生的各种事件，其具体实现又分两种方式，一种是匿名内部类，一种是一般内部类。为了说明如何使用上述三种方法实现事件的处理方法，我们建立一个简单的应用程序。该程序界面有两个按钮，当用户点击相应的按钮，就会弹出一个对话框显示相应的内容。通过这个简单程序，你可以实现自己更多、更复杂的用户界面程序。首先，

我们利用单个监听器来实现该程序。我们定义一个名为Simple1的类来包括所有代码。所有的用户行为（如点击按钮）由一个监听器SimpleListener中的actionPerformed方法来处理。以下是代码：

```
/* * Simple1.java - 处理事件的第一种方法
* 在这个例子中，利用一个ActionListener来监听事件源产生的事件 * 用一些if语句来决定是哪个事件源 */ import java.awt.*.
import java.awt.event.*. import javax.swing.*. public class Simple1 {
private static JFrame frame. // 定义为静态变量以便main使用
private static JPanel myPanel. // 该面板用来放置按钮组件 private
JButton button1. // 这里定义按钮组件 private JButton button2. //
以便让ActionListener使用 public Simple1() // 构造器, 建立图形
界面 { // 新建面板 myPanel = new JPanel(). // 新建按钮 button1
= new JButton("按钮1"). // 新建按钮1 button2 = new JButton("按
钮2"). SimpleListener ourListener = new SimpleListener(). // 建立
一个actionlistener让两个按钮共享
button1.addActionListener(ourListener).
button2.addActionListener(ourListener). myPanel.add(button1). //
添加按钮到面板 myPanel.add(button2). } private class
SimpleListener implements ActionListener { /* * 利用该内部类来
监听所有事件源产生的事件 * 便于处理事件代码模块化 */
public void actionPerformed(ActionEvent e) { // 利
用getActionCommand获得按钮名称 // 也可以利用getSource()
来实现 // if (e.getSource() ==button1) String buttonName =
e.getActionCommand(). if (buttonName.equals("按钮1")) 100Test
下载频道开通，各类考试题目直接下载。详细请访问
www.100test.com
```