

SWING - COMPONENTLISTENER INTERFACE

http://www.tutorialspoint.com/swing/swing_component_listener.htm

Copyright © tutorialspoint.com

The class which processes the `ComponentEvent` should implement this interface. The object of that class must be registered with a component. The object can be registered using the `addComponentListener` method. Component event are raised for information only.

Interface declaration

Following is the declaration for **java.awt.event.ComponentListener** interface:

```
public interface ComponentListener
    extends EventListener
```

Interface methods

S.N.	Method & Description
1	void componentHidden <i>ComponentEvent</i> Invoked when the component has been made invisible.
2	void componentMoved <i>ComponentEvent</i> Invoked when the component's position changes.
3	void componentResized <i>ComponentEvent</i> Invoked when the component's size changes.
4	void componentShown <i>ComponentEvent</i> Invoked when the component has been made visible.

Methods inherited

This interface inherits methods from the following interfaces:

- `java.awt.EventListener`

ComponentListener Example

Create the following java program using any editor of your choice in say **D:/ > SWING > com > tutorialspoint > gui >**

SwingListenerDemo.java

```
package com.tutorialspoint.gui;

import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class SwingListenerDemo {
    private JFrame mainFrame;
    private JLabel headerLabel;
    private JLabel statusLabel;
    private JPanel controlPanel;
```

```

public SwingListenerDemo(){
    prepareGUI();
}

public static void main(String[] args){
    SwingListenerDemo swingListenerDemo = new SwingListenerDemo();
    swingListenerDemo.showComponentListenerDemo();
}

private void prepareGUI(){
    mainFrame = new JFrame("Java SWING Examples");
    mainFrame.setSize(400,400);
    mainFrame.setLayout(new GridLayout(3, 1));

    headerLabel = new JLabel("",JLabel.CENTER );
    statusLabel = new JLabel("",JLabel.CENTER);

    statusLabel.setSize(350,100);
    mainFrame.addWindowListener(new WindowAdapter() {
        public void windowClosing(WindowEvent windowEvent){
            System.exit(0);
        }
    });
    controlPanel = new JPanel();
    controlPanel.setLayout(new FlowLayout());

    mainFrame.add(headerLabel);
    mainFrame.add(controlPanel);
    mainFrame.add(statusLabel);
    mainFrame.setVisible(true);
}

private void showComponentListenerDemo(){
    headerLabel.setText("Listener in action: ComponentListener");

    JPanel panel = new JPanel();
    panel.setBackground(Color.magenta);

    JLabel msglabel =
    new JLabel("Welcome to Tutorialspoint SWING Tutorial."
        ,JLabel.CENTER);
    panel.add(msglabel);

    msglabel.addComponentListener(new CustomComponentListener());
    controlPanel.add(panel);
    mainFrame.setVisible(true);
}

class CustomComponentListener implements ComponentListener {

    public void componentResized(ComponentEvent e) {
        statusLabel.setText(statusLabel.getText()
            + e.getComponent().getClass().getSimpleName() + " resized. ");
    }

    public void componentMoved(ComponentEvent e) {
        statusLabel.setText(statusLabel.getText()
            + e.getComponent().getClass().getSimpleName() + " moved. ");
    }

    public void componentShown(ComponentEvent e) {
        statusLabel.setText(statusLabel.getText()
            + e.getComponent().getClass().getSimpleName() + " shown. ");
    }

    public void componentHidden(ComponentEvent e) {
        statusLabel.setText(statusLabel.getText()
            + e.getComponent().getClass().getSimpleName() + " hidden. ");
    }
}

```

```
}  
}
```

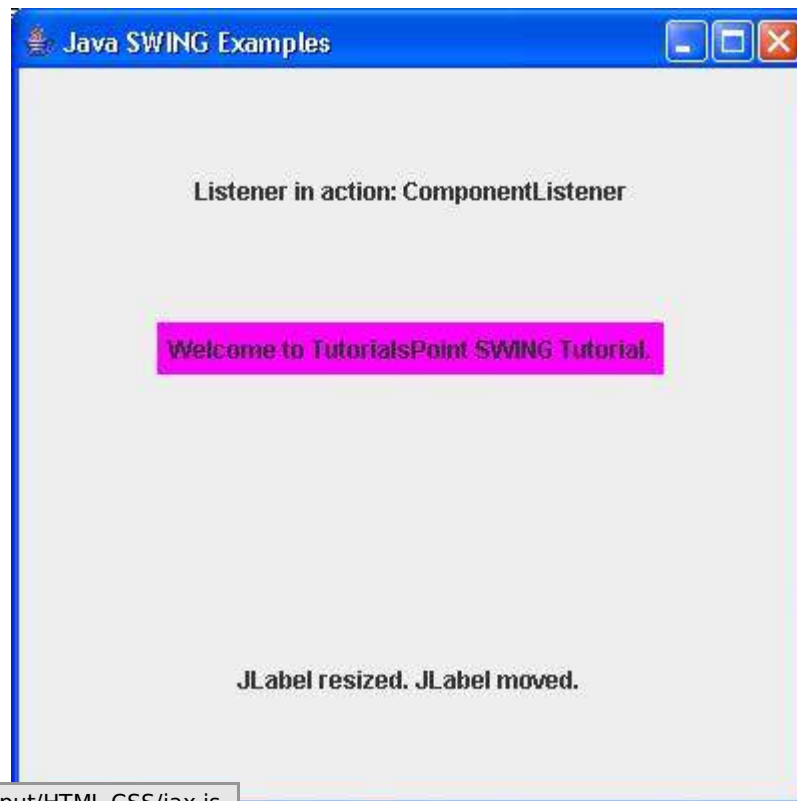
Compile the program using command prompt. Go to **D:/ > SWING** and type the following command.

```
D:\SWING>javac com\tutorialspoint\gui\SwingListenerDemo.java
```

If no error comes that means compilation is successful. Run the program using following command.

```
D:\SWING>java com.tutorialspoint.gui.SwingListenerDemo
```

Verify the following output



Loading [MathJax]/jax/output/HTML-CSS/jax.js