

SWING - CONTAINERLISTENER INTERFACE

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

Copyright © tutorialspoint.com

Introduction

The interface **ContainerListener** is used for receiving container events. The class that process container events needs to implements this interface.

Class declaration

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

```
public interface ContainerListener
extends EventListener
```

Interface methods

S.N.	Method & Description
1	void componentAddedContainerEvente Invoked when a component has been added to the container.
2	void componentRemovedContainerEvente Invoked when a component has been removed from the container.

Methods inherited

This class inherits methods from the following interfaces:

- java.awt.event.EventListener

ContainerListener 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.showContainerListenerDemo();
    }
}
```

```

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 showContainerListenerDemo(){
    headerLabel.setText("Listener in action: ContainerListener");

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

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

    controlPanel.add(panel);
    mainFrame.setVisible(true);
}

class CustomContainerListener implements ContainerListener {
    public void componentAdded(ContainerEvent e) {
        statusLabel.setText(statusLabel.getText()
        + e.getComponent().getClass().getSimpleName() + " added. ");
    }

    public void componentRemoved(ContainerEvent e) {
        statusLabel.setText(statusLabel.getText()
        + e.getComponent().getClass().getSimpleName() + " removed. ");
    }
}
}

```

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



Listener in action: ContainerListener

Welcome to TutorialsPoint SWING Tutorial.

JPanel added.

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