

SWING - JBUTTON CLASS

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

Copyright © tutorialspoint.com

Introduction

The class **JButton** is an implementation of a push button. This component has a label and generates an event when pressed. It can have Image also.

Class declaration

Following is the declaration for **javax.swing.JButton** class –

```
public class JButton
    extends AbstractButton
    implements Accessible
```

Class constructors

S.N.	Constructor & Description
1	JButton Creates a button with no set text or icon.
2	JButtonActiona Creates a button where properties are taken from the Action supplied.
3	JButtonIconicon Creates a button with an icon.
4	JButtonStringtext Creates a button with text.
5	JButtonStringtext, Iconicon Creates a button with initial text and an icon.

Class methods

S.N.	Method & Description
1	AccessibleContext getAccessibleContext Gets the AccessibleContext associated with this JButton.
2	String getUIClassID Returns a string that specifies the name of the L&F class that renders this component.
3	boolean isDefaultButton

Gets the value of the `defaultButton` property, which if true means that this button is the current default button for its `JRootPane`.

4 **boolean isDefaultCapable**

Gets the value of the `defaultCapable` property.

5 **protected String paramString**

Returns a string representation of this `JButton`.

6 **void removeNotify**

Overrides `JComponent.removeNotify` to check if this button is currently set as the default button on the `RootPane`, and if so, sets the `RootPane`'s default button to null to ensure the `RootPane` doesn't hold onto an invalid button reference.

7 **void setDefaultCapablebooleandefaultCapable**

Sets the `defaultCapable` property, which determines whether this button can be made the default button for its root pane.

8 **void updateUI**

Resets the UI property to a value from the current look and feel.

Methods inherited

This class inherits methods from the following classes:

- `javax.swing.AbstractButton`
- `javax.swing.JComponent`
- `java.awt.Container`
- `java.awt.Component`
- `java.lang.Object`

JButton Example

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

SwingControlDemo.java

```
package com.tutorialspoint.gui;

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

public class SwingControlDemo {

    private JFrame mainFrame;
    private JLabel headerLabel;
    private JLabel statusLabel;
    private JPanel controlPanel;

    public SwingControlDemo(){
        prepareGUI();
    }
```

```

}

public static void main(String[] args){
    SwingControlDemo swingControlDemo = new SwingControlDemo();
    swingControlDemo.showButtonDemo();
}

private void prepareGUI(){
    mainFrame = new JFrame("Java Swing Examples");
    mainFrame.setSize(400,400);
    mainFrame.setLayout(new GridLayout(3, 1));
    mainFrame.addWindowListener(new WindowAdapter() {
        public void windowClosing(WindowEvent windowEvent){
            System.exit(0);
        }
    });
    headerLabel = new JLabel("", JLabel.CENTER);
    statusLabel = new JLabel("",JLabel.CENTER);

    statusLabel.setSize(350,100);

    controlPanel = new JPanel();
    controlPanel.setLayout(new FlowLayout());

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

private static ImageIcon createImageIcon(String path,
    String description) {
    java.net.URL imgURL = SwingControlDemo.class.getResource(path);
    if (imgURL != null) {
        return new ImageIcon(imgURL, description);
    } else {
        System.err.println("Couldn't find file: " + path);
        return null;
    }
}

private void showButtonDemo(){

    headerLabel.setText("Control in action: Button");

    //resources folder should be inside SWING folder.
    ImageIcon icon = createImageIcon("/resources/java_icon.png", "Java");

    JButton okButton = new JButton("OK");
    JButton javaButton = new JButton("Submit", icon);
    JButton cancelButton = new JButton("Cancel", icon);
    cancelButton.setHorizontalTextPosition(SwingConstants.LEFT);

    okButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            statusLabel.setText("Ok Button clicked.");
        }
    });

    javaButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            statusLabel.setText("Submit Button clicked.");
        }
    });

    cancelButton.addActionListener(new ActionListener() {
        public void actionPerformed(ActionEvent e) {
            statusLabel.setText("Cancel Button clicked.");
        }
    });
}

```

```
});  
  
controlPanel.add(okButton);  
controlPanel.add(javaButton);  
controlPanel.add(cancelButton);  
  
mainFrame.setVisible(true);  
}  
}
```

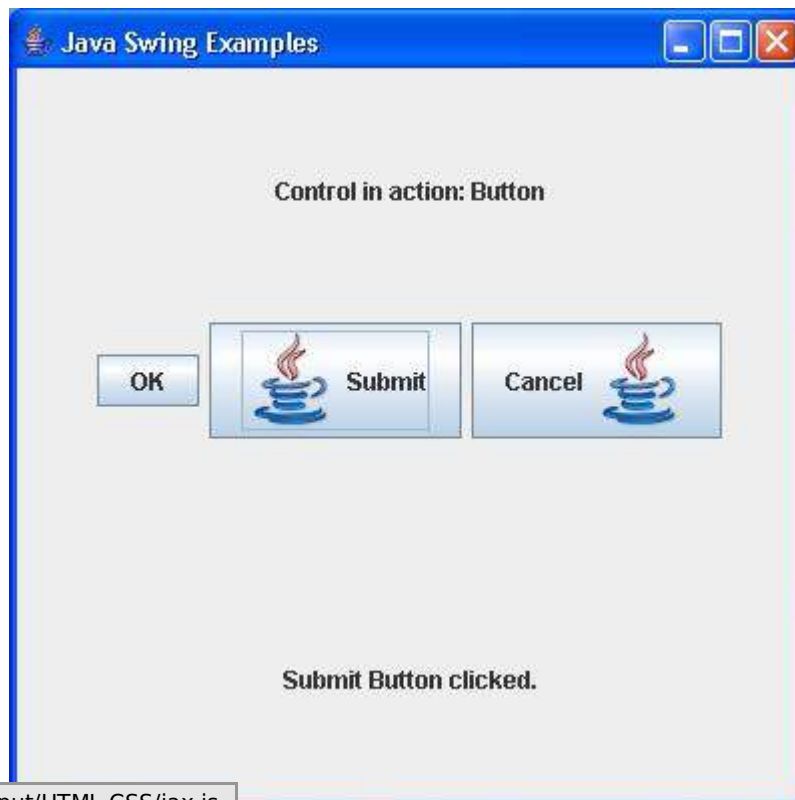
Compile the program using command prompt. Go to **D:/ > SWING** and type the following command.

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

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

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

Verify the following output



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