

SWING - JRadioButtonMenuItem CLASS

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

Copyright © tutorialspoint.com

Introduction

The JRadioButtonMenuItem class represents a check box which can be included in a menu. Selecting the check box in the menu changes control's state from **on** to **off** or from **off** to **on**.

Class declaration

Following is the declaration for **javax.swing.JRadioButtonMenuItem** class:

```
public class JRadioButtonMenuItem
    extends JMenuItem
    implements Accessible
```

Class constructors

S.N.	Constructor & Description
1	JRadioButtonMenuItem Creates a JRadioButtonMenuItem with no set text or icon.
2	JRadioButtonMenuItemActiona Creates a radio button menu item whose properties are taken from the Action supplied.
3	JRadioButtonMenuItemIconicon Creates a JRadioButtonMenuItem with an icon.
4	JRadioButtonMenuItemIconicon, booleanselected Creates a radio button menu item with the specified image and selection state, but no text.
5	JRadioButtonMenuItemStringtext Creates a JRadioButtonMenuItem with text.
6	JRadioButtonMenuItemStringtext, booleanselected Creates a radio button menu item with the specified text and selection state.
7	JRadioButtonMenuItemStringtext, Iconicon Creates a radio button menu item with the specified text and Icon.
8	JRadioButtonMenuItemStringtext, Iconicon, booleanselected Creates a radio button menu item that has the specified text, image, and selection state.

Class methods

S.N.	Method & Description
1	AccessibleContext getAccessibleContext Gets the AccessibleContext associated with this JRadioButtonMenuItem.
2	String getUIClassID Returns the name of the L&F class that renders this component.
3	protected String paramString Returns a string representation of this JRadioButtonMenuItem.

Methods inherited

This class inherits methods from the following classes:

- javax.swing.JMenuItem
- javax.swing.JAbstractButton
- javax.swing.JComponent
- java.awt.Container
- java.awt.Component
- java.lang.Object

JRadioButtonMenuItem Example

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

SwingMenuDemo.java

```
package com.tutorialspoint.gui;

import java.awt.*;
import java.awt.event.*;

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

    public SwingMenuDemo(){
        prepareGUI();
    }

    public static void main(String[] args){
        SwingMenuDemo swingMenuDemo = new SwingMenuDemo();
        swingMenuDemo.showMenuDemo();
    }

    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 showMenuDemo(){
    //create a menu bar
    final JMenuBar menuBar = new JMenuBar();

    //create menus
    JMenu fileMenu = new JMenu("File");
    JMenu editMenu = new JMenu("Edit");
    final JMenu aboutMenu = new JMenu("About");
    final JMenu linkMenu = new JMenu("Links");

    //create menu items
    JMenuItem newItem = new JMenuItem("New");
    newItem.setMnemonic(KeyEvent.VK_N);
    newItem.setActionCommand("New");

    JMenuItem openMenuItem = new JMenuItem("Open");
    openMenuItem.setActionCommand("Open");

    JMenuItem saveMenuItem = new JMenuItem("Save");
    saveMenuItem.setActionCommand("Save");

    JMenuItem exitMenuItem = new JMenuItem("Exit");
    exitMenuItem.setActionCommand("Exit");

    JMenuItem cutMenuItem = new JMenuItem("Cut");
    cutMenuItem.setActionCommand("Cut");

    JMenuItem copyMenuItem = new JMenuItem("Copy");
    copyMenuItem.setActionCommand("Copy");

    JMenuItem pasteMenuItem = new JMenuItem("Paste");
    pasteMenuItem.setActionCommand("Paste");

    MenuItemListener menuItemListener = new MenuItemListener();

    newItem.addActionListener(menuItemListener);
    openMenuItem.addActionListener(menuItemListener);
    saveMenuItem.addActionListener(menuItemListener);
    exitMenuItem.addActionListener(menuItemListener);
    cutMenuItem.addActionListener(menuItemListener);
    copyMenuItem.addActionListener(menuItemListener);
    pasteMenuItem.addActionListener(menuItemListener);

    final JCheckBoxMenuItem showWindowMenu = new JCheckBoxMenuItem("Show About", true);
    showWindowMenu.addItemListener(new ItemListener() {
        public void itemStateChanged(ItemEvent e) {
            if(showWindowMenu.getState()){
                menuBar.add(aboutMenu);
            }else{
                menuBar.remove(aboutMenu);
            }
        }
    });
}

```

```

final JRadioButtonMenuItem showLinksMenu =
    new JRadioButtonMenuItem("Show Links", true);
showLinksMenu.addItemListener(new ItemListener() {
    public void itemStateChanged(ItemEvent e) {
        if(menuBar.getMenu(3) != null){
            menuBar.remove(linkMenu);
            mainFrame.repaint();
        }else{
            menuBar.add(linkMenu);
            mainFrame.repaint();
        }
    }
});

//add menu items to menus
fileMenu.add(newMenuItem);
fileMenu.add(openMenuItem);
fileMenu.add(saveMenuItem);
fileMenu.addSeparator();
fileMenu.add(showWindowMenu);
fileMenu.addSeparator();
fileMenu.add(showLinksMenu);
fileMenu.addSeparator();
fileMenu.add(exitMenuItem);
editMenu.add(cutMenuItem);
editMenu.add(copyMenuItem);
editMenu.add(pasteMenuItem);

//add menu to menubar
menuBar.add(fileMenu);
menuBar.add(editMenu);
menuBar.add(aboutMenu);
menuBar.add(linkMenu);

//add menubar to the frame
mainFrame.setJMenuBar(menuBar);
mainFrame.setVisible(true);
}

class MenuItemListener implements ActionListener {
    public void actionPerformed(ActionEvent e) {
        statusLabel.setText(e.getActionCommand()
            + " JMenuItem clicked.");
    }
}
}

```

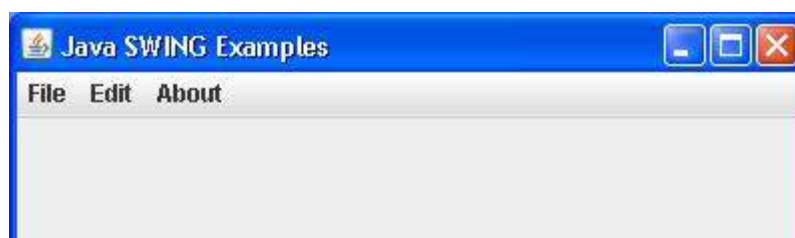
Compile the program using command prompt. Go to **D:/ > SWING** and type the following command.

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

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

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

Verify the following output. *Click on File Menu. Unselect " ShowLink " menuitem.*



Processing math: 100%

