



Componentes GUI e Eventos (Parte 2)

Richarlyson A. D'Emery

site: <https://sites.google.com/site/profricodemery/mpoo>

grupo: http://groups.google.com/group/mpoo_uast

email grupo: mpoo_uast@googlegroups.com

contato: rico_demery@yahoo.com.br

Sumário



- ItemListener e ListSelectionListener
- JTextField
- JCheckBox e JRadioButton
- JToggleButton

Outros Listeners

- Interface **ItemListener**

- método manipulador:

```
public void itemStateChanged( ItemEvent event ) {...}
```

- GUI: JCheckBox, JRadioButton, JToggleButton

- Interface **ListSelectionListener**

- método manipulador:

```
public void valueChanged( ListSelectionEvent event ) {...}
```

- GUI: JList

JCheckBox e JRadioButton

- Os componentes GUI Swing contêm *botões de estado*
 - JCheckBox
 - JRadioButton
 - JToggleButton
- Seus estados
 - Ativados / desativados
 - Verdadeiro / falso

JCheckBox e JRadioButton

- São subclasses de JToggleButton
- O **JRadioButton** é diferente do **JCheckBox** porque:
 - Elementos de JRadioButton devem estar agrupados;
 - Dos vários JRadioButton de um mesmo grupo apenas um pode estar acionados.
 - Os JCheckBox são independentes

Eventos de JCheckBox

- Quando o usuário clica em um JCheckBox é gerado um **ItemEvent**
 - O tratamento é feito por um **ItemListener**
- O ItemListener deve definir
 - **itemStateChanged**

Exemplo JCheckBox

```
import java.awt.FlowLayout;
import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import javax.swing.JCheckBox;
import javax.swing.JFrame;
import javax.swing.JOptionPane;

public class ExemploCheckBoxcomEvento extends JFrame implements ItemListener{
    JCheckBox cb1;
    JCheckBox cb2;

    public ExemploCheckBoxcomEvento(){
        super("CheckBox");
        setLayout (new FlowLayout());
        cb1 = new JCheckBox("opcao1", false);
        cb2 = new JCheckBox("opcao2", false);
        cb1.addItemListener(this);
        cb2.addItemListener(this);
        add(cb1); add(cb2);
        setSize(100,100);
        setVisible(true);
        setLocationRelativeTo(null);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    public void itemStateChanged(ItemEvent event){
        if ( event.getSource() == cb1)
            JOptionPane.showMessageDialog(null, cb1.getText() + " selecionado");
        if ( event.getSource() == cb2)
            JOptionPane.showMessageDialog(null, cb2.getText() + " selecionado");
    }

    public static void main(String [] args){
        new ExemploCheckBoxcomEvento();
    }
}
```

Exemplo JRadioButton e JTextField

```

import java.awt.event.ItemEvent;
import java.awt.event.ItemListener;
import java.awt.FlowLayout;
import javax.swing.JFrame;
import javax.swing.JTextField;
import javax.swing.JRadioButton;
import javax.swing.ButtonGroup;

public class ExemploRadioButtoncomEvento extends
    JFrame{

    JRadioButton op1, op2;
    JTextField field;
    ButtonGroup grupo;
    String texto="opcao1 selecionada";

    public ExemploRadioButtoncomEvento(){
        super("CheckBox");
        setLayout (new FlowLayout());

        field = new JTextField();
        field.setEditable(false);
        field.setText(texto);

        op1 = new JRadioButton("opcao1", true);
        op2 = new JRadioButton("opcao2", false);

        RadioButtonHandler handler = new
        RadioButtonHandler();
        op1.addItemListener(handler);
        op2.addItemListener(handler);

        grupo = new ButtonGroup();
        grupo.add(op1);
        grupo.add(op2);
  
```

```

        add(field); add(op1); add(op2);

        setSize(200,100);
        setVisible(true);
        setLocationRelativeTo(null);
        setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    }

    private class RadioButtonHandler implements ItemListener
    {
        public void itemStateChanged(ItemEvent event){
            if (event.getSource() == op1){
                texto=(op1.getText() + " selecionada");
                field.setText(texto);
            }

            if (event.getSource() == op2){
                texto=(op2.getText() + " selecionada");
                field.setText(texto);
            }
        }
    }

    public static void main(String [] args){
        new ExemploRadioButtoncomEvento();
    }
}
  
```


JToggleButton



- Utilizados com *barras de ferramentas*
 - Conjunto de pequenos botões em geral localizados em uma barra ao longo da parte superior de uma janela

JToggleButton



```
import javax.swing.*;
import java.awt.GridLayout;
import java.awt.event.ItemListener;
import java.awt.event.ItemEvent;

public class ExemploJToggleButton extends JFrame
    implements ItemListener {
    public ExemploJToggleButton() {
        super("Exemplo de JToggleButton");
        setLayout(new GridLayout(6,1));

        JToggleButton tg1 = new JToggleButton("Opção
1");
        JToggleButton tg2 = new JToggleButton("Opção
2");
        JToggleButton tg3 = new JToggleButton("Opção
3");
        JToggleButton tg4 = new JToggleButton("Opção
4");

        add(new JLabel("Selecione as opções
desejadas:"));
        add(new JLabel(""));

        add(tg1); add(tg2); add(tg3); add(tg4);

        tg1.addItemListener(this);
        tg2.addItemListener(this);
        tg3.addItemListener(this);
        tg4.addItemListener(this);

        setVisible(true);
    }
}
```

```
public void itemStateChanged(ItemEvent evt) {
    JToggleButton tg =
        (JToggleButton)evt.getItemSelectable();
    if (evt.getStateChange() ==
        ItemEvent.SELECTED)
        System.out.println(tg.getText() + "
        marcado.");
    else if (evt.getStateChange() ==
        ItemEvent.DESELECTED)
        System.out.println(tg.getText() + "
        desmarcado.");
}

public static void main(String[] args) {
    new ExemploJToggleButton();
}
```

Exercício

- Crie a seguinte GUI:
 - Deve ter 1 checkbox
 - Mostrar gráfico (carrega uma imagem de um gráfico)
 - Deve ter dois labels (X, Y) para duas textAreas (valores de X e Y)
 - Quatro botões: Fechar, Ajuda, Incrementar X e Incrementar Y. Onde:
 - Fechar – encerra a aplicação
 - Ajuda – abre uma caixa de mensagem
 - Incrementar X – aumenta em uma unidade o valor de X e atualiza o valor do textArea
 - Incrementar Y – aumenta em uma unidade o valor de Y e atualiza o valor do textArea



FIM

Prof. Richarlyson D'Emery

site: <https://sites.google.com/site/profricodemery/mpoo>

grupo: http://groups.google.com/group/mpoo_uast

email grupo: mpoo_uast@googlegroups.com

contato: rico_demery@yahoo.com.br