
Eventos em Aplicações SWING

Werther

Eventos

- Qualquer interação do usuário com a tela e seus objetos
- Gerenciadores de eventos → **Listeners** (ouvintes)
 - Implementados através de Interfaces
 - Interface: classe especial que define um conjunto de métodos a serem implementados
- Tipos de eventos mais comuns:
 - Ação → implementado por **ActionListener**
 - Janela → implementado por **WindowListener**
 - Teclado → implementado por **KeyListener**
 - Mouse → implementado por **MouseListener**

Existe mais de 40 Listeners, entre eles podemos destacar:

ActionListener, AdjustmentListener, ComponentListener, ContainerListener, FocusListener, ItemListener, KeyListener, MouseListener, MouseMotionListener, TextListener, WindowListener

Eventos Aplicações SWING

Eventos: Principais Componentes x *Listeners*

Component	<u>Action Listener</u>	<u>Change Listener</u>	<u>Focus Listener</u>	<u>Item Listener</u>	<u>Key Listener</u>	<u>List Selection Listener</u>	<u>Mouse Listener</u>
<u>button</u>	✓	✓	✓	✓	✓		✓
<u>check box</u>	✓	✓	✓	✓			✓
<u>combo box</u>	✓		✓	✓	✓		✓
<u>dialog</u>							✓
<u>frame</u>							✓
<u>list</u>			✓			✓	✓
<u>menu</u>							✓
<u>menu item</u>	✓	✓		✓	✓		✓
<u>option pane</u>							✓
<u>password field</u>	✓		✓		✓		✓
<u>popup menu</u>			✓				
<u>progress bar</u>		✓					✓
<u>radio button</u>	✓	✓	✓				✓
<u>spinner</u>		✓	✓		✓		✓
<u>tabbed pane</u>		✓	✓				✓
<u>table</u>						✓	✓
<u>text area</u>			✓		✓		✓
<u>text field</u>	✓		✓		✓		✓
<u>toggle button</u>	✓	✓	✓	✓			✓

Eventos Aplicações SWING

Exemplo: Evento de KeyListener

- Implementando método para verificação entrada de números teclado em um **JTextField**
- Necessário que a classe implemente **KeyListener**

```
import java.awt.*; import java.awt.event.*;
public class Exemplo1 extends JFrame implements KeyListener {
    ...
    public JTextField tx;
    ...
    public Exemplo1() {
        ...
        tx.addKeyListener(this);
        ...
    }

    public void keyTyped(KeyEvent e) {
        int key = e.getKeyChar();
        if(key<KeyEvent.VK_0 || key>KeyEvent.VK_9)
            e.consume();
    }
    public void keyPressed(KeyEvent e) { }
    public void keyReleased(KeyEvent e) { }
    ...
}
```

A classe tem que implementar a interface **KeyListener**

No construtor, o **Listener** é adicionado

Todos os 3 métodos da interface **KeyListener** têm que ser implementados, senão a classe fica abstrata

Efetivamente nesse exemplo apenas um método é implementado

Eventos Aplicações SWING

Exemplo: Evento de ActionListener

Utilizando interface
ActionListener

Implementando único método da
interface **ActionListener**, o
actionPerformed

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

public class Exemplo2 implements ActionListener
{

    public Exemplo2() {
        ...
        b1.addActionListener(this);
        b2.addActionListener(this);

        b1.setMnemonic(KeyEvent.VK_S);
        b2.setMnemonic(KeyEvent.VK_C);
    }
}
```

```
public void actionPerformed(ActionEvent e) {
    if(e.getSource()==b1)
        ...<code>...;
    else if(e.getSource()==b2)
        ...<code>...;
}
```

Adicionando ouvinte de
ação (*clique*) aos botões

Classe com código **VK_??**
para cada tecla ou
combinação delas

Registrando **Alt-S** e **Alt-C** como ouvinte
de ação (*clique*) de **b1** e **b2**
respectivamente