

JAVA

LINGUAGEM DE PROGRAMAÇÃO
INTERFACE GRÁFICA (GUI)

Eventos

LINGUAGEM DE PROGRAMAÇÃO JAVA
INTERFACE GRÁFICA (GUI)

Eventos

As aplicações em geral precisam de alguns eventos para o funcionamento completo do programa.

Normalmente as GUIs do Java são baseadas em eventos, que são tarefas realizadas quando um usuário faz a interação com algum componente GUI, convertendo as ações do usuário em eventos.

Eventos

A interface gráfica em Java é dita orientada a eventos:

- Cada vez que um usuário clica em um botão, seleciona um item em uma lista, ou pressiona uma tecla, o sistema operacional gera um evento;
- Se uma aplicação está interessada em um evento específico (por exemplo, clique em um botão), deve solicitar ao sistema para “escutar” o evento;

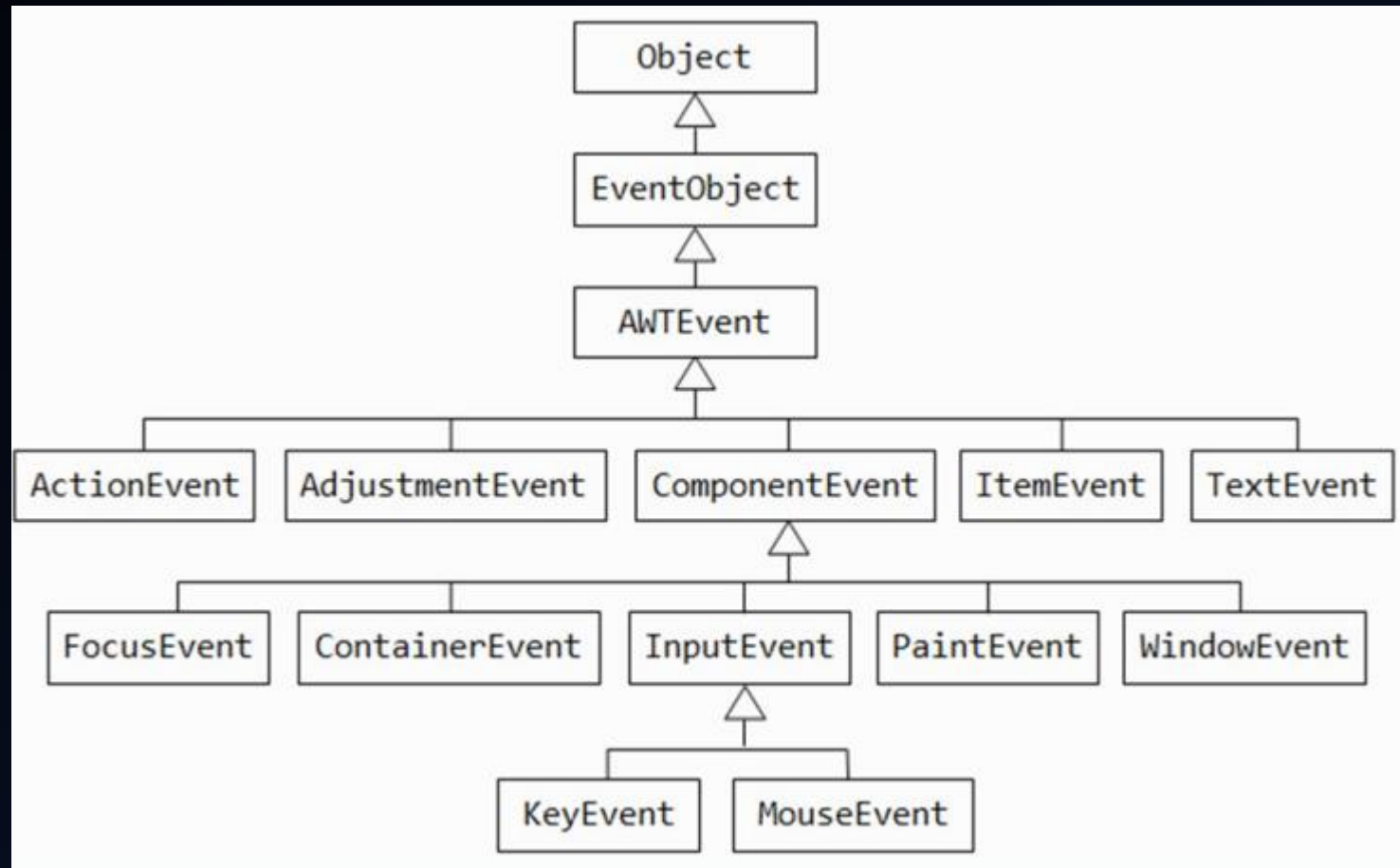
É importante observar que a aplicação não espera pela ocorrência de eventos: isso é controlado pelo sistema

Eventos

Existem três elementos para o funcionamento do tratamento de evento que são:

- **Origem do Evento:** é o componente GUI com qual o usuário interage.
- **Objeto do Evento:** contém os dados do evento invocado, sendo como uma referência à origem do evento e quaisquer informações específicas do evento que podem ser exigidas pelo ouvinte para testar o evento.
- **Ouvinte do Evento:** é um objeto que é notificado pela origem de evento quando um evento ocorre.

Eventos



Eventos

Listeners são classes criadas especificamente para o tratamento de eventos

A classe ActionListener é uma interface, ou seja, é classe composta apenas de métodos abstratos.

Eventos

Ocorre uma ação

Quando ocorre uma ação

Clique do mouse

Apertar em uma tecla

Fechar uma janela

Nome Evento

(ActionEvent);

(MouseEvent);

(KeyEvent);

(WindowEvent);

Eventos

AÇÃO DO USUÁRIO	Evento disparado	LISTENER
Clicar em um botão - JButton	ActionEvent	ActionListener
Abrir, minimizar ou fechar uma janela - JFrame	WindowEvent	WindowListener
Clicar em um componente - JComponent	MouseEvent	MouseListener
Alteração de texto em um campo de texto - JTextField	TextEvent	TextListener
Digitar uma tecla	KeyEvent	KeyListener
Clicar em um item de escolha - Checkbox	ItemEvent	ItemListener

Eventos

Interface WindowListener

Métodos	Função
<code>windowActivated(WindowEvent e)</code>	Invocada quando a janela recebe o foco
<code>windowClosed(WindowEvent e)</code>	Invocado quando a janela foi fechada
<code>windowClosing(WindowEvent e)</code>	Invocado quando a janela está fechando
<code>windowDeactivated(WindowEvent e)</code>	Invocado quando a janela perde o foco
<code>windowDeiconified(WindowEvent e)</code>	Invocado quando a janela é maximizada
<code>windowIconified(WindowEvent e)</code>	Invocado quando a janela é minimizada

Eventos

Interface `MouseListener`

Métodos	Função
<code>mouseClicked(MouseEvent e)</code>	Invocada quando o botão é pressionado e solto sobre o componente
<code>mouseEntered(MouseEvent e)</code>	Invocado quando o mouse entra na região gráfica do componente
<code>mouseExited(MouseEvent e)</code>	Invocado quando o mouse sai da região gráfica do componente
<code>mousePressed(MouseEvent e)</code>	Invocado quando o botão do mouse é apertado
<code>mouseReleased(MouseEvent e)</code>	Invocado quando o botão do mouse é solto
<code>mouseDragged(MouseEvent e)</code>	Invocado quando o mouse se movimenta e o seu botão principal está apertado

The background is a dark navy blue. On the left side, there are several parallel teal lines that form a corner-like shape. On the bottom right, there are several parallel teal lines that form a diagonal shape.

Obrigado!

UC - JAVA