

Interface gráfica de usuário

Introdução

- Uma **interface gráfica de usuário** (GUI – *Graphics User Interface*) permite ao usuário interagir com a aplicação utilizando ícones e desenhos, ao contrário de interfaces baseadas apenas em texto
 - Fornecem apresentação mais amigável ao usuário.
- GUIs podem ser construídas reutilizando componentes de bibliotecas gráficas.
 - Os componentes também são conhecidos por “controles GUI” ou simplesmente “controles”;
- As principais bibliotecas nativas do Java para construção de GUIs são:
 - AWT
 - Swing
 - JavaFX

AWT



AWT – Design e comportamentos distintos entre os sistemas operacionais

Netbeans GUI Builder

- Ferramenta do Netbeans que permite arrastar e posicionar *componentes GUI* de uma **paleta de componentes** para uma **área de desenho**
- Cada componente GUI inserido na área de desenho é um objeto Java. A classe do objeto depende do tipo de componente utilizado
- O GUI Builder procura alinhar automaticamente os componentes inseridos na área de desenho

Acionando o GUI Builder

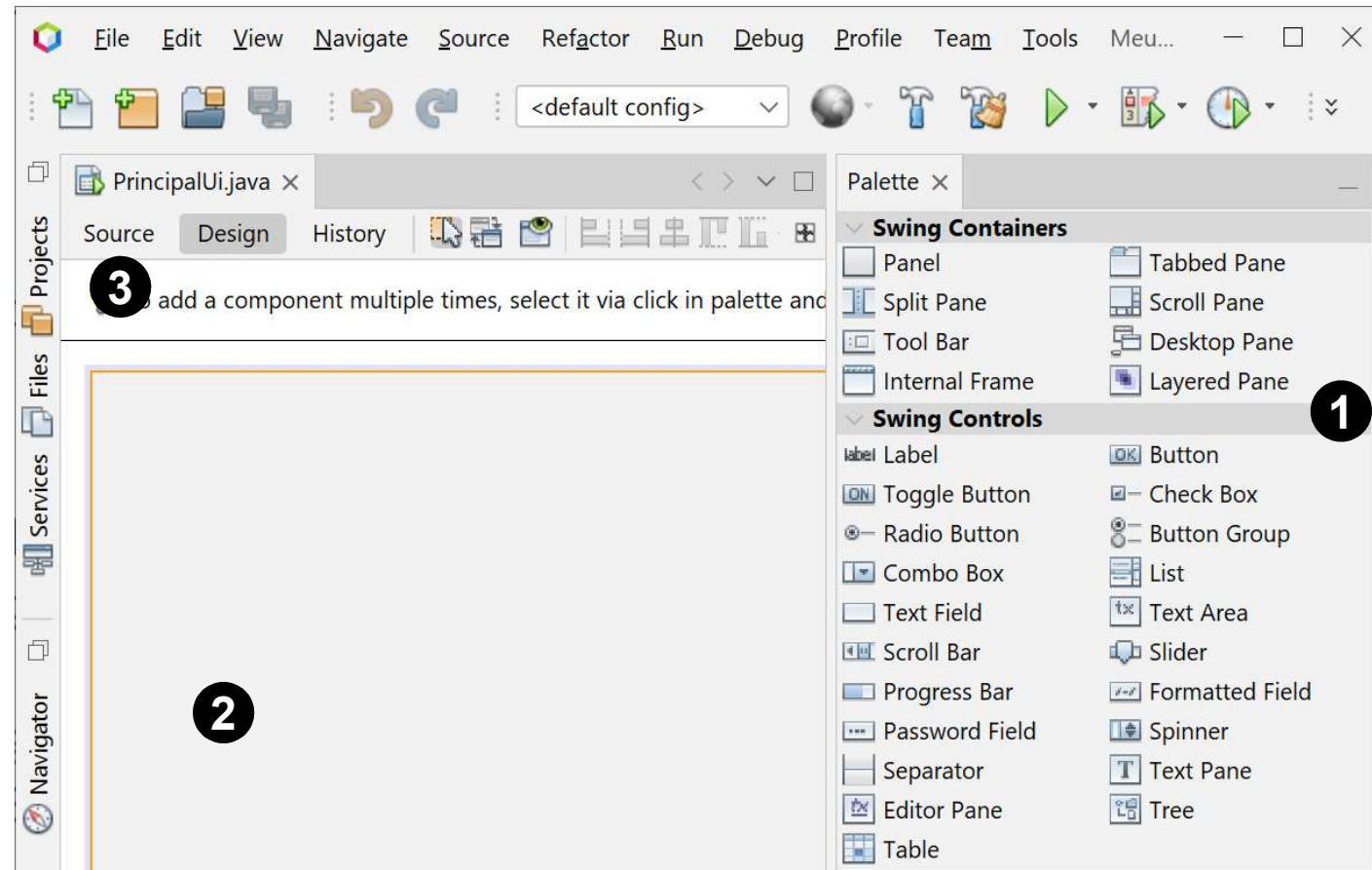
- Para criar uma GUI:
 - Clicar em *File > New File*
 - Em *Categories*, selecionar ***Swing GUI Forms***
 - Em *Tipos de Arquivo*, selecionar ***JFrame Form***
 - Clicar em ***Next***
 - Em ***Nome da Classe*** informar um nome para a classe da GUI

Utilizando o GUI Builder

1 – Selecione um componente na *Paleta de Componentes*

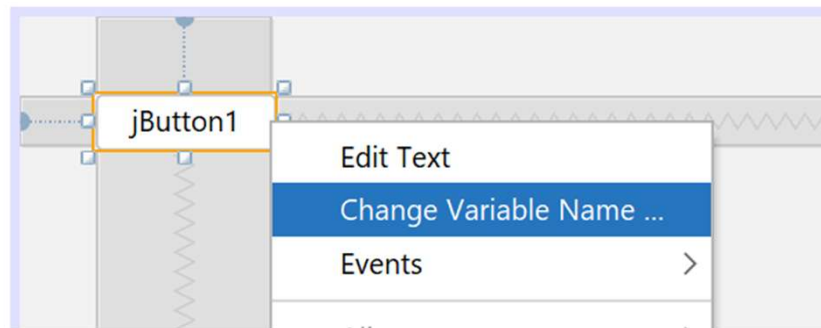
2 – Clique sobre a *área de desenho* para incluir o componente previamente selecionado

3 – Clique em *Source* para visualizar o código Java gerado pelo GUI Builder. Para retornar ao desenho da GUI, clique em *Design*



Utilizando o GUI Builder

Como cada componente GUI constitui de um objeto Java, é possível definir o nome de cada variável que referencia o objeto











1 - Clicar com o botão direito sobre o componente

2 - Selecionar *Change Variable Name*

3 - Informar um novo nome de variável

É altamente recomendável utilizar nomes legíveis para as variáveis que referenciam os objetos de controles GUI.

Alguns componentes GUI

Nome do componente na Paleta de Componentes do GUI Builder	Classe do Componente
 Label	JLabel
 Text Field	TextField
 Button	JButton
 Check Box	JCheckBox
 Combo Box	JComboBox
 List	JList
 Panel	JPanel
 Table	JTable

Programação orientada a eventos

- É um estilo de programação na qual o fluxo de execução do programa é determinado por eventos
- Um evento é um sinal recebido pelo programa indicando que algo aconteceu
- São considerados eventos:
 - **Ações do usuário: movimentos do mouse, teclado**
 - Mensagens de outros programas
 - Periféricos enviando sinais ao programa
- O programa pode escolher responder ou ignorar o evento

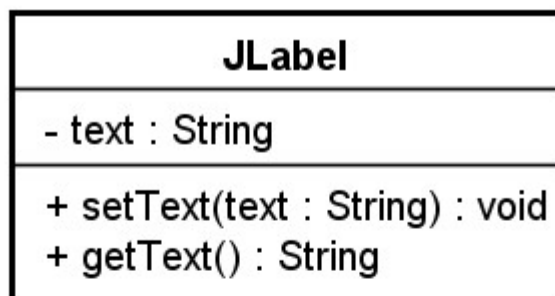
Programando um evento

- Na *área de desenho*, ao clicar duplo sobre um *botão*, o *GUI Builder* cria um método para inserção do programa que será executado quando o usuário clicar sobre o botão.
- Exemplo:

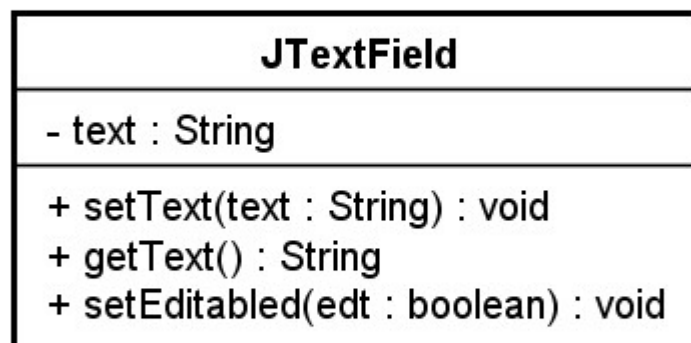
```
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
      
}
```

Digitar aqui o que
deve acontecer
quando o botão for
clicado pelo usuário

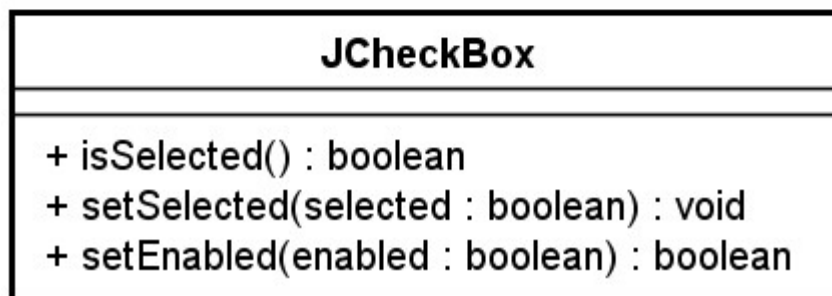
Diagrama de classes de alguns componentes



label Label



Campo de Texto



☒ Caixa de Seleção

JLabel


JLabel
- text : String
+ setText(text : String) : void + getText() : String

label Label

Método	Descrição
getText()	Obtém o valor exibido no componente
setText()	Altera o valor exibido pelo componente

JTextField

JTextField
- text : String
+ setText(text : String) : void + getText() : String + setEditable(edt : boolean) : void

 Text Field

Método	Descrição
getText()	Obtém o valor exibido no componente
setText()	Altera o valor exibido pelo componente
setEditable()	Habilita ou desabilita a edição

JCheckBox

JCheckBox
+ isSelected() : boolean + setSelected(selected : boolean) : void + setEnabled(enabled : boolean) : boolean



Método	Descrição
isSelected()	Retorna true se o componente estiver marcado
setSelected()	Marca ou desmarca a caixa de seleção
setEnabled()	Habilita ou desabilita a edição