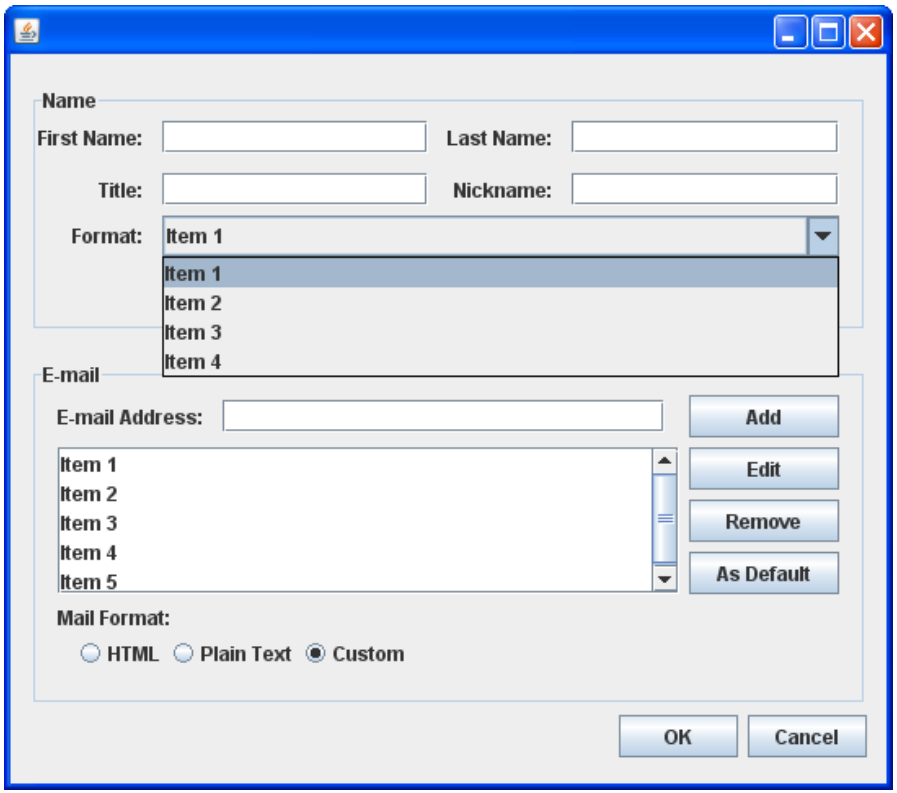
Técnicas de Programação II

Aula 03 – GUI e Swing

Edirlei Soares de Lima
<edirlei.lima@uniriotec.br>

GUI – Graphical User Interface

- A API Java fornece diversas classes destinadas a criação de interfaces gráficas.

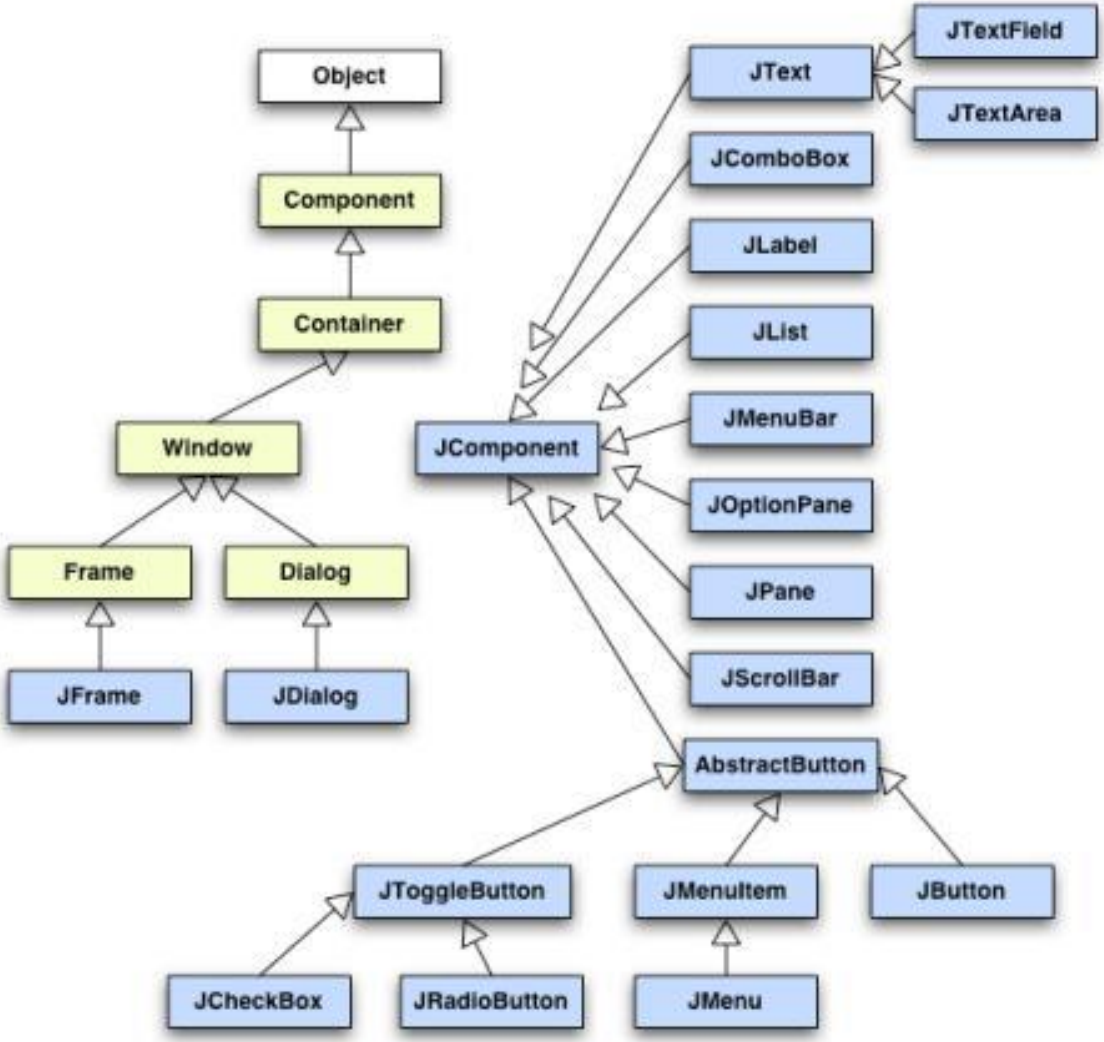


AWT e Swing

- **AWT (Abstract Window Toolkit):**
  - API padrão para criação de componentes GUI no início da plataforma Java (entre 1995 até 1998);
  - Os objetos AWT são construídos sobre objetos de código nativo do Sistema Operacional em uso;
  - Os componentes GUI originais do pacote java.awt estão diretamente associados com as capacidades de GUI da plataforma local.

- **Swing:**
  - API escrita puramente em Java (padrão desde 1998);
  - JLabel, JButton, JTextField, etc, são componentes GUI do pacote javax.swing.
  - Mais flexível que o java.awt porque é implementada toda em Java, enquanto que java.awt é implementada em código nativo.

AWT e Swing


Object
Component
Container
Window
Frame
Dialog
JFrame
JDialog
JComponent
JText
JTextField
JTextArea
JComboBox
JLabel
JList
JMenuBar
JOptionPane
JPane
JScrollBar
AbstractButton
JToggleButton
JMenuItem
JButton
JCheckBox
JRadioButton
JMenu

Containers e Componentes

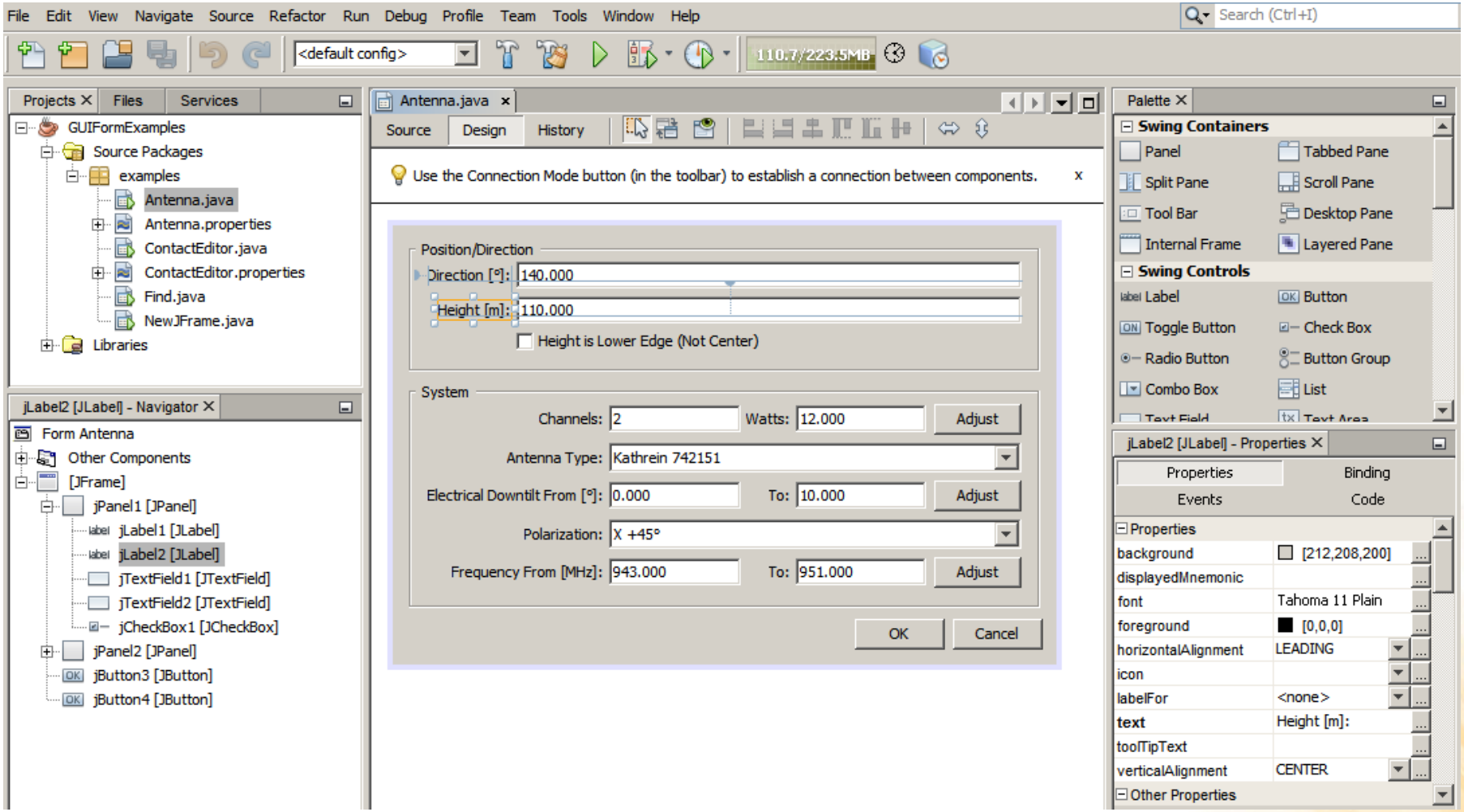
- Uma interface gráfica em Java é baseada em dois elementos:
  - **Containers**: servem para agrupar e exibir outros componentes
  - **Componentes**: botões, labels, scrollbars, etc.

- Todo programa Java que ofereça uma interface possuí pelo menos um container.

- Uma janela de nível mais alto (que não fica contida dentro de outra janela) é um Frame ou, na versão Swing, um **JFrame**;

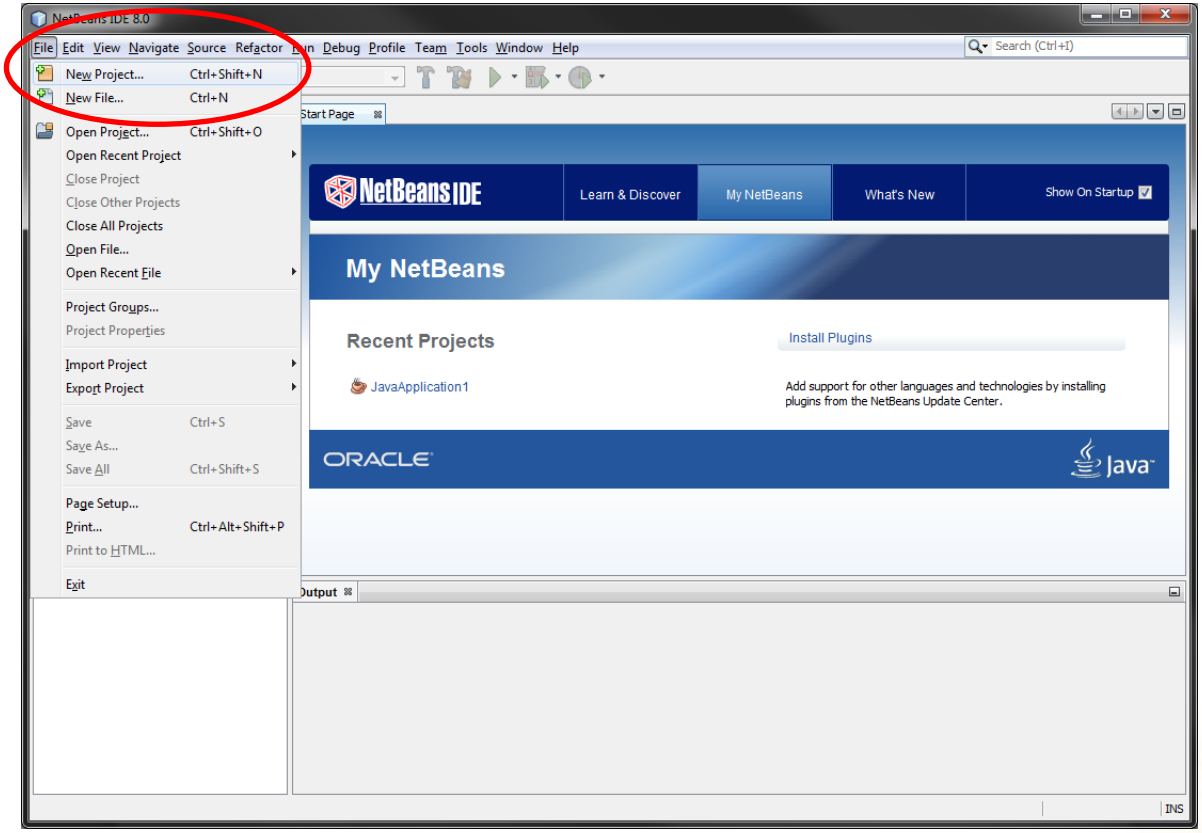
- O JFrame é um container. Isso significa que ele pode conter outros componentes de interface com o usuário.

Netbeans GUI Builder



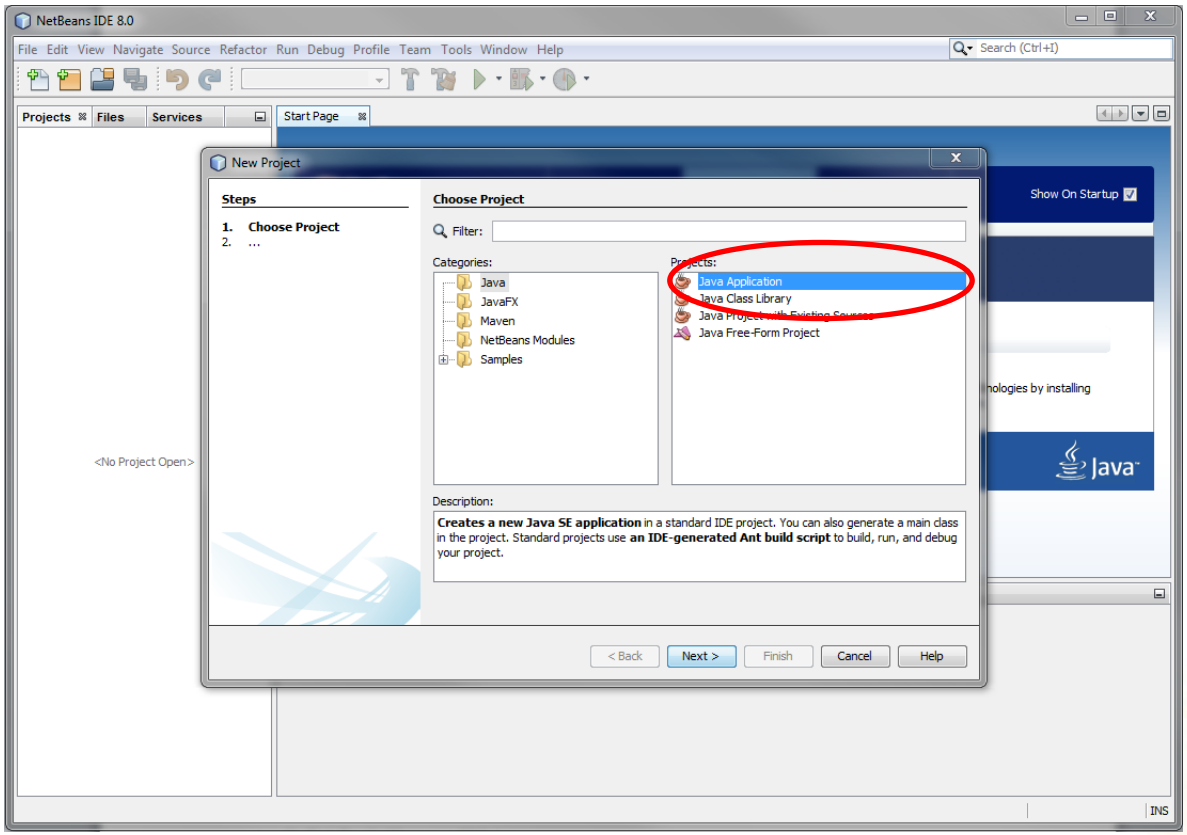
Netbeans GUI Builder – Criando Projeto

1) Acesse o menu File->New Project...



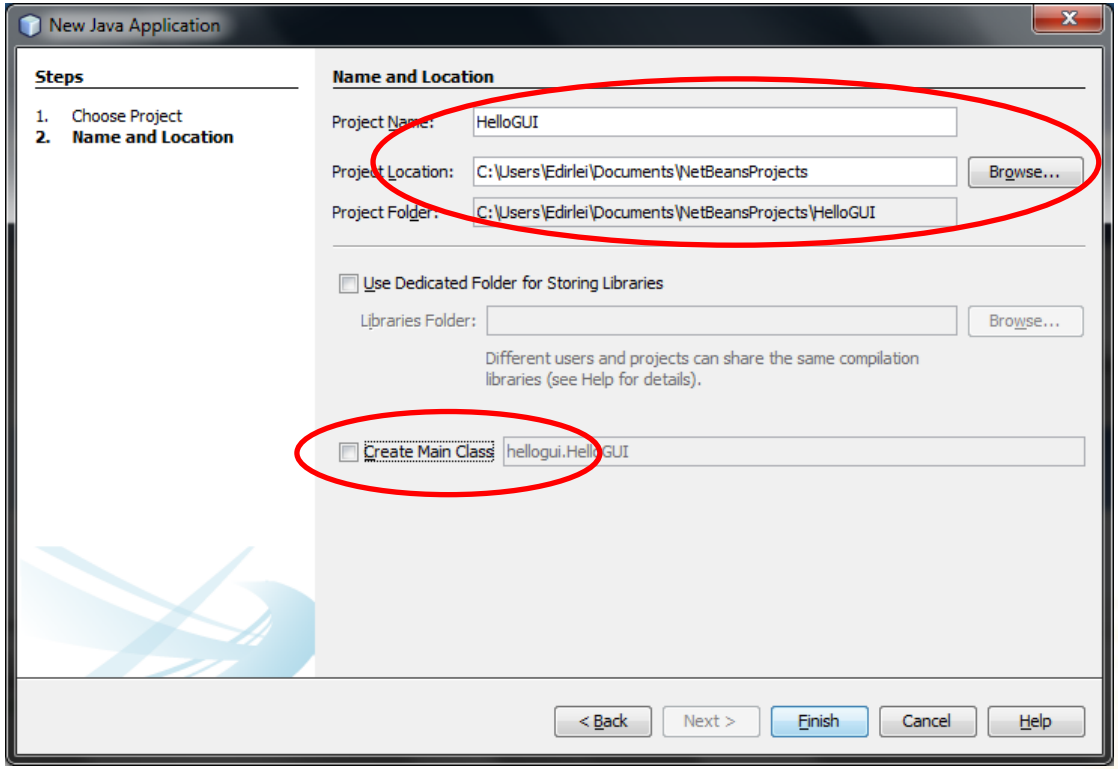
Netbeans GUI Builder – Criando Projeto

2) Selecione o tipo de projeto “Java Application” e em seguida clique em “Next”:



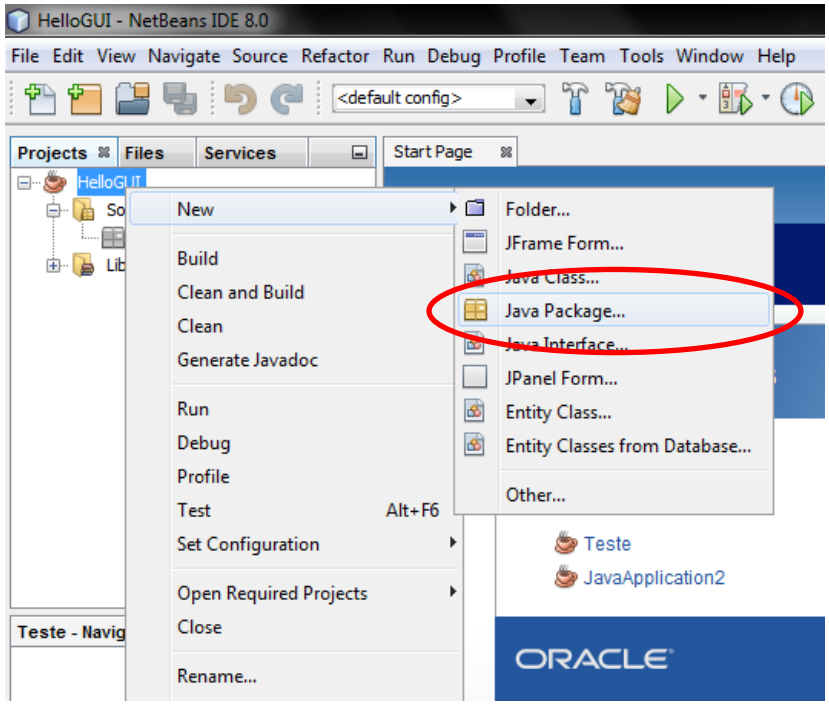
Netbeans GUI Builder – Criando Projeto

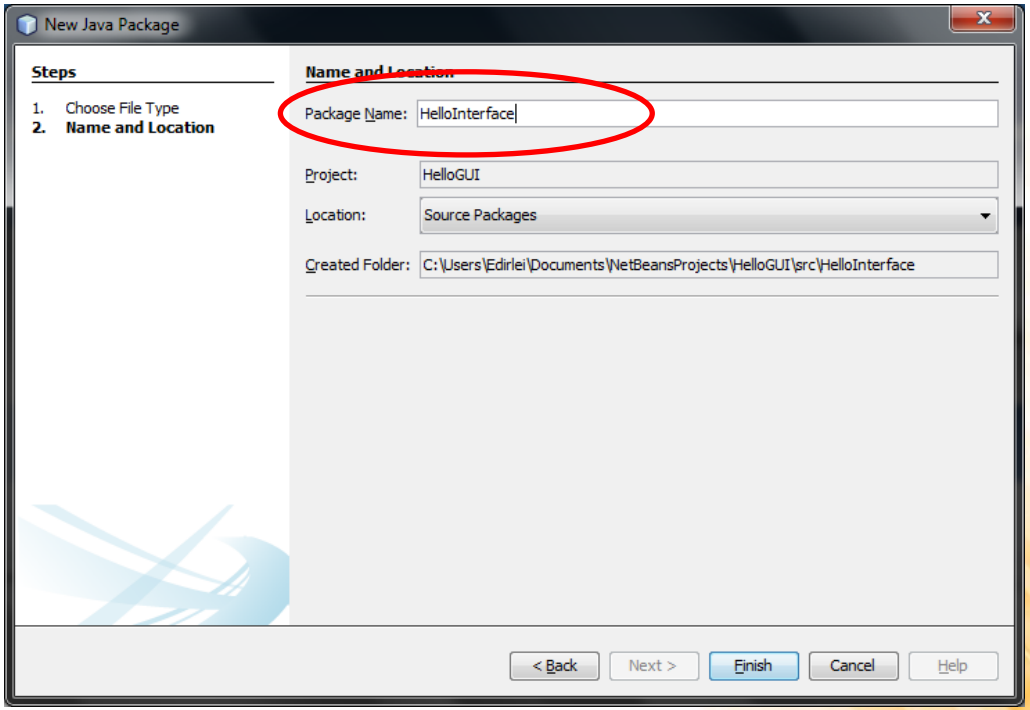
3) De um nome para o projeto, selecione o local onde ele será salvo e desmarque a opção “Create Main Class”. Em seguida clique em “Finish”:



Netbeans GUI Builder – Criando Projeto

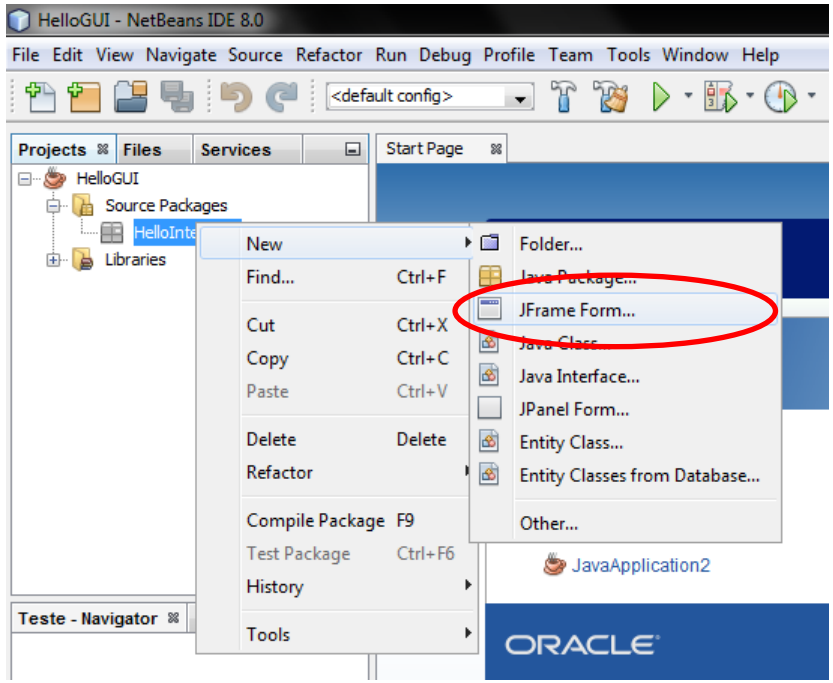
4) Crie um novo “Java Package” no projeto:

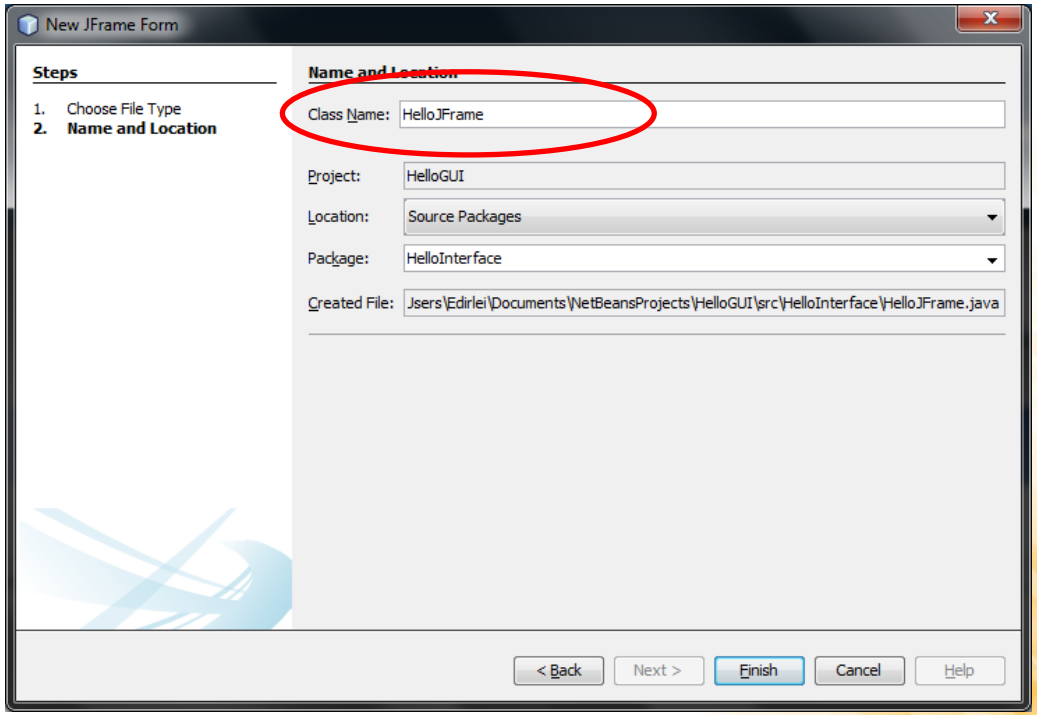




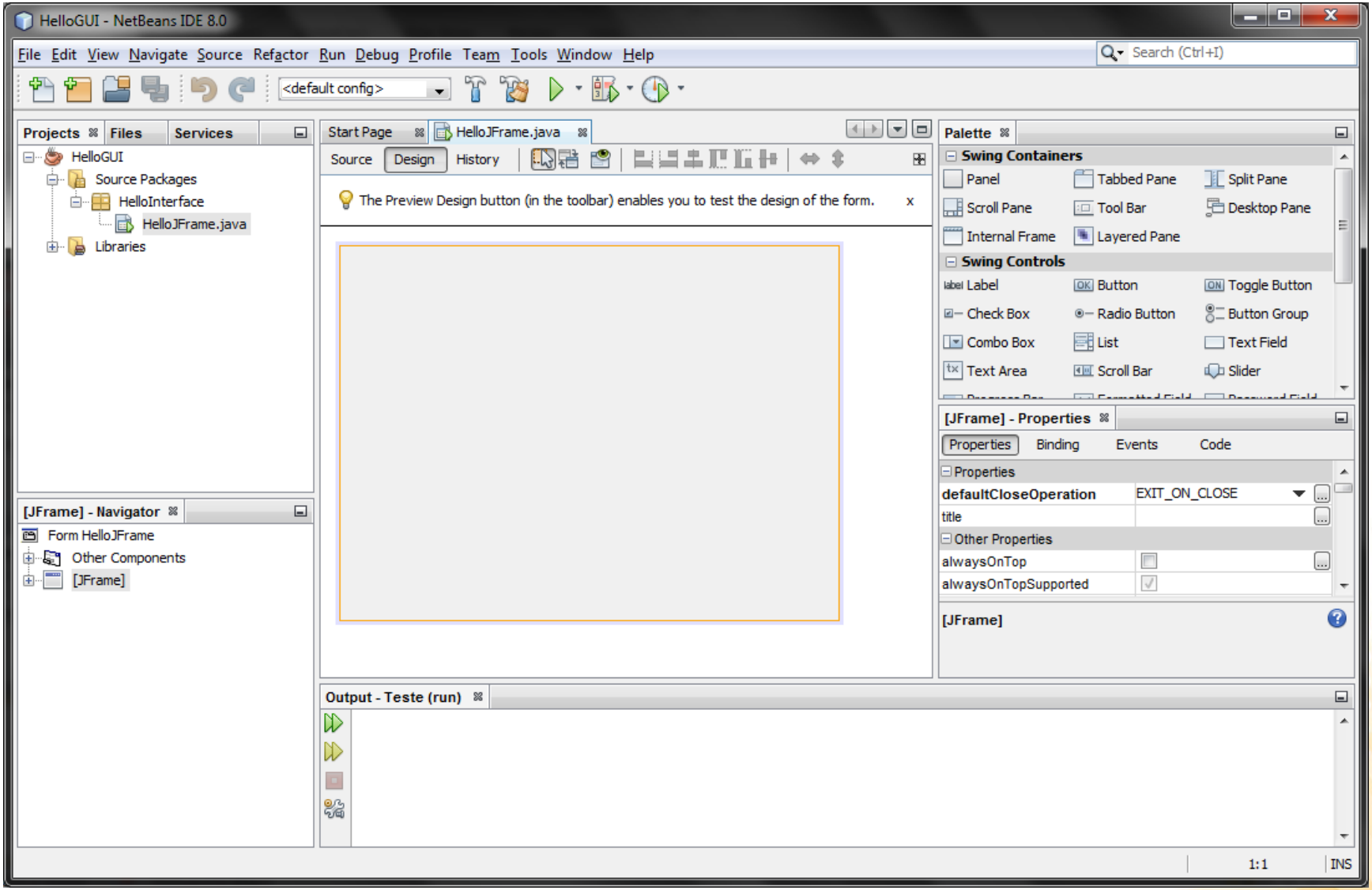
Netbeans GUI Builder – Criando Projeto

5) Crie um novo “JFrame Form”:



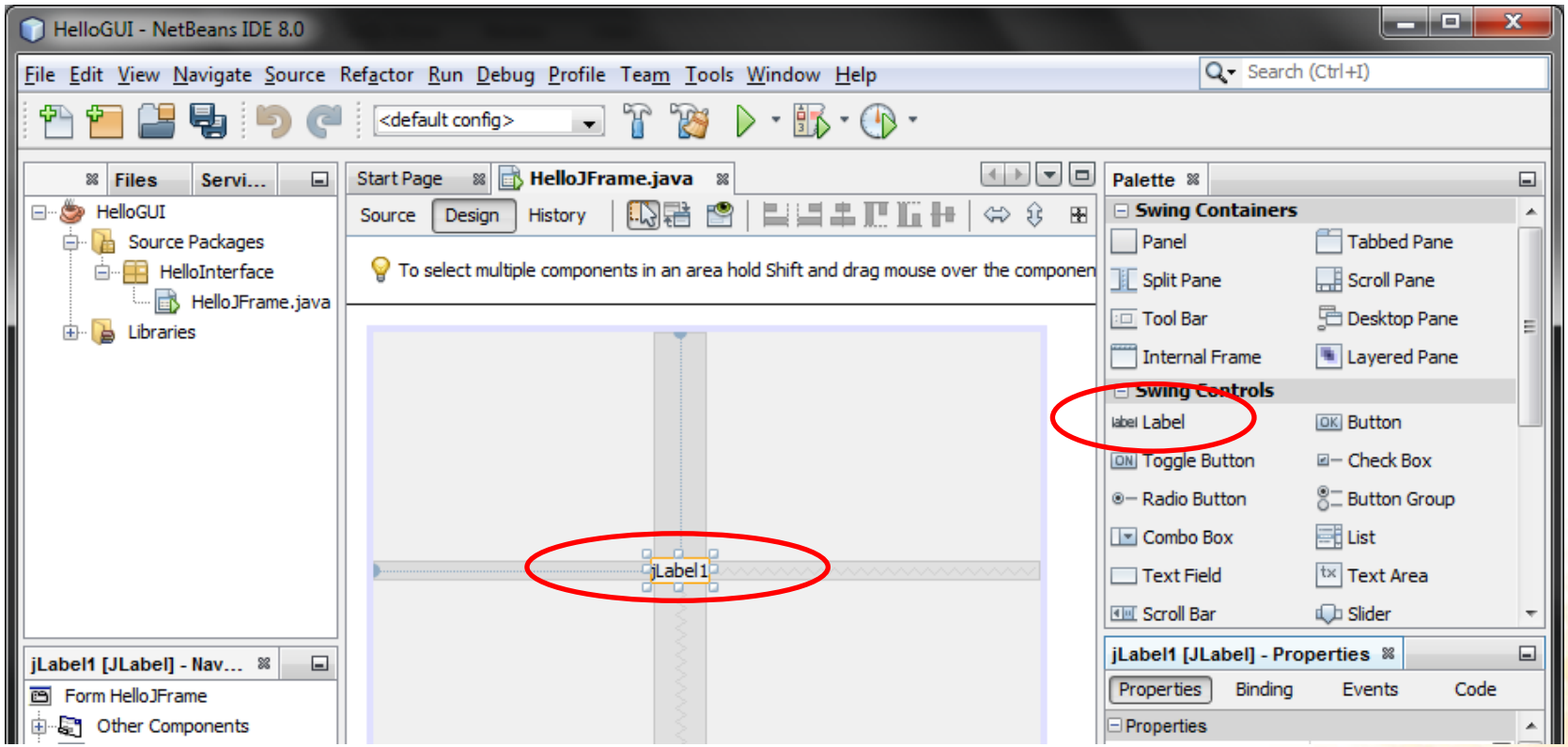


Netbeans GUI Builder



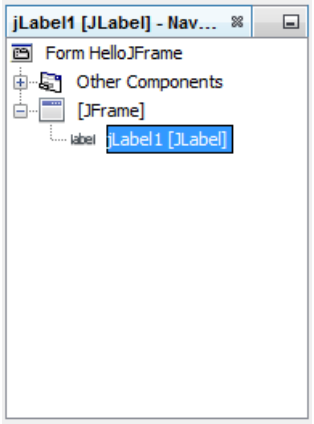
Componentes Básicos - Label

- Componente para exibição de texto não-editável ou ícones.



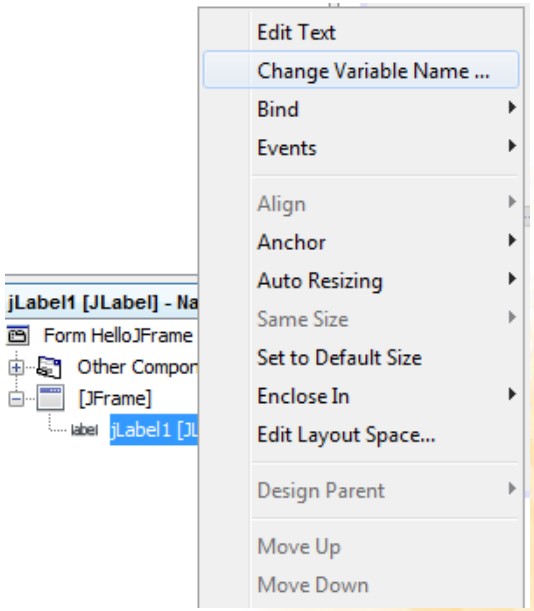
Componentes Básicos - Label

- Containers e componentes e estrutura da interface gráfica;



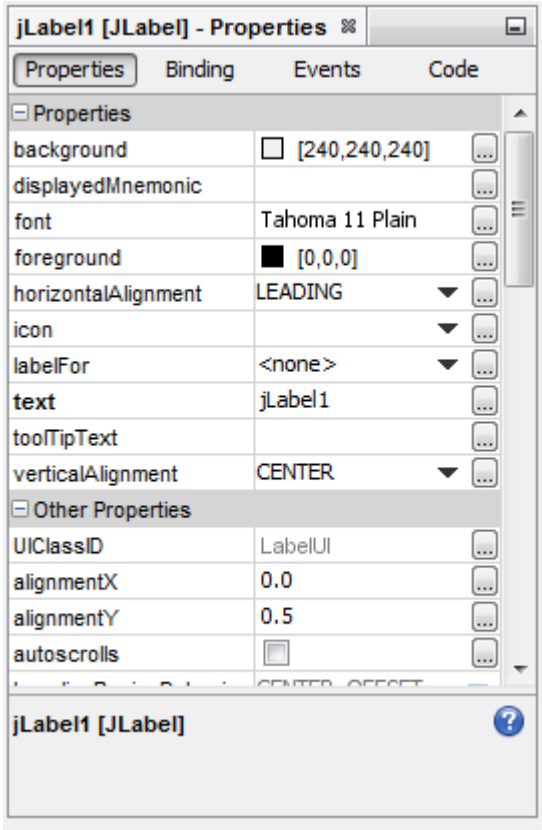
- Todos os elementos que fazem parte da interface gráfica são **objetos**;

- Todos os objetos possuem um <u>nome</u> (variable name) que pode (e deve!) ser alterado.



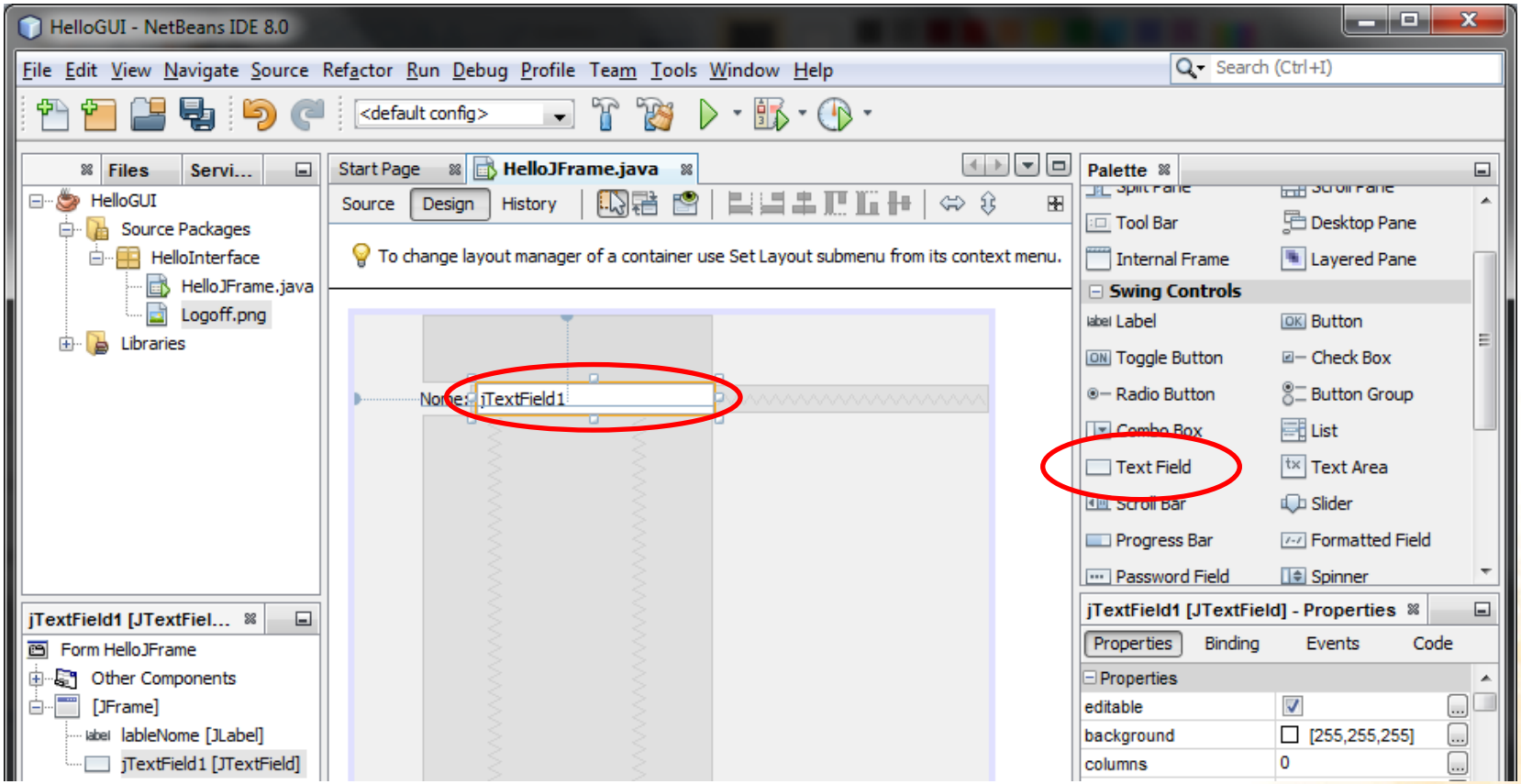
Componentes Básicos - Label

- Principais Propriedades (JLable):
  - text;
  - foreground;
  - background;
  - font;
  - icon;
  - toolTipText;
  - border;



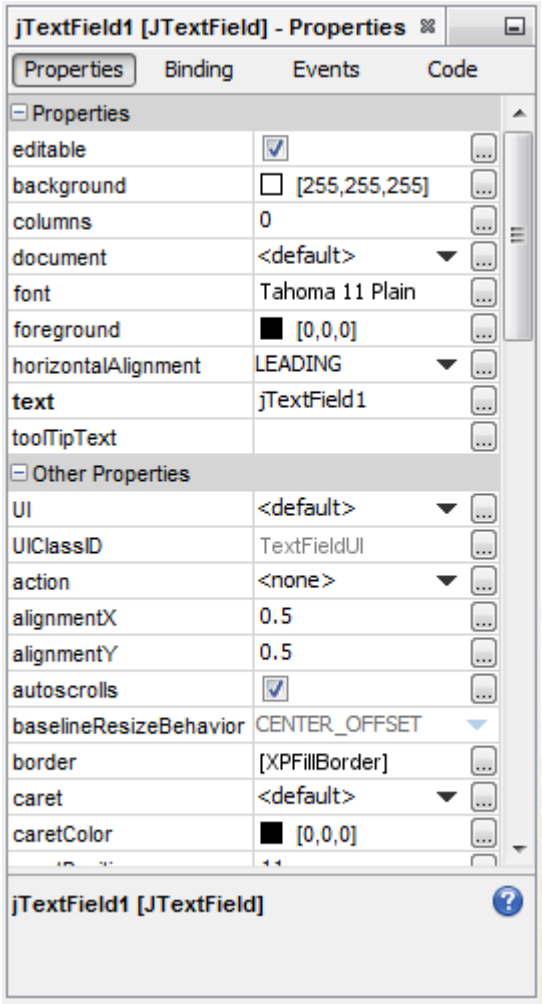
Componentes Básicos – TextField

- Componente para entrada, edição e exibição de texto.



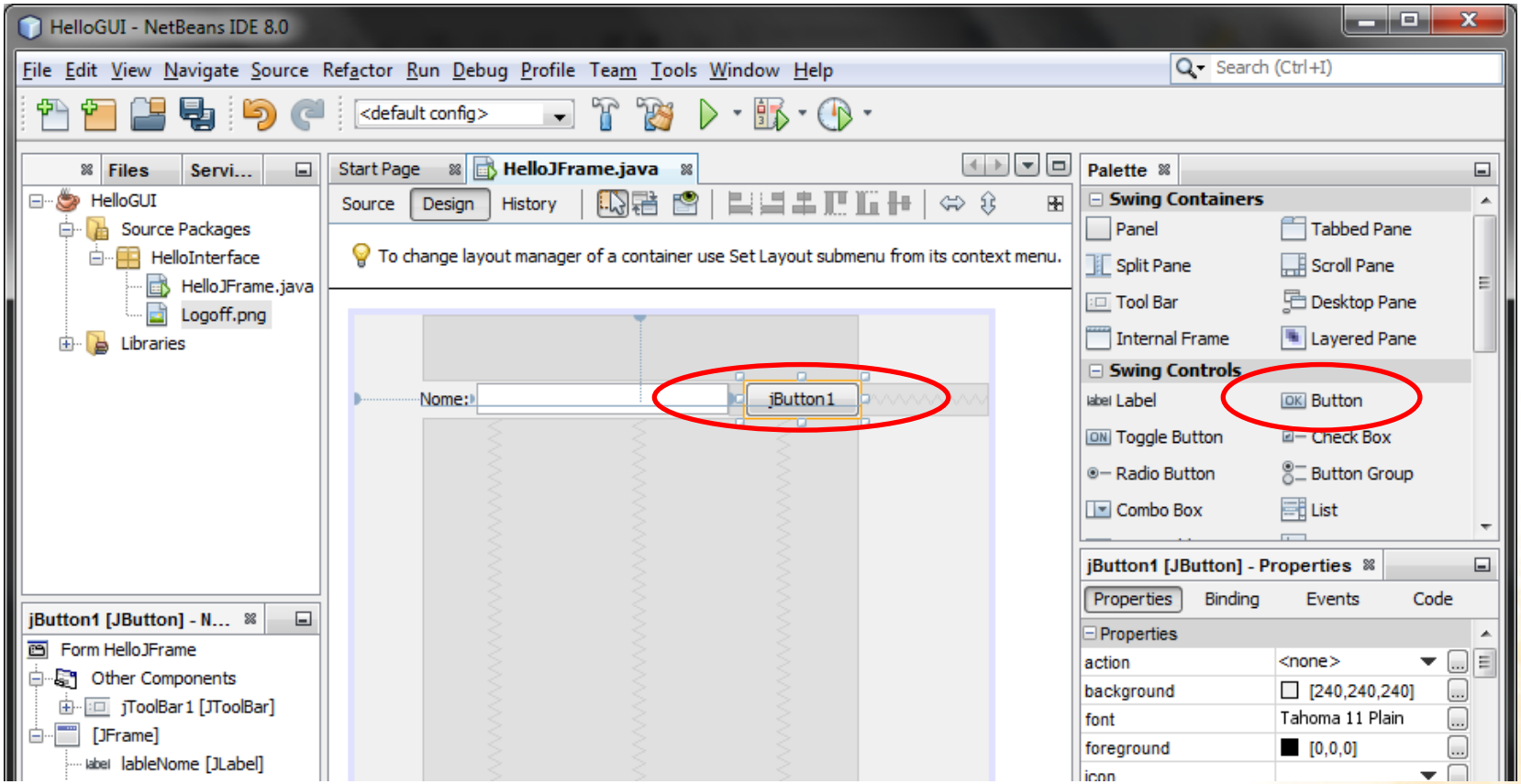
Componentes Básicos – TextField

- Principais Propriedades (JTextField):
  - text;
  - editable;
  - foreground;
  - background;
  - font;
  - toolTipText;
  - border;
  - enabled;



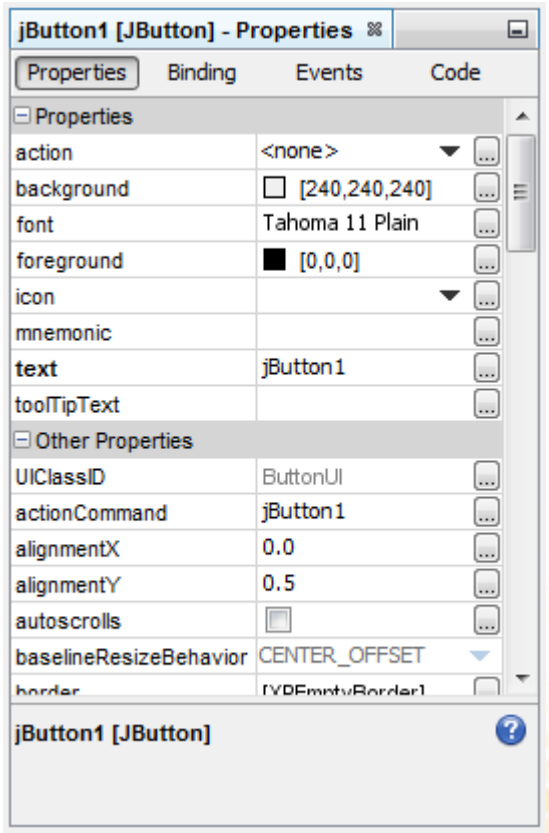
Componentes Básicos – Button

- Componente que representa um botão.



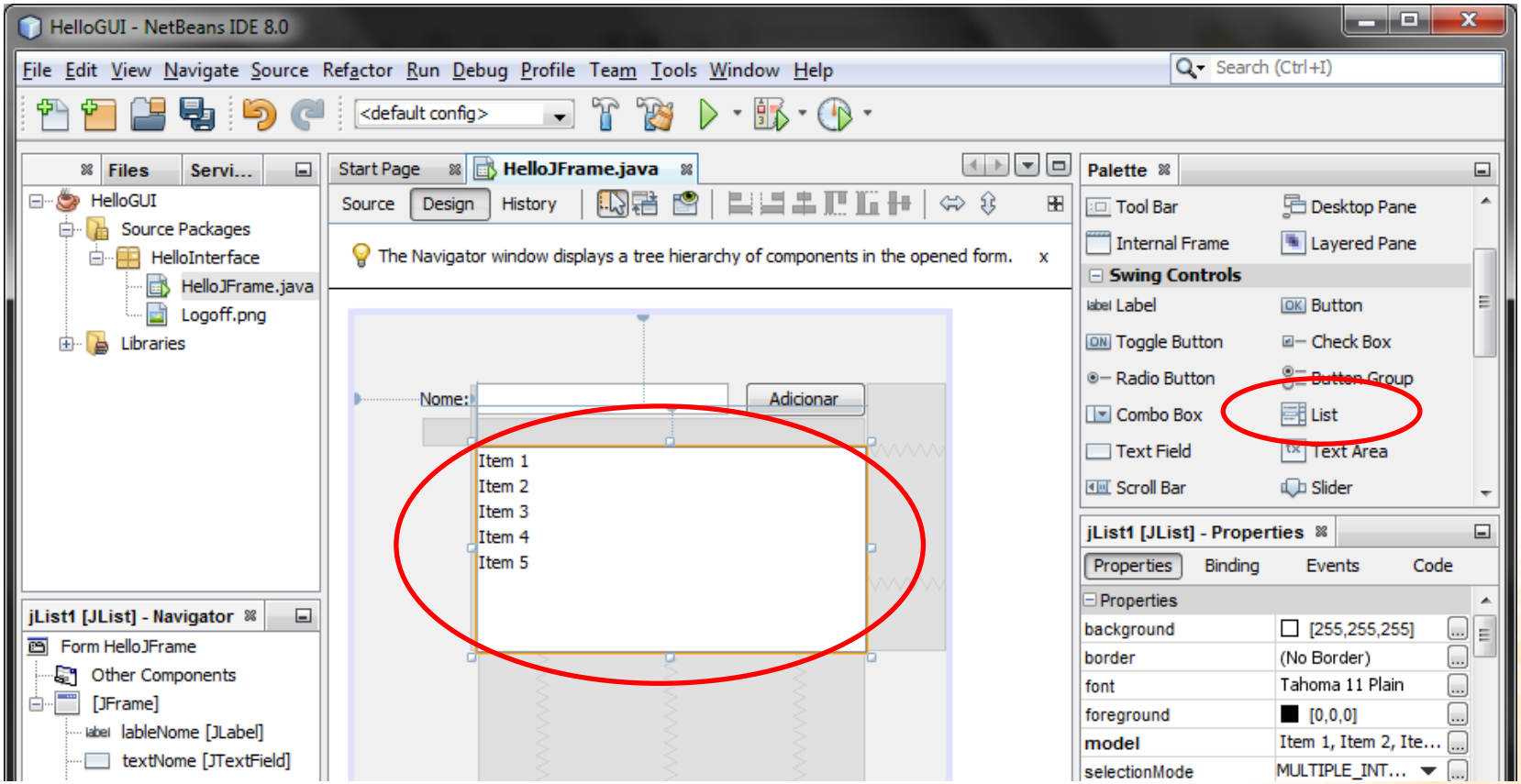
Componentes Básicos – Button

- Principais Propriedades (JButton):
  - text;
  - foreground;
  - background;
  - font;
  - icon;
  - toolTipText;
  - border;
  - enabled;



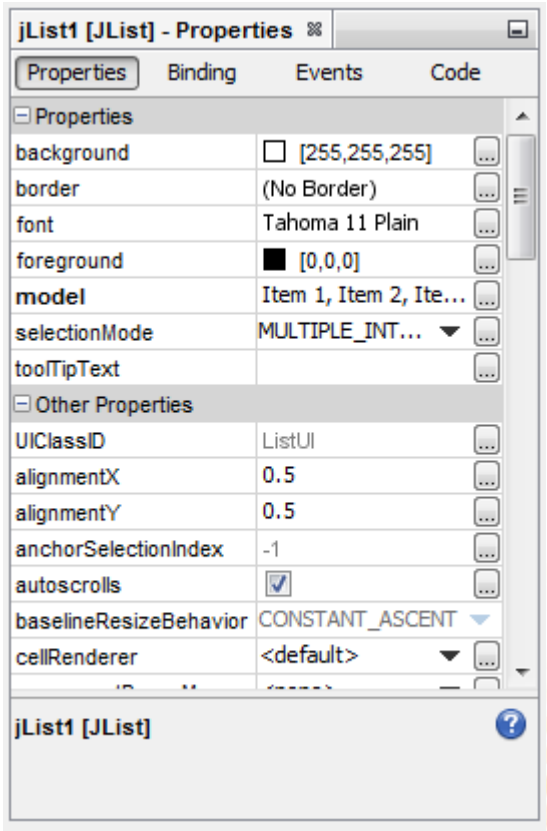
Componentes Básicos – List

- Componente que exibe uma lista de itens e permite que o usuário possa seleciona-los.

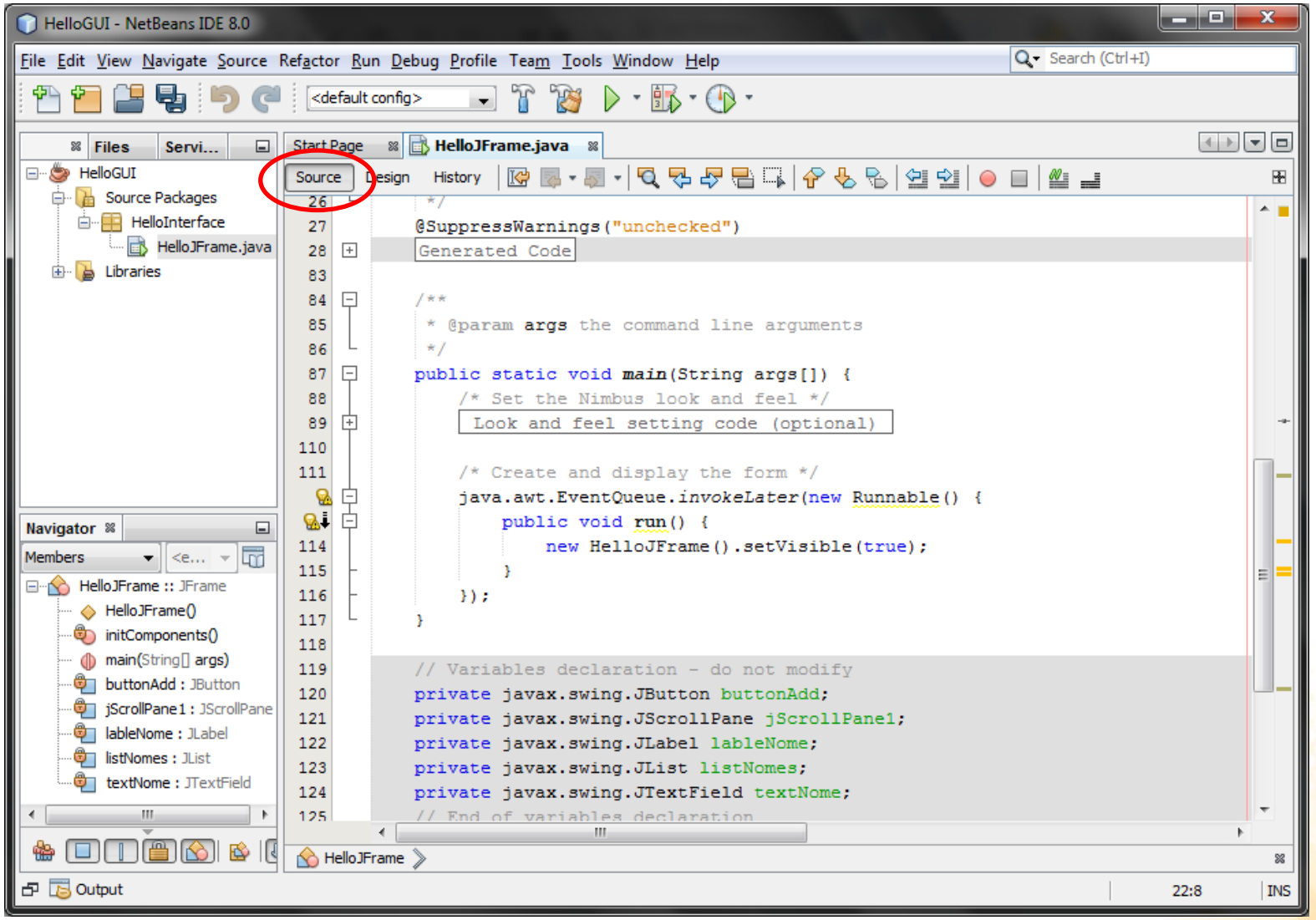


Componentes Básicos – List

- Principais Propriedades (JList):
  - model;
  - selectionMode;
  - selectedIndex;
  - visibleRowCount;
  - foreground;
  - background;
  - font;
  - toolTipText;
  - border;
  - enabled;

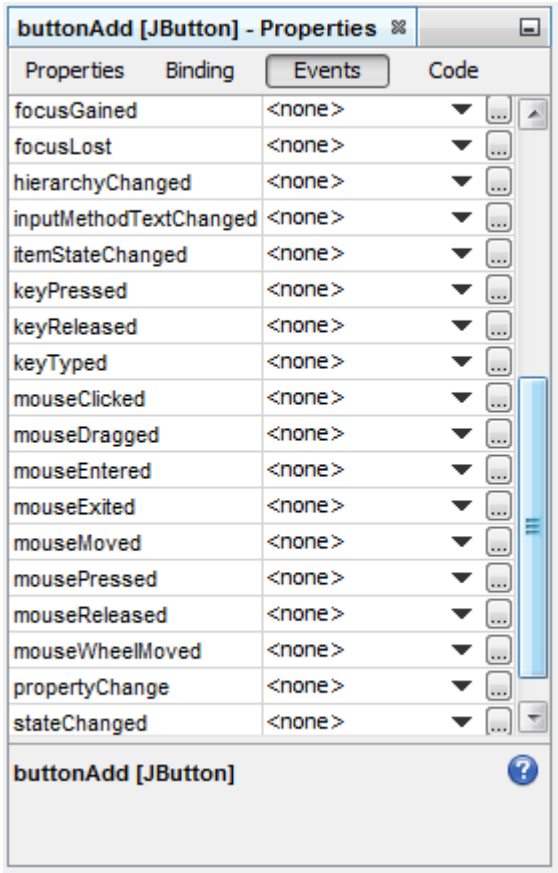


Netbeans GUI Builder – Código Gerado



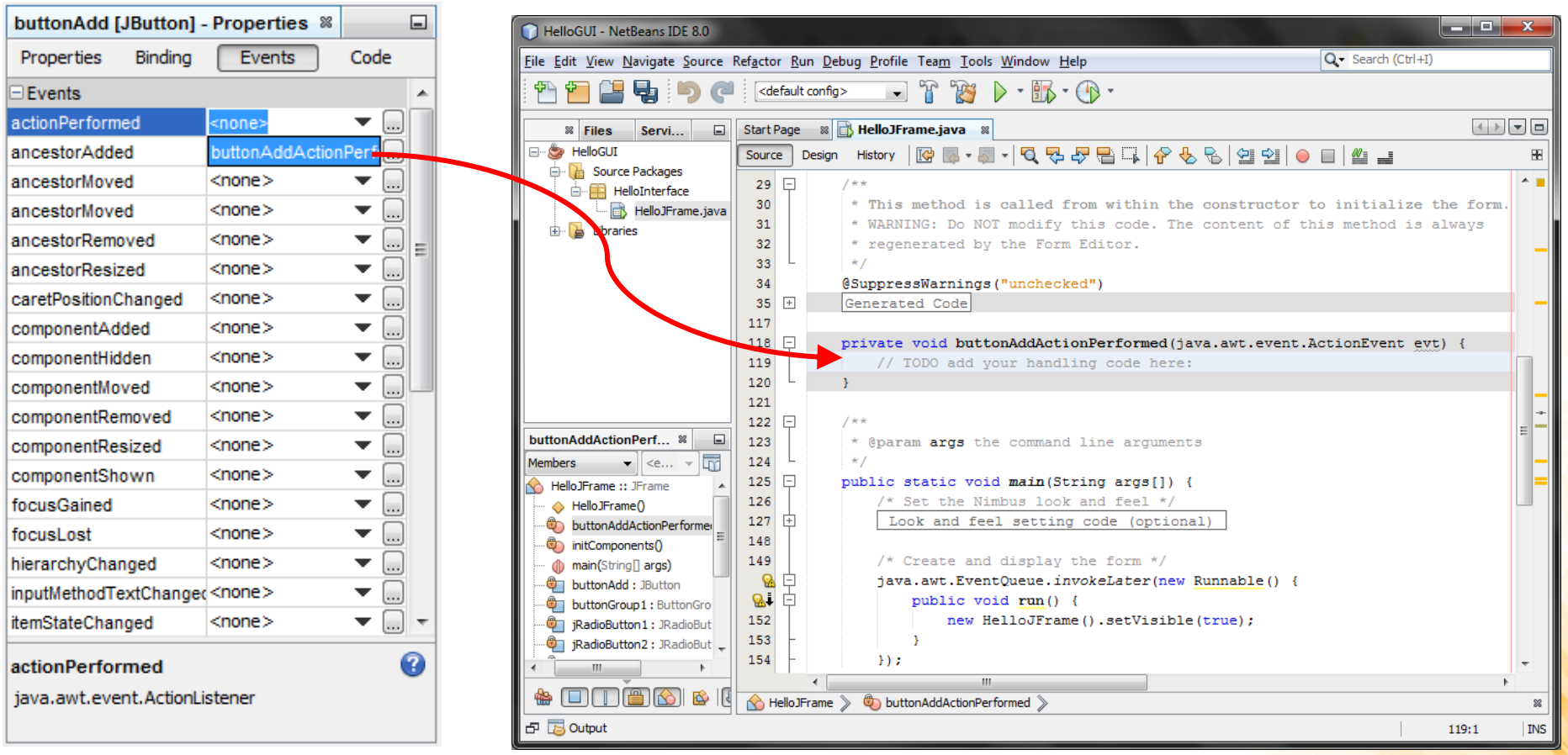
Eventos – Button

- Principais Eventos (JButton):
  - actionPerformed;
  - mouseClicked;
  - mousePressed;
  - mouseRelesead;
  - mouseMoved;
  - mouseEntered
  - mouseExited;
  - focusGained;
  - focusLost;



Evento actionPerformed – Button

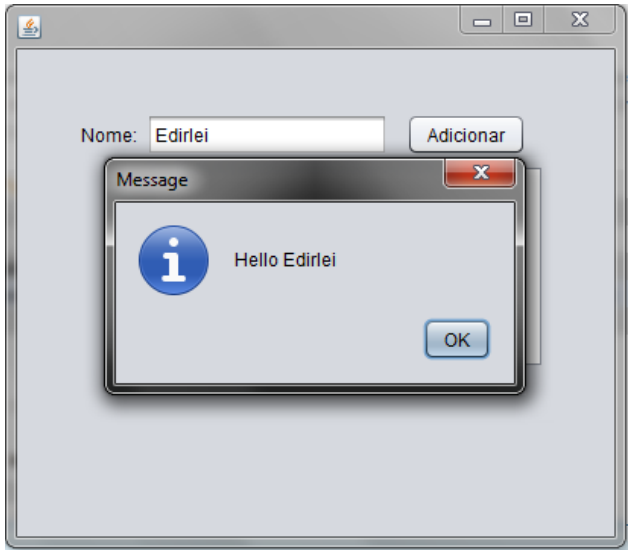
- Criando um evento de ativação para o botão (actionPerformed):



Evento actionPerformed – Button

- **Exemplo 1 –** Mostrar mensagem com o conteúdo do TextField:

```
private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
{

  JOptionPane.showMessageDialog(this, "Hello " + textNome.getText());

}
```



Evento actionPerformed – Button

- **Exemplo 2 –** Adicionar o conteúdo do TextField na List:

a) Crie um objeto do tipo `DefaultListModel` para armazenar os elementos da lista:

```
...
public class HelloJFrame extends javax.swing.JFrame {

    private DefaultListModel listModel = new DefaultListModel();
...
```

b) No método construtor da classe, associe o `listModel` a lista:

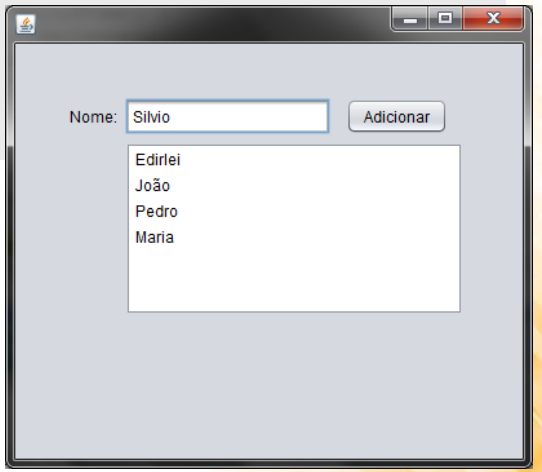
```
...
public HelloJFrame() {
  initComponents();

  listNomes.setModel(listModel);
}
...
```

Evento actionPerformed – Button

- **Exemplo 2 –** Adicionar o conteúdo do TextField na List:

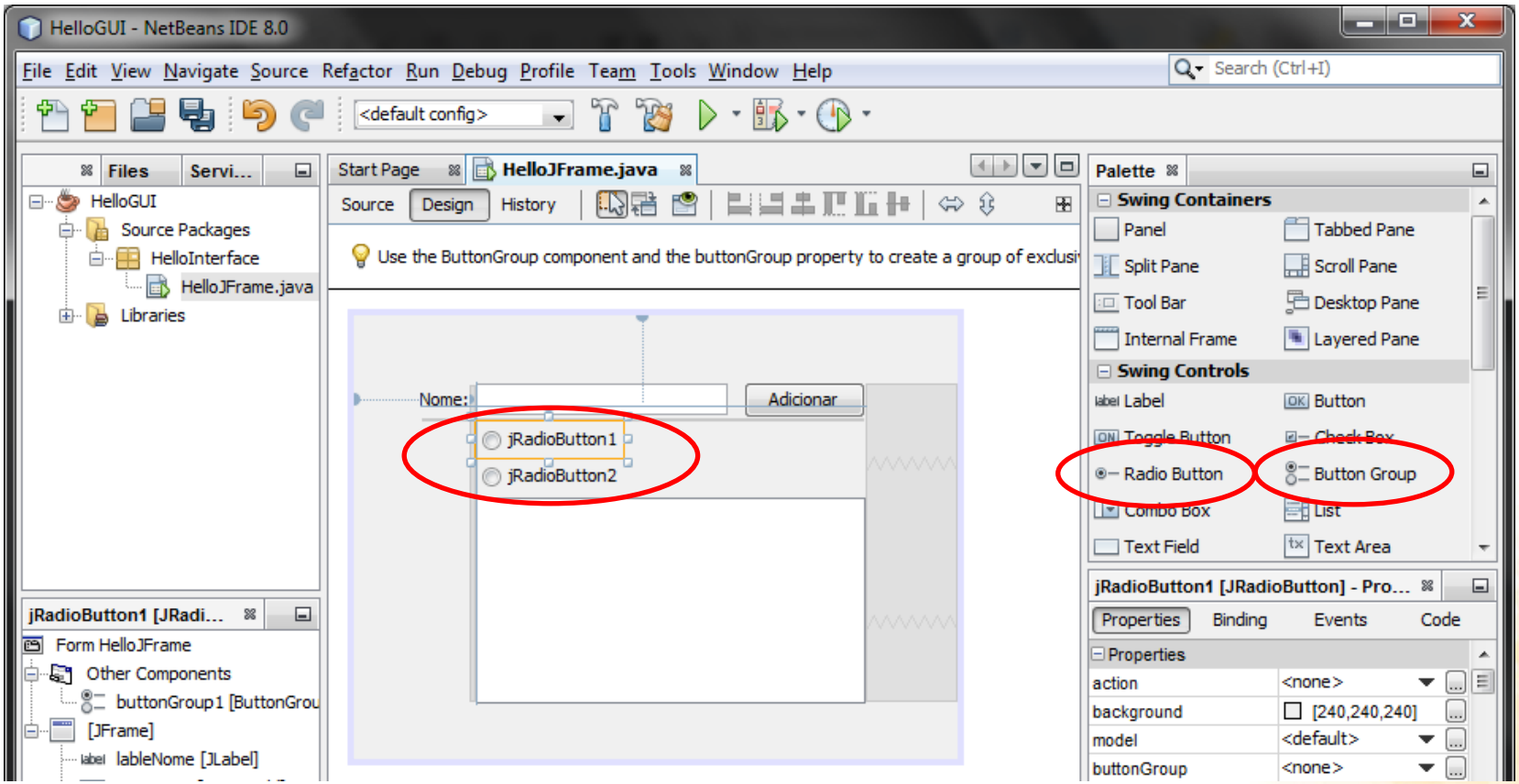
c) No evento actionPerformed, adicione o conteúdo do TextField na List:

```
...

private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
{
  listModel.addElement(textNome.getText());
  textNome.setText("");
}

...
```



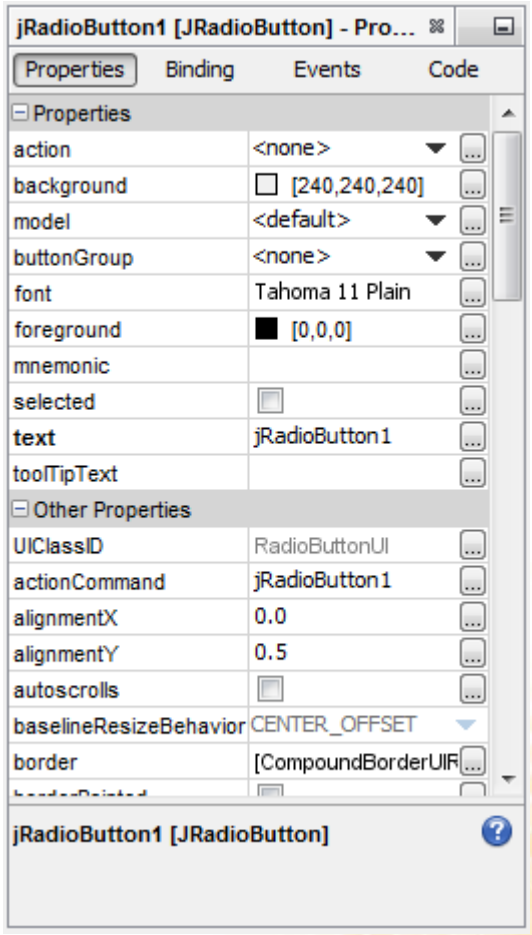
Componentes Básicos – RadioButton e ButtonGroup

- Componentes que permitem a seleção de opções.



Componentes Básicos – RadioButton

- Principais Propriedades (JRadioButton):
  - text;
  - buttonGroup;
  - Selected;
  - foreground;
  - background;
  - font;
  - icon;
  - toolTipText;
  - border;
  - enabled;



Usando o RadioButton

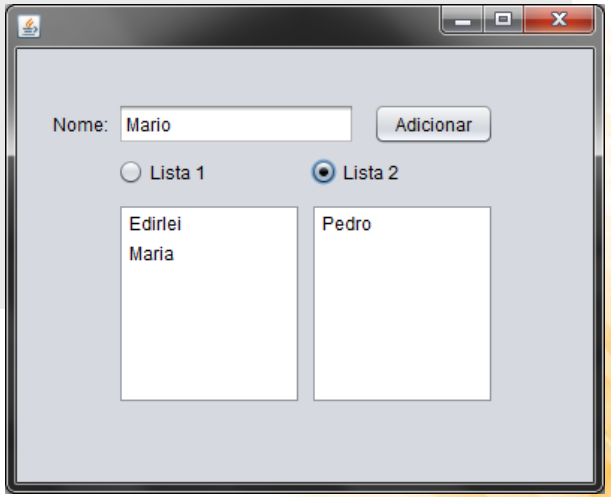
- **Exemplo 3 –** Adicionar o conteúdo do TextField em duas List de acordo com a opção selecionada no RadioButton.

  - **Interface:**

Usando o RadioButton

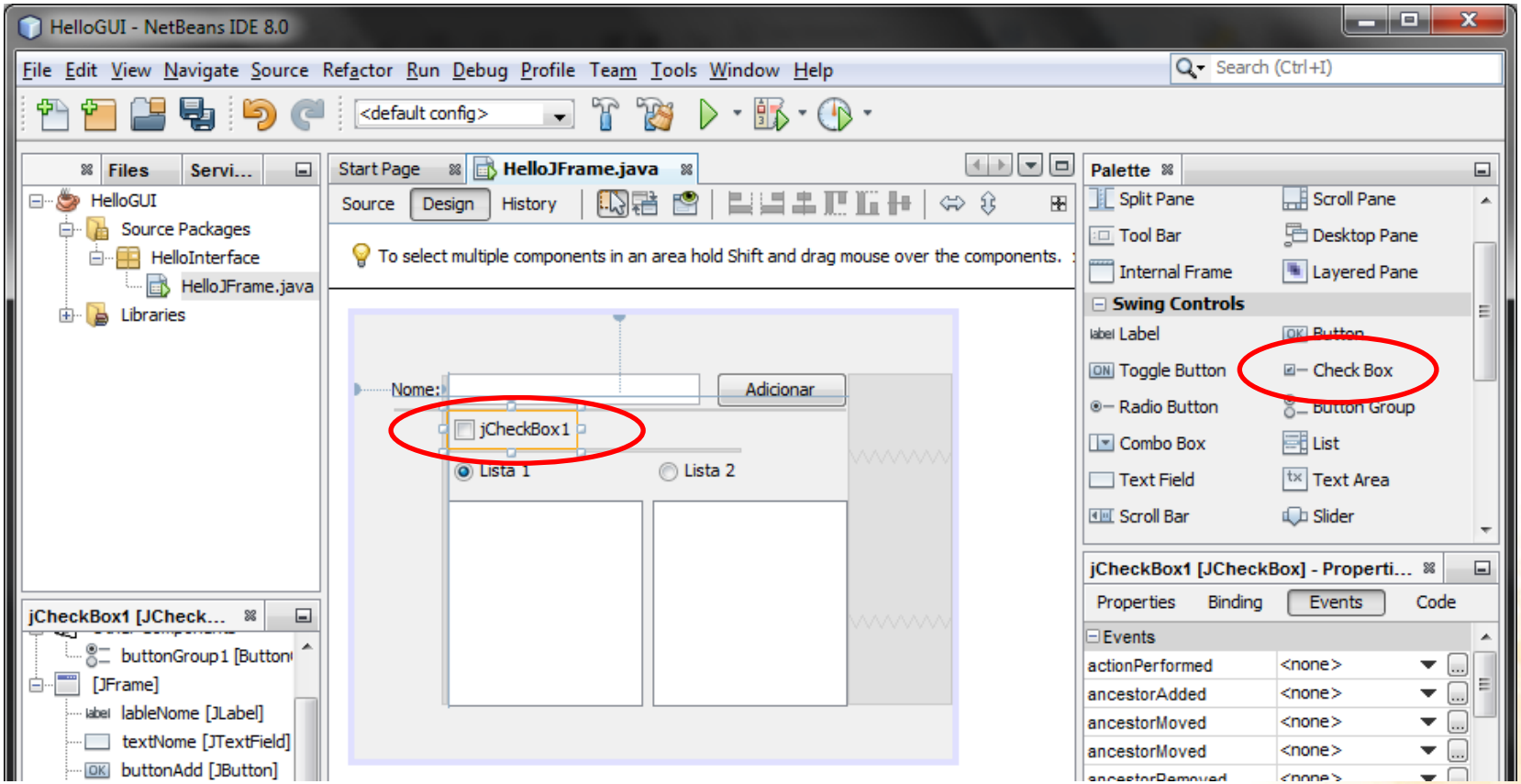
- **Exemplo 3 –** Adicionar o conteúdo do TextField nas Lists de acordo com a opção selecionada no RadioButton.

```
...

private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
{
  if (radioBt1.isSelected())
    listModel1.addElement(textNome.getText());
  else if (radioBt2.isSelected())
    listModel2.addElement(textNome.getText());

  textNome.setText("");
}

...
```



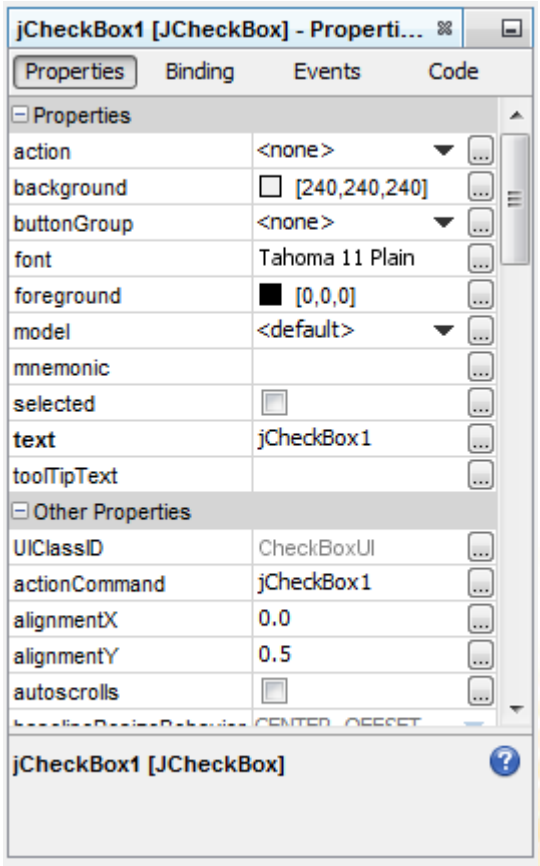
Componentes Básicos – CheckBox

- Componente que permite a seleção de opções (marcado ou não marcado).



Componentes Básicos – CheckBox

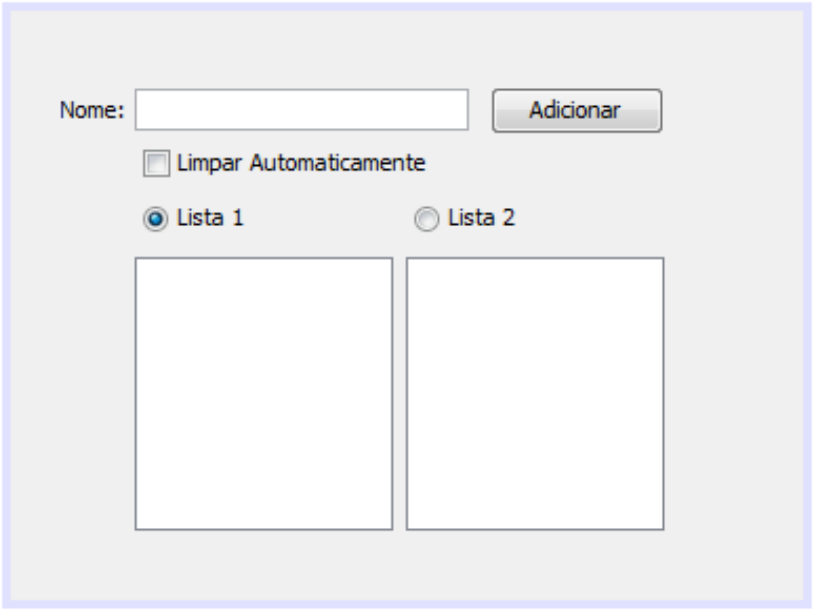
- Principais Propriedades (JCheckBox):
  - text;
  - buttonGroup;
  - Selected;
  - foreground;
  - background;
  - font;
  - icon;
  - toolTipText;
  - border;
  - enabled;



Usando o CheckBox

- **Exemplo 4 –** Permitir ao usuário escolher se ele deseja limpar automaticamente o conteúdo do TextField após adiciona-lo à List.

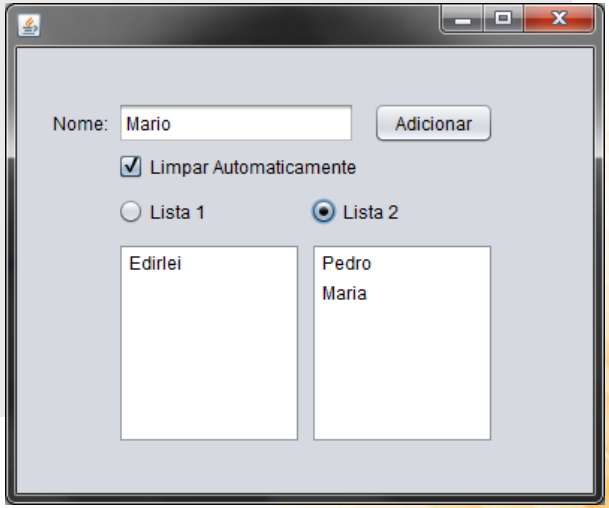
  - **Interface:**



Usando o CheckBox

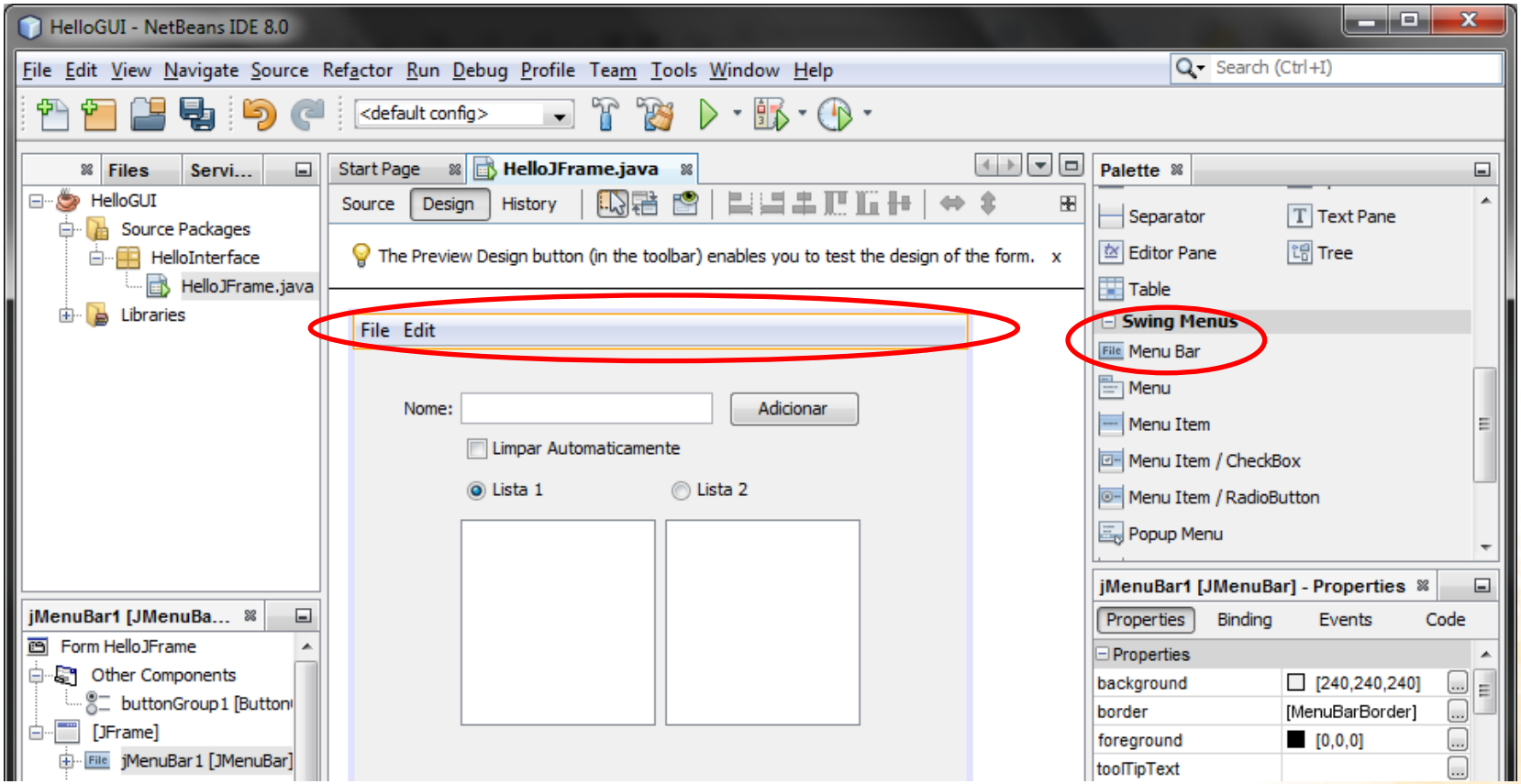
- **Exemplo 4 –** Permitir ao usuário escolher se ele deseja limpar automaticamente o conteúdo do TextField após adiciona-lo à List.

```
...

private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
{
  if (radioBt1.isSelected())
    listModel1.addElement(textNome.getText());
  else if (radioBt2.isSelected())
    listModel2.addElement(textNome.getText());

  if (checkBoxLimpar.isSelected())
    textNome.setText("");
}

...
```



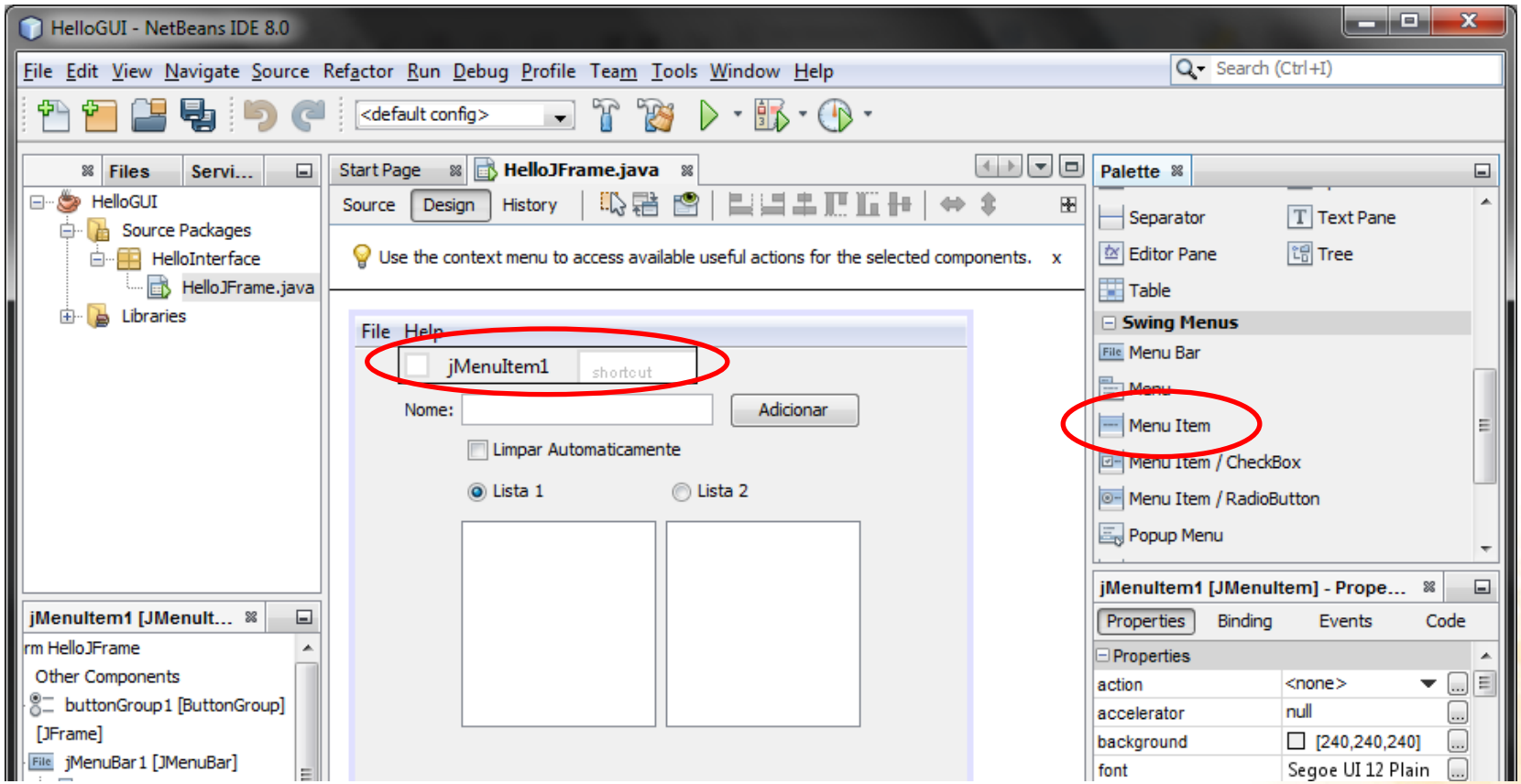
Componentes Básicos – MenuBar

- Componente que permite a criação de uma barra de menu.



Componentes Básicos – MenuItem

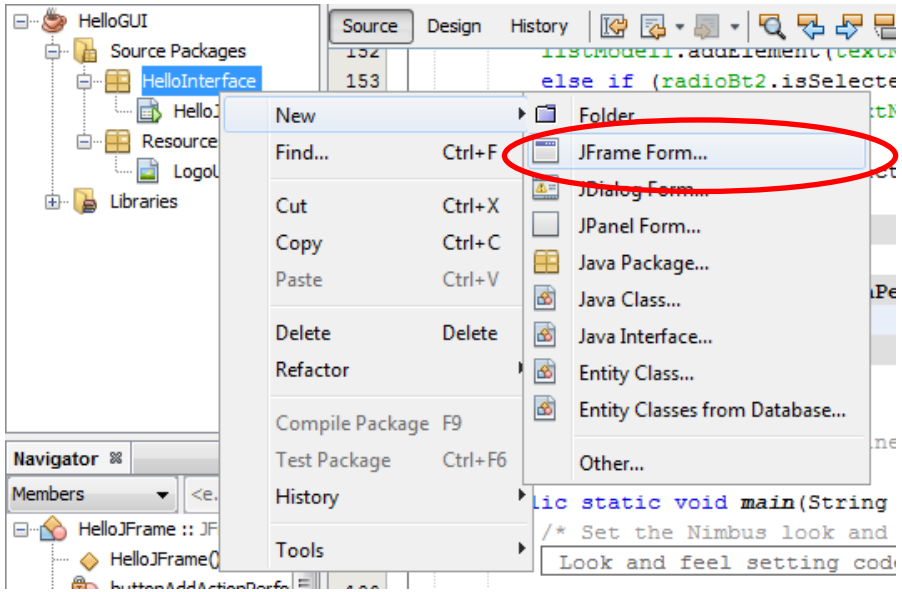
- Componente que permite a criação de itens para a barra de menu.

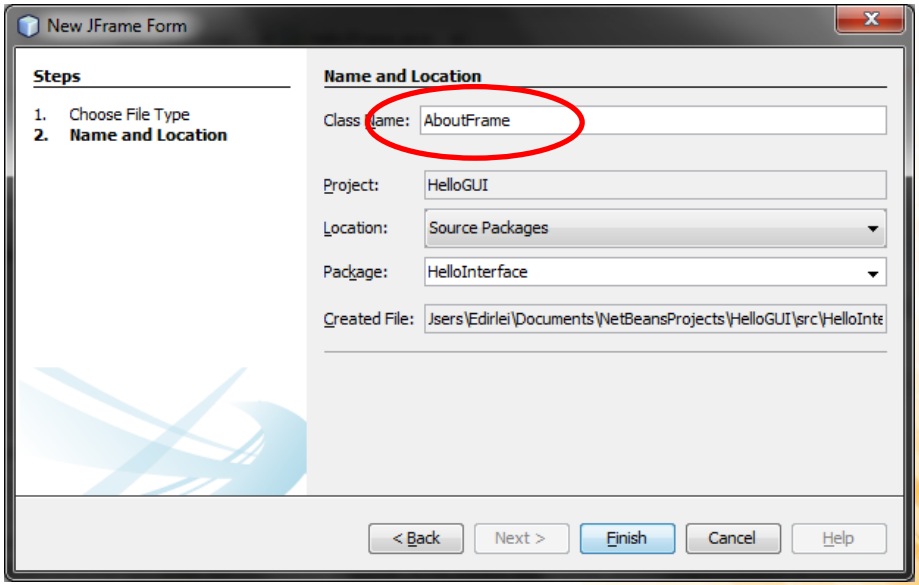


Abrindo uma Nova Janela

- **Exemplo 5 –** Criando e abrindo uma nova janela.

a) Crie um novo JFrame;

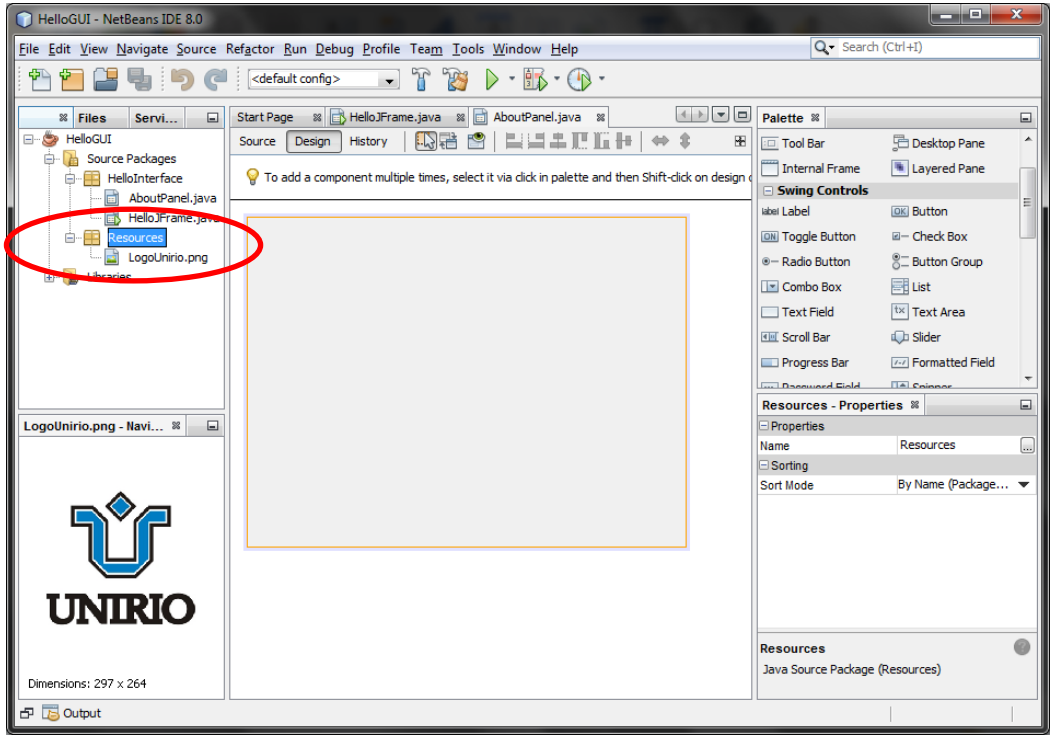




Abrindo uma Nova Janela

- **Exemplo 5 –** Criando e abrindo uma nova janela.

b) Crie um novo Package e arraste o arquivo de uma imagem para o projeto;



Abrindo uma Nova Janela

- **Exemplo 5 –** Criando e abrindo uma nova janela.

c) Adicione um novo label ao formulário, colocando a imagem previamente selecionada como icone. Adicione também um botão para fechar a janela.

Abrindo uma Nova Janela

- **Exemplo 5 –** Criando e abrindo uma nova janela.

d) Na janela principal, crie um novo evento para criar e abrir a nova janela quando o usuário clicar no item “About” do menu “Help”:

```
private void menuAboutActionPerformed(java.awt.event.ActionEvent evt)
{

  AboutJFrame aboutJFrame = new AboutJFrame();

  aboutJFrame.setVisible(true);

}
```

Abrindo uma Nova Janela

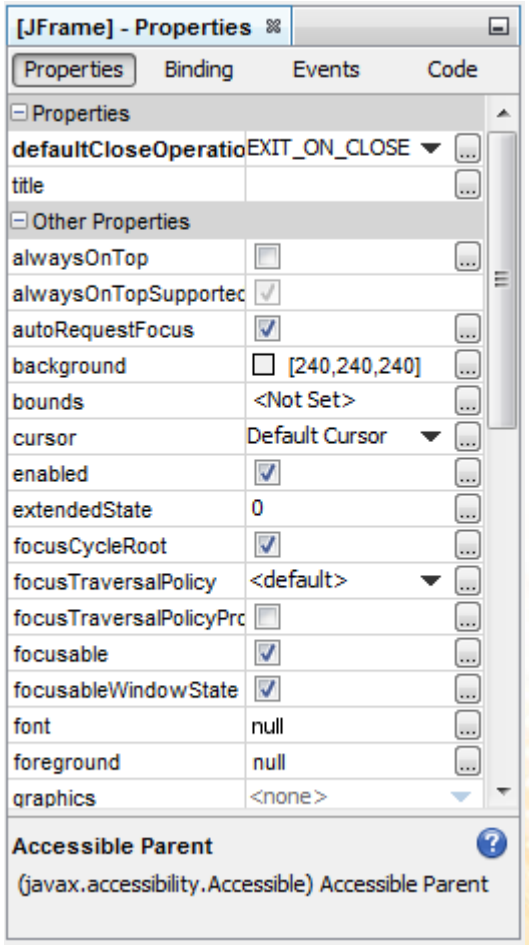
- **Exemplo 5 –** Criando e abrindo uma nova janela.

e) Crie um evento “actionPerformed” para o botão fechar, ocultando e destruindo a janela quando o usuário clicar no botão:

```
private void buttonFecharActionPerformed(java.awt.event.ActionEvent evt)
{

  this.setVisible(false);

  this.dispose();

}
```

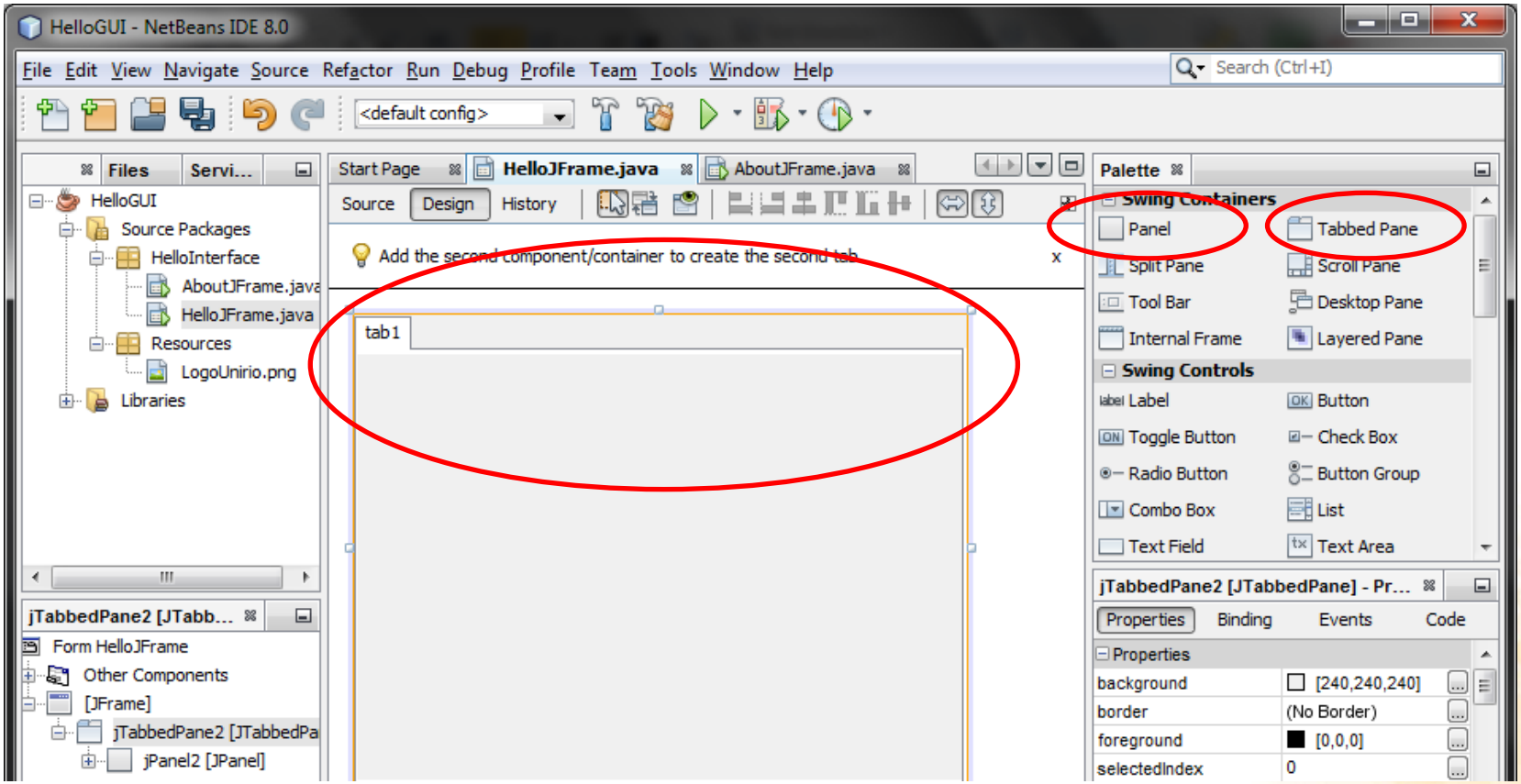
Componentes Básicos – Frame

- Principais Propriedades (JFrame):
  - defaultCloseOperation;
  - title;
  - background;
  - awaysOnTop;
  - iconImage;
  - resizable;
  - undecorated;
  - type;

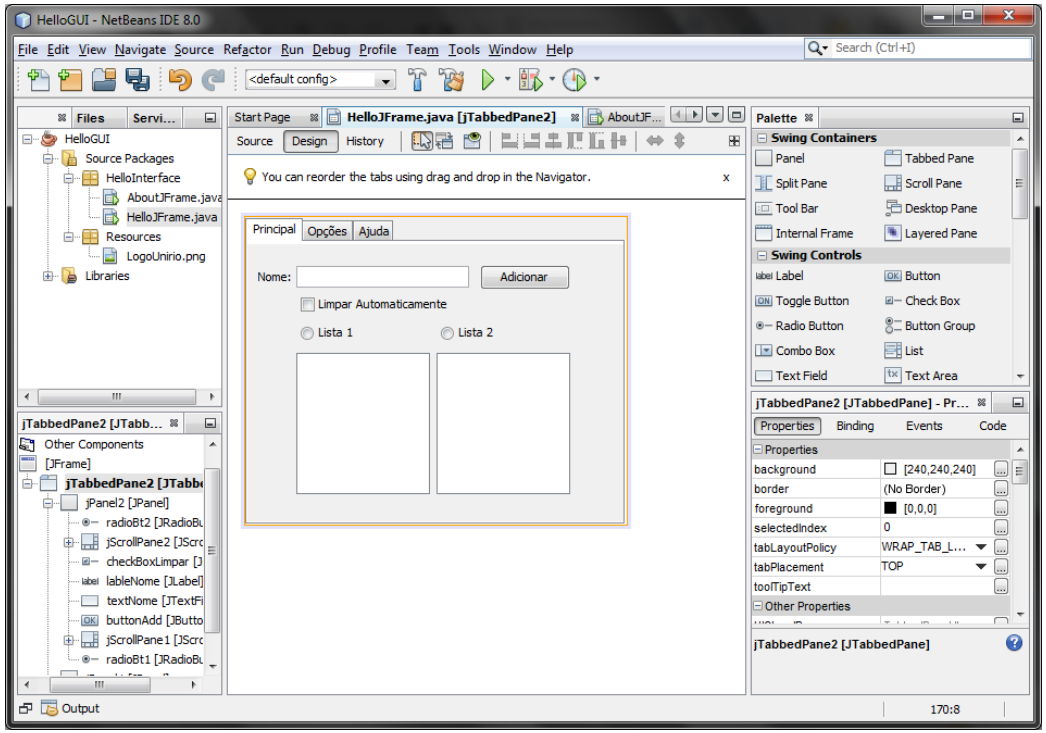


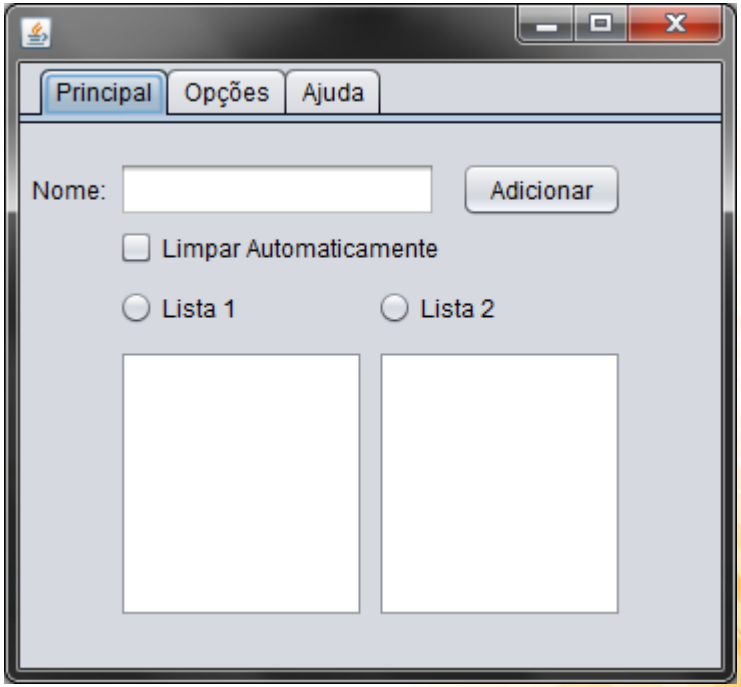
Componentes Básicos – Tabbed Pane

- Componente que permite a criação de telas com abas.



Componentes Básicos – Tabbed Pane





Exercícios

Lista de Exercícios 03 – GUI e Swing

http://uniriodb2.uniriotec.br