



INSTITUTO FEDERAL
DE EDUCAÇÃO, CIÊNCIA E TECNOLOGIA
Bahia

Linguagem de Programação II

Introdução às Interfaces Gráficas de
Usuário (GUI) em Java: Pacote Swing
(JFrame, JPanel, JLabel, JTextField,
JPasswordField e JButton)



Roteiro

- Introdução às Interfaces Gráficas de Usuário (GUI) em Java (Swing):
 - Introdução;
 - Componentes;
 - JFrame (Janela);
 - Netbeans e Swing
 - JButton (Botão)
 - JLabel (Rótulo ou Etiquetas);
 - JTextField (Campo de Texto);
 - JPasswordField (Campo de Senha);



Java Swing - Introdução

- Na **Interface Gráfica** é necessário definir quais **componentes (objetos)** serão utilizados e a **disposição** que **eles** terão na **janela (objeto)**.

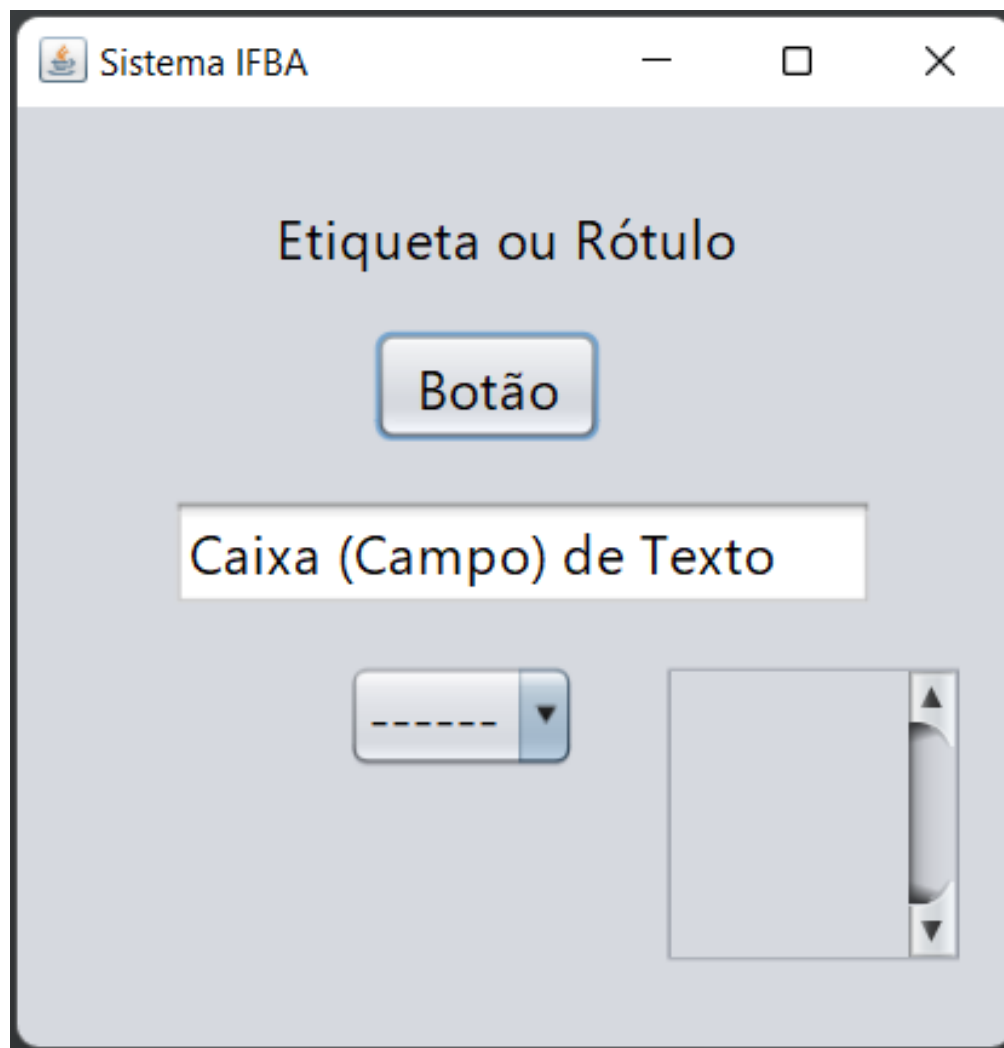


Java Swing – Componentes

- Um **componente** da **Interface Gráfica** é um **objeto visual (Java)** que possibilita realizar a **interação com o programa** por meio do **mouse** e do **teclado**.
- Ex.: **etiquetas, botões, caixas de texto, painéis de rolagem, menus, objetos de múltipla escolha**, entre outros.



Java Swing – Componentes

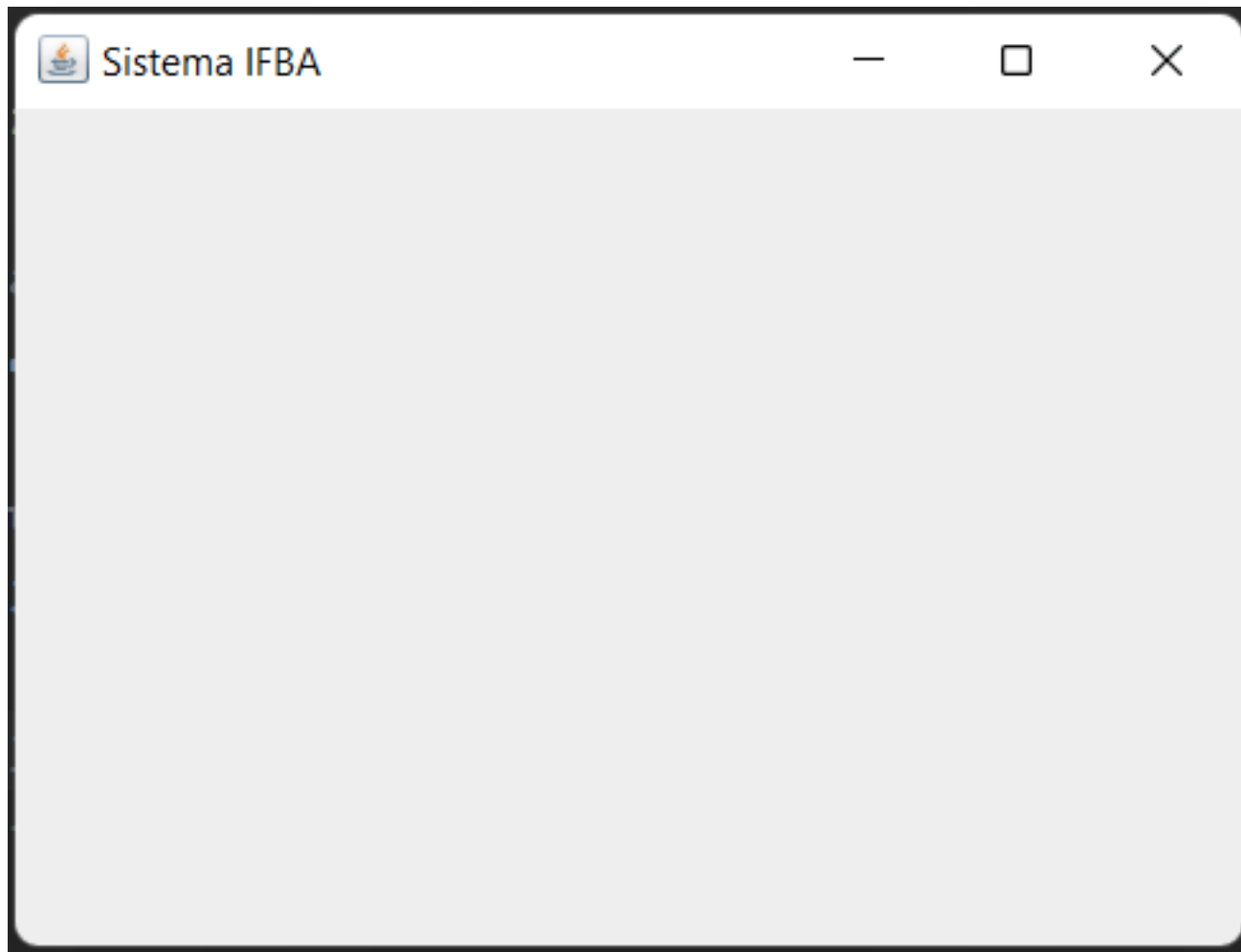


Java Swing – JFrame

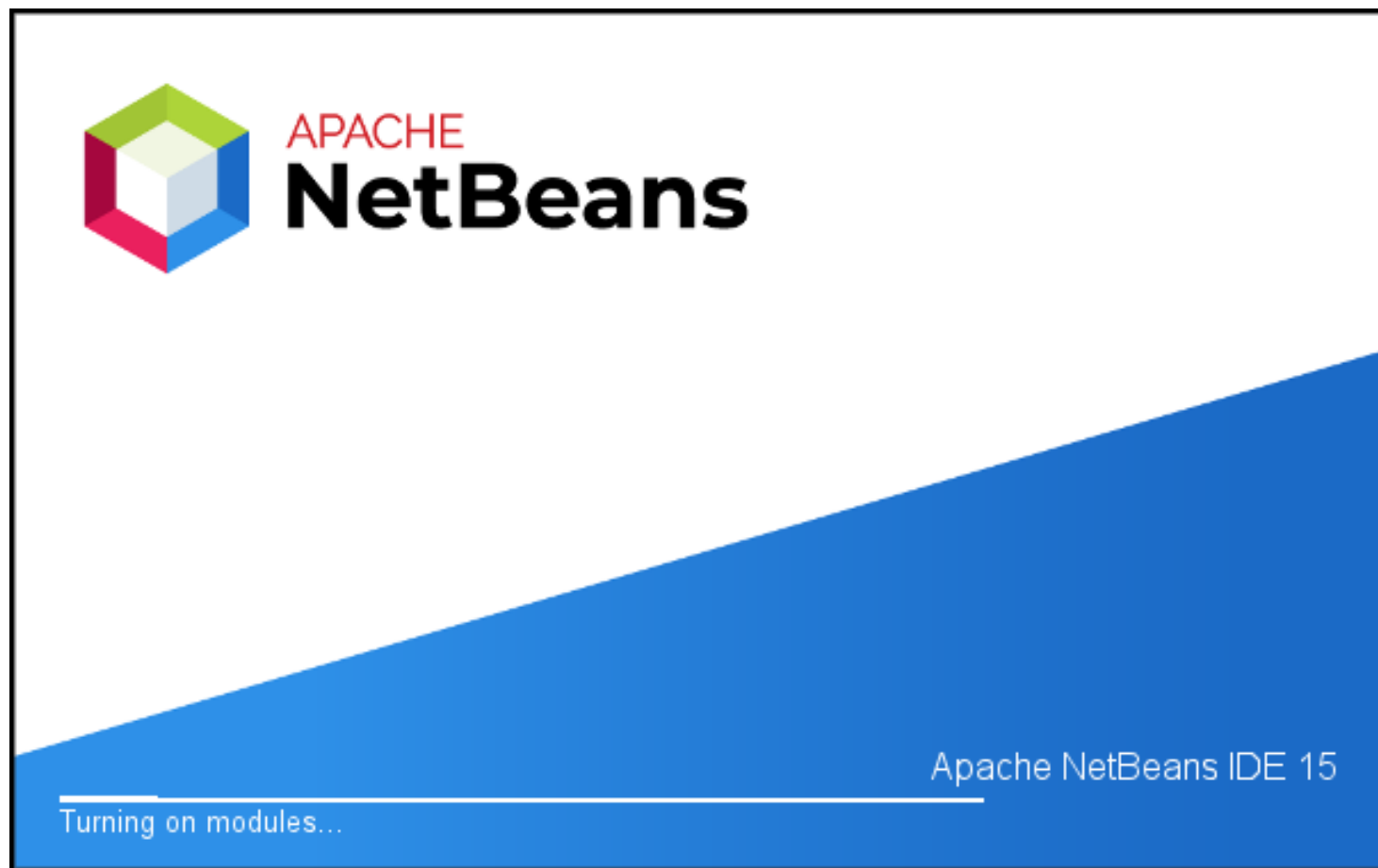
- Para criar **Janelas** utilizaremos a classe **JFrame** disponível no pacote **swing**, a qual gera uma **janela** com **barra de título**, **bordas** e eventualmente outros **componentes visuais**.

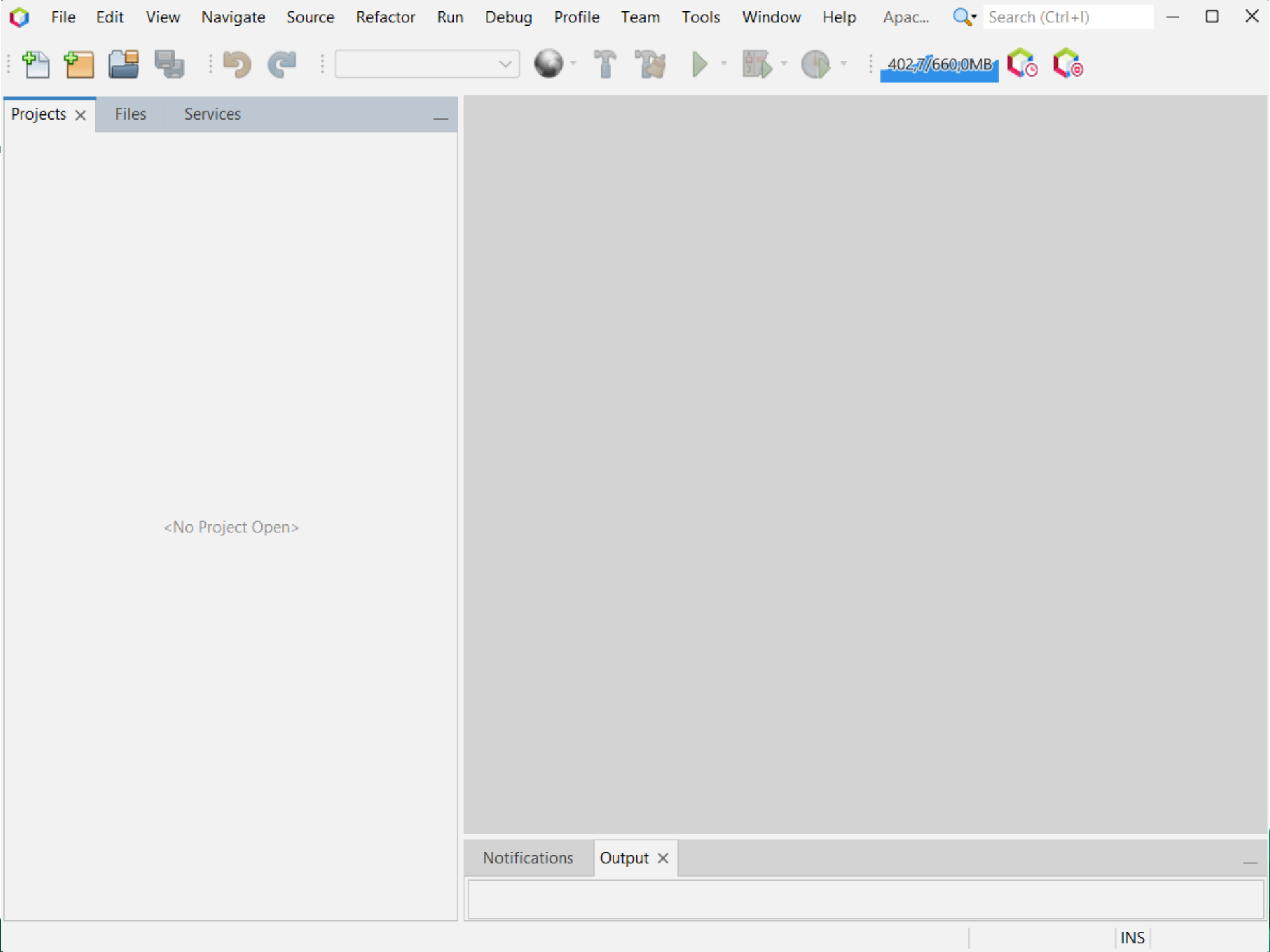


Java Swing – JFrame

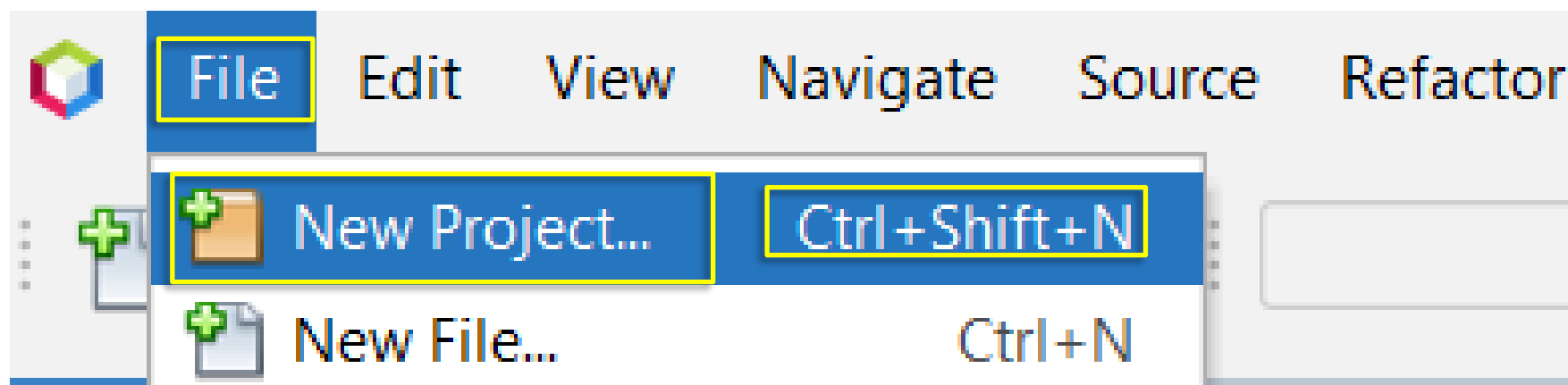


Netbeans: IDE





Netbeans: Novo Projeto



Netbeans: Novo Projeto

Choose Project



Filter:

Categories:

- Java with Maven
- Java with Gradle
- Java with Ant
- JavaFX
- Java Web

Projects:

- Java Application
- Java Class Library
- Java Project with Existing Sources
- Java Modular Project
- Java Free-Form Project



Netbeans: Novo Projeto

Name and Location

Project Name:

Project Location:

Project Folder:

☐ Use Dedicated Folder for Storing Libraries

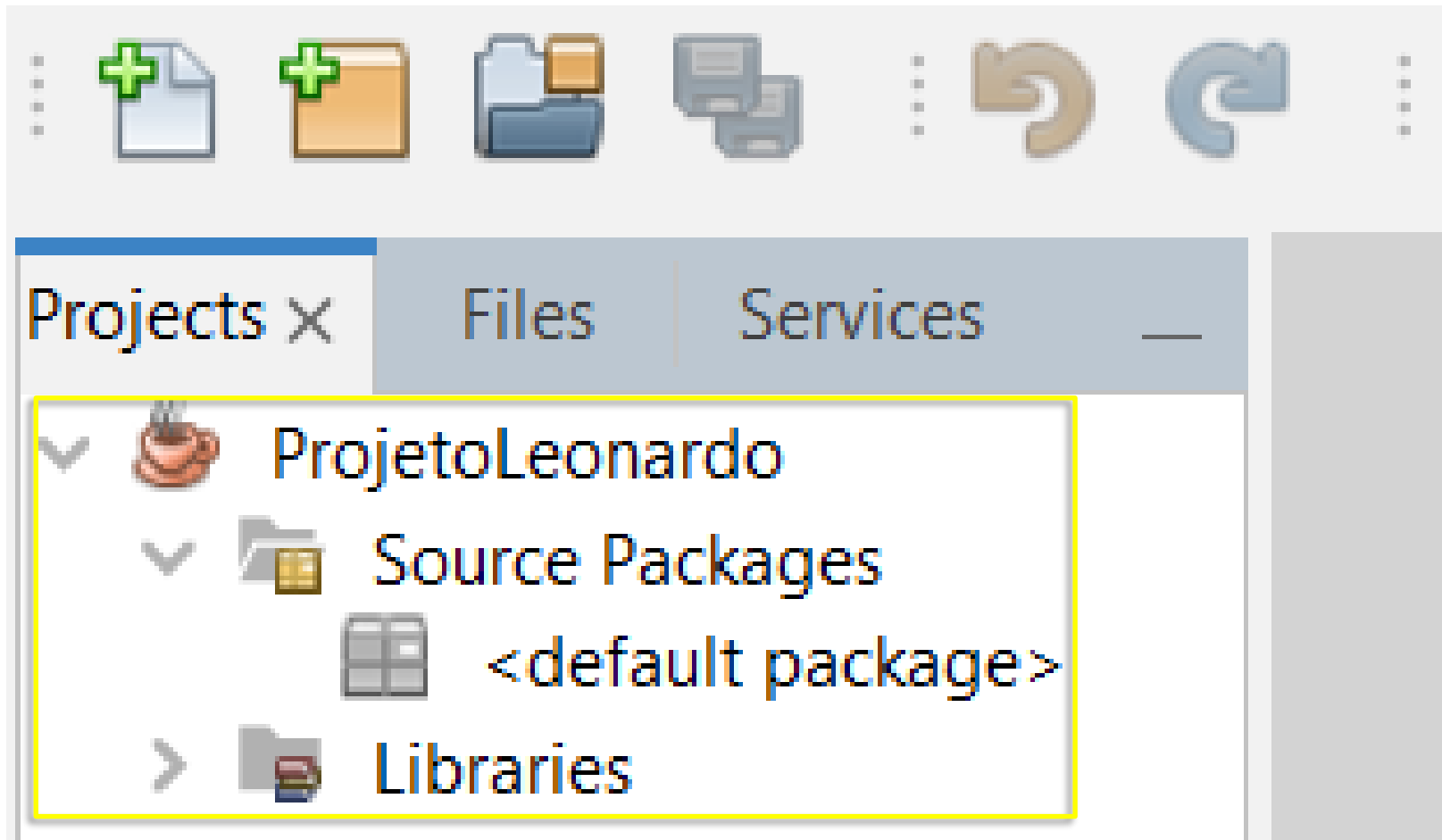
Libraries Folder:

Different users and projects can share the same compilation libraries (see Help for details).

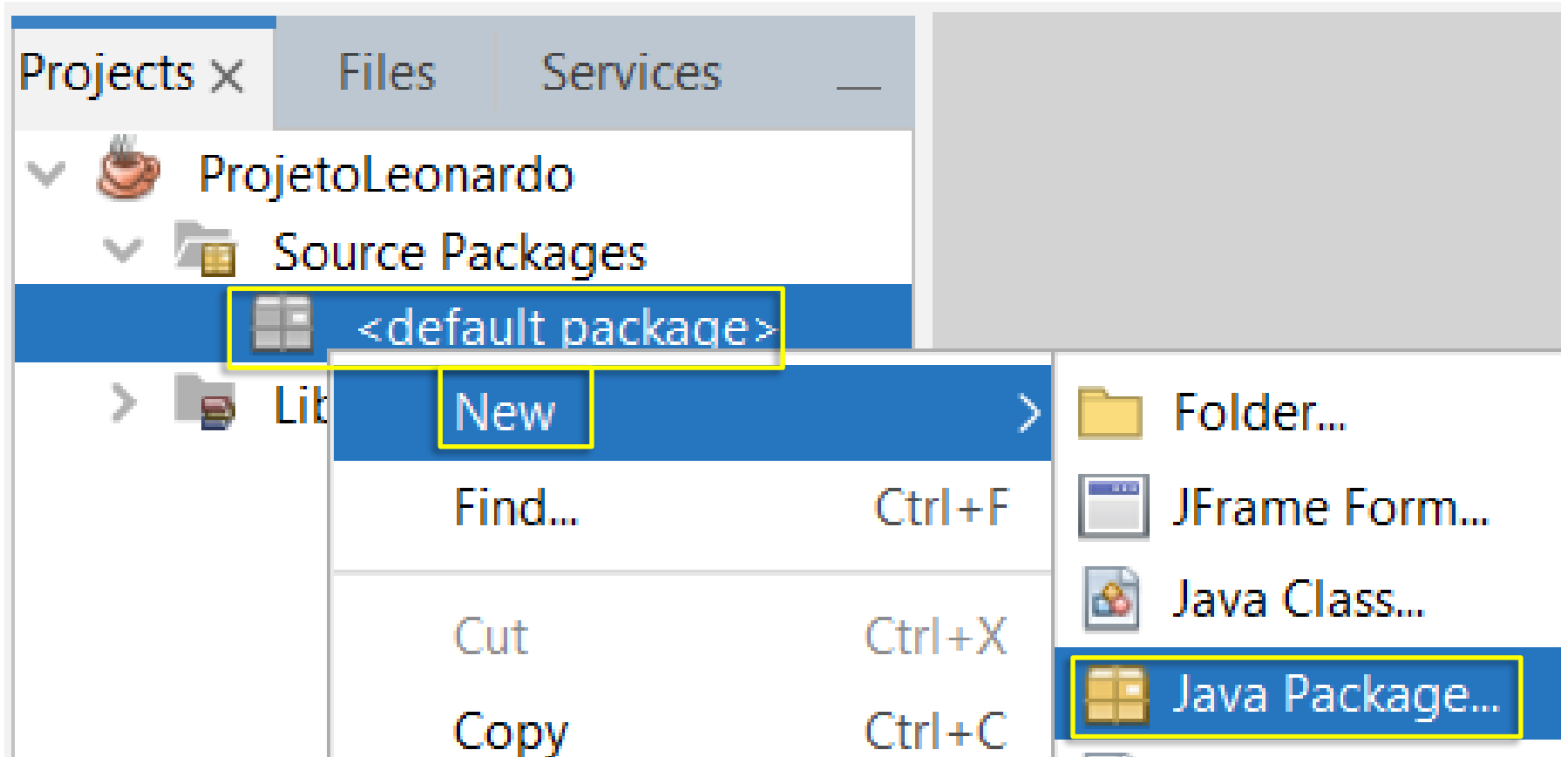
☐ Create Main Class



Netbeans: Novo Projeto



Netbeans: Novo Pacote



Netbeans: Novo Pacote

Name and Location

Package Name:

Principal

Project:

ProjetoLeonardo

Location:

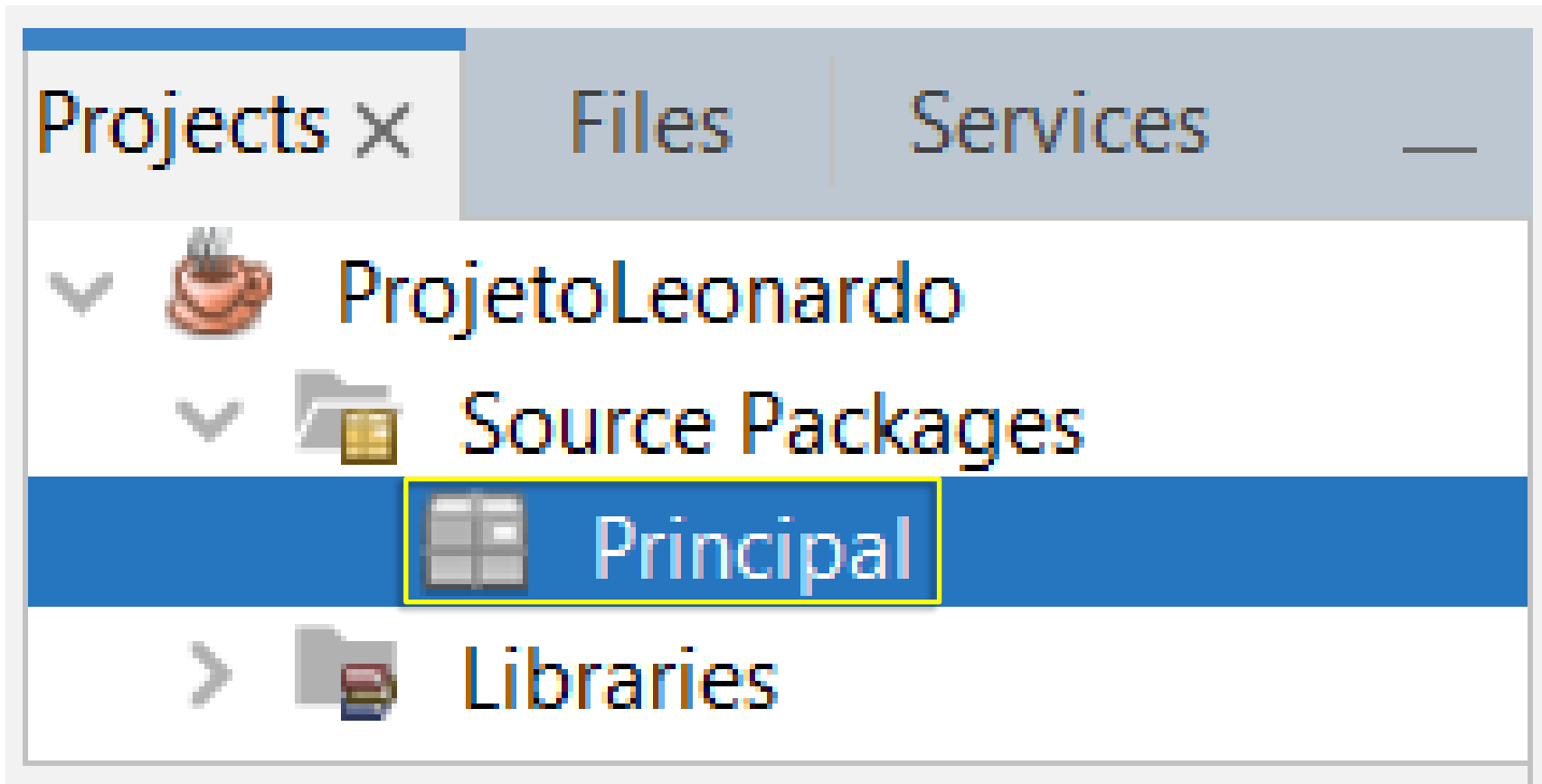
Source Packages

Created Folder:

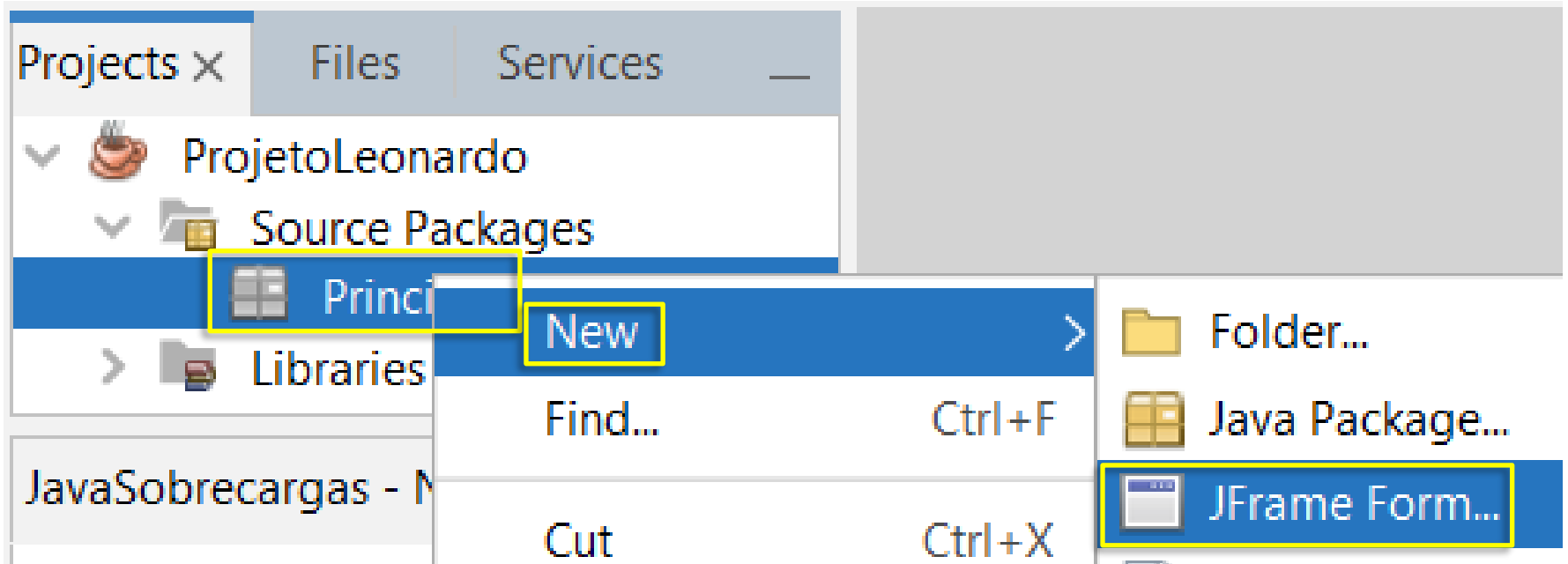
.:\\Users\\leomi\\Documents\\NetBeansProjects\\ProjetoLeonardo\\src\\Principal



Netbeans: Novo Pacote



Netbeans: Nova Janela (JFrame)



Netbeans: Nova Janela (Jframe)

Name and Location

Class Name:

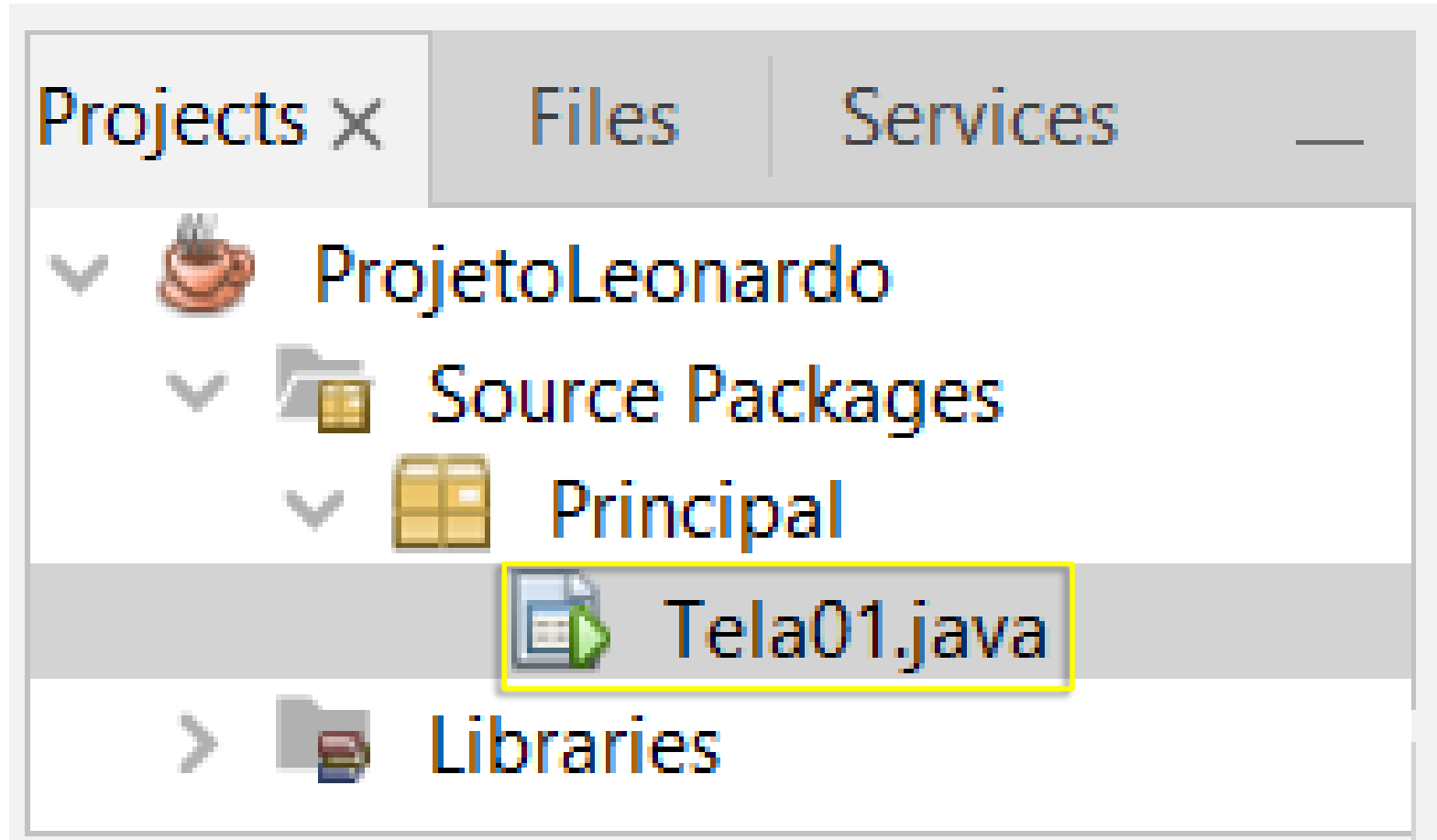
Project:

Location:

Package:



Netbeans: Nova Janela (JFrame)



Projects x Files Services


- ProjetoLeonardo
 - Source Packages
 - Principal
 - Tela01.java
 - Libraries

[JFrame] - Navigator x

- Form Tela01
 - Other Components
 - [JFrame]

Tela01.java x

Source Design History



Palette x

Swing Containers

- Panel
- Tabbed Pane
- Split Pane
- Scroll Pane
- Tool Bar
- Desktop Pane
- Internal Frame

[JFrame] - Properties x

Properties Events

Code

Properties

defaultCloseOperation	EXIT_ON...	...
title		...

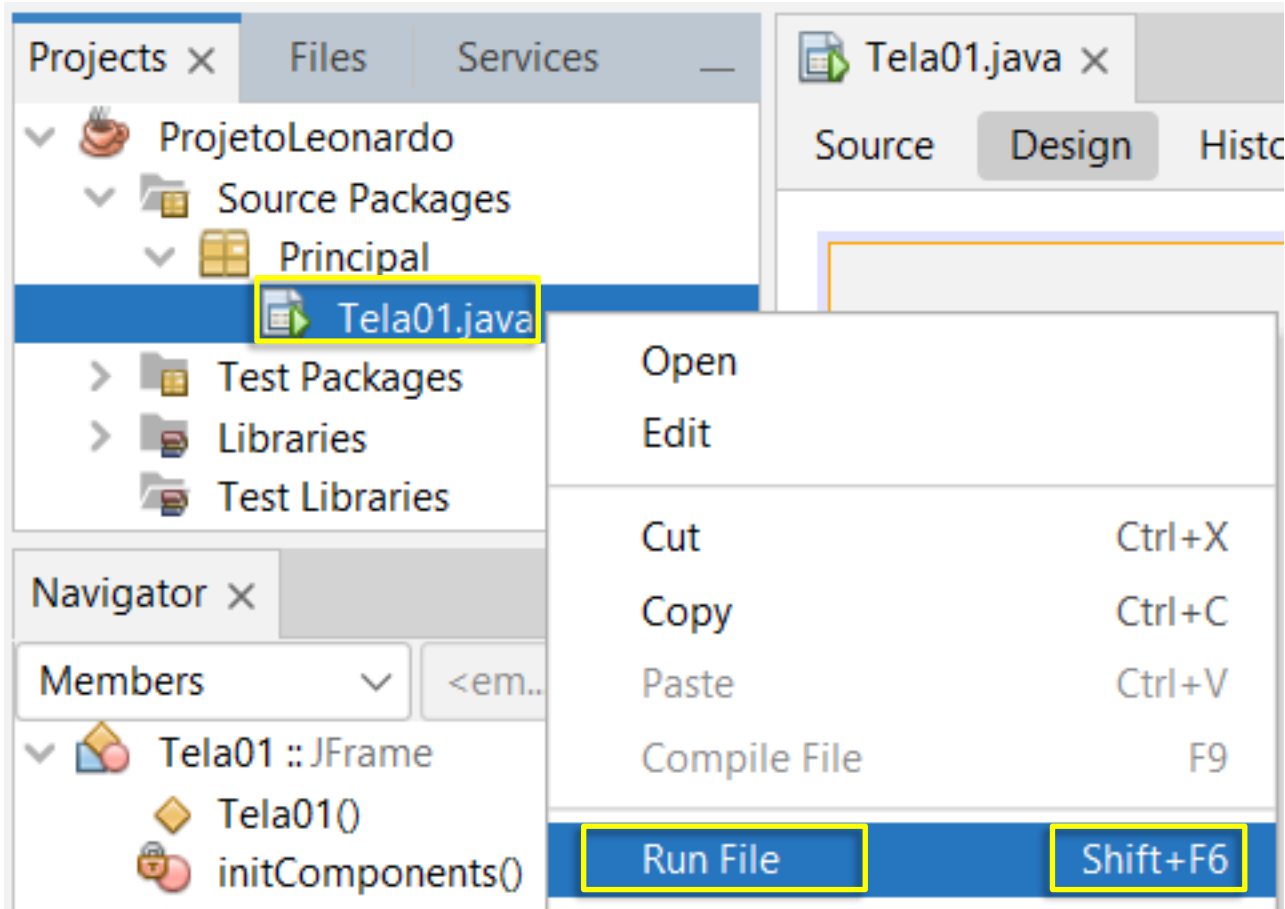
Other Properties

alwaysOnTop	<input type="checkbox"/>	...
alwaysOnTopSupport	<input checked="" type="checkbox"/>	...

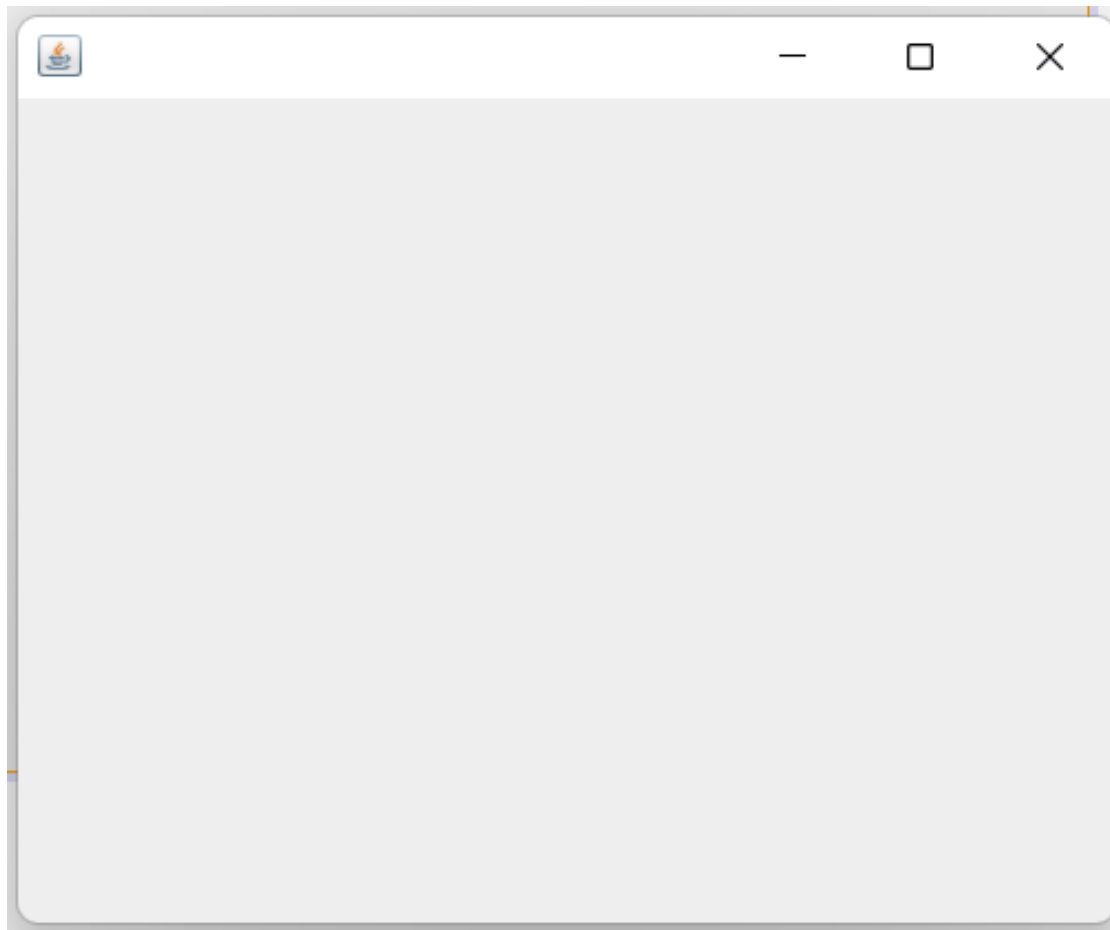
[JFrame] ?

Notifications Output x

Netbeans: Executando Janela (JFrame)



Netbeans: Executando Janela (JFrame)





```
1 package Principal;
```

```
2  
3 public class Tela01 extends javax.swing.JFrame {
```

```
4  
5     public Tela01() {  
6         initComponents();  
7     }  
8
```

```
9 + Generated Code
```

```
27  
28     public static void main(String args[]) {  
29         new Tela01().setVisible(true);  
30     }  
31
```

```
32 // Variables declaration - do not modify  
33 // End of variables declaration  
34
```

```
}
```




```
1 package Principal;
```

Pacote

```
2  
3 public class Tela01 extends javax.swing.JFrame {
```

Construtor

Classe

Herança

```
4 public Tela01() {  
5     initComponents();  
6 }  
7
```

Chamada de método que
inicializa componentes

Generated Code

Inicialização e configuração de
componentes

Método estático "main"

```
27  
28 public static void main(String args[]) {  
29     new Tela01().setVisible(true);  
30 }  
31
```

Declaração
de variáveis

Cria objeto (componente) Tela01 e o torna visível

```
32 // Variables declaration - do not modify  
33 // End of variables declaration  
34 }
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
```

```
    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
    javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
```

```
    getContentPane().setLayout(layout);
```

```
    layout.setHorizontalGroup(
```

```
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
            .addGap(0, 400, Short.MAX_VALUE)
```

```
    );
```

```
    layout.setVerticalGroup(
```

```
        layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
            .addGap(0, 300, Short.MAX_VALUE)
```

```
    );
```

```
    pack();
```

```
}// </editor-fold>
```

```
// <editor-fold defaultstate="collapsed" desc="Generated Code">
```

```
private void initComponents() {
```

Método

Comportamento do ícone (botão) de fechar o JFrame

```
setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
```

```
javax.swing.GroupLayout layout = new javax.swing.GroupLayout(getContentPane());
```

```
getContentPane().setLayout(layout);
```

```
layout.setHorizontalGroup(
```

```
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
        .addGap(0, 400, Short.MAX_VALUE)
```

```
);
```

```
layout.setVerticalGroup(
```

```
    layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
```

```
        .addGap(0, 300, Short.MAX_VALUE)
```

```
);
```

```
pack();
```

Layout Manager do JFrame ajusta o tamanho da tela (Frame) com base em seus componentes internos

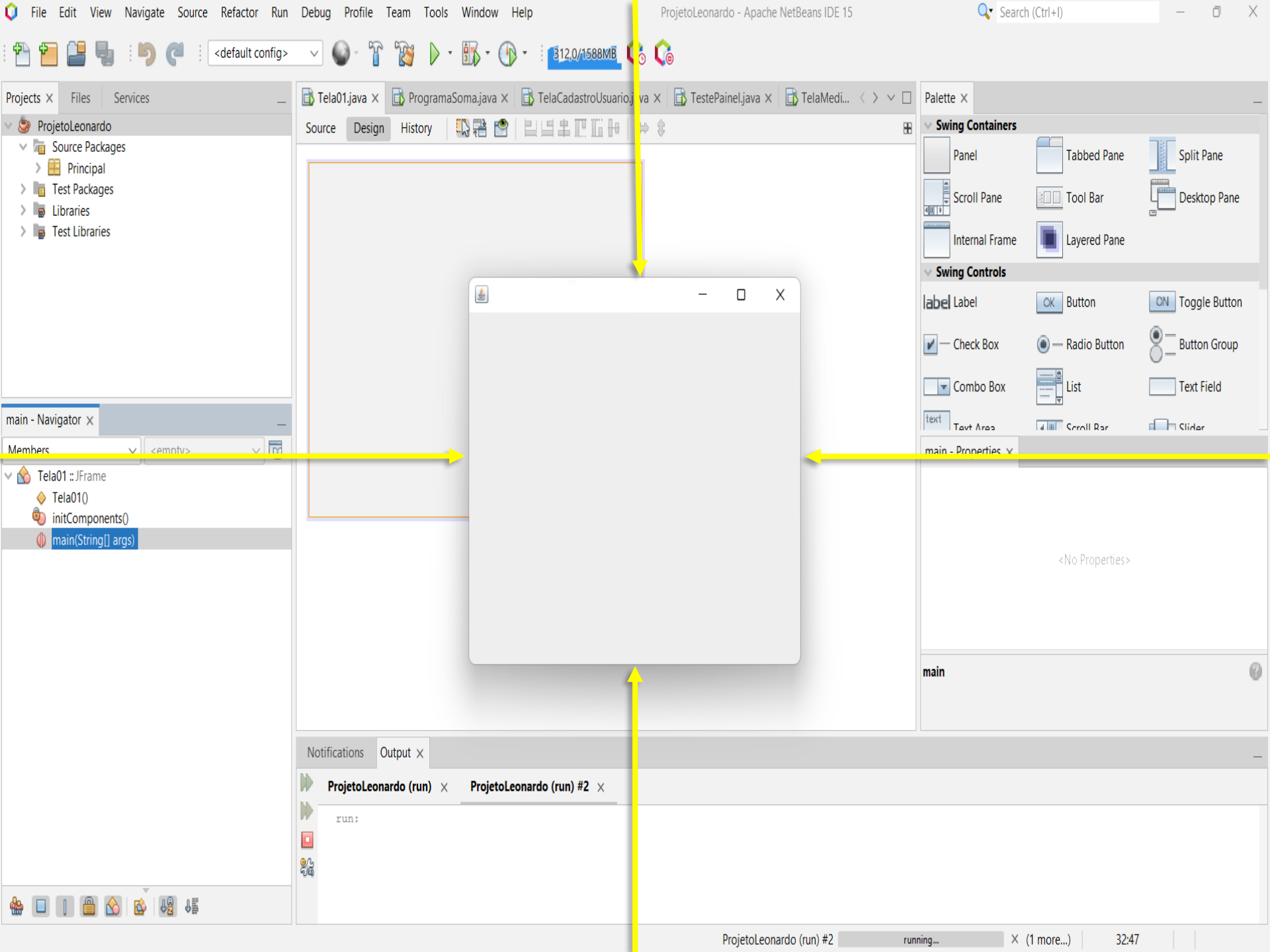
Configurações de Layout do JFrame

```
} // </editor-fold>
```

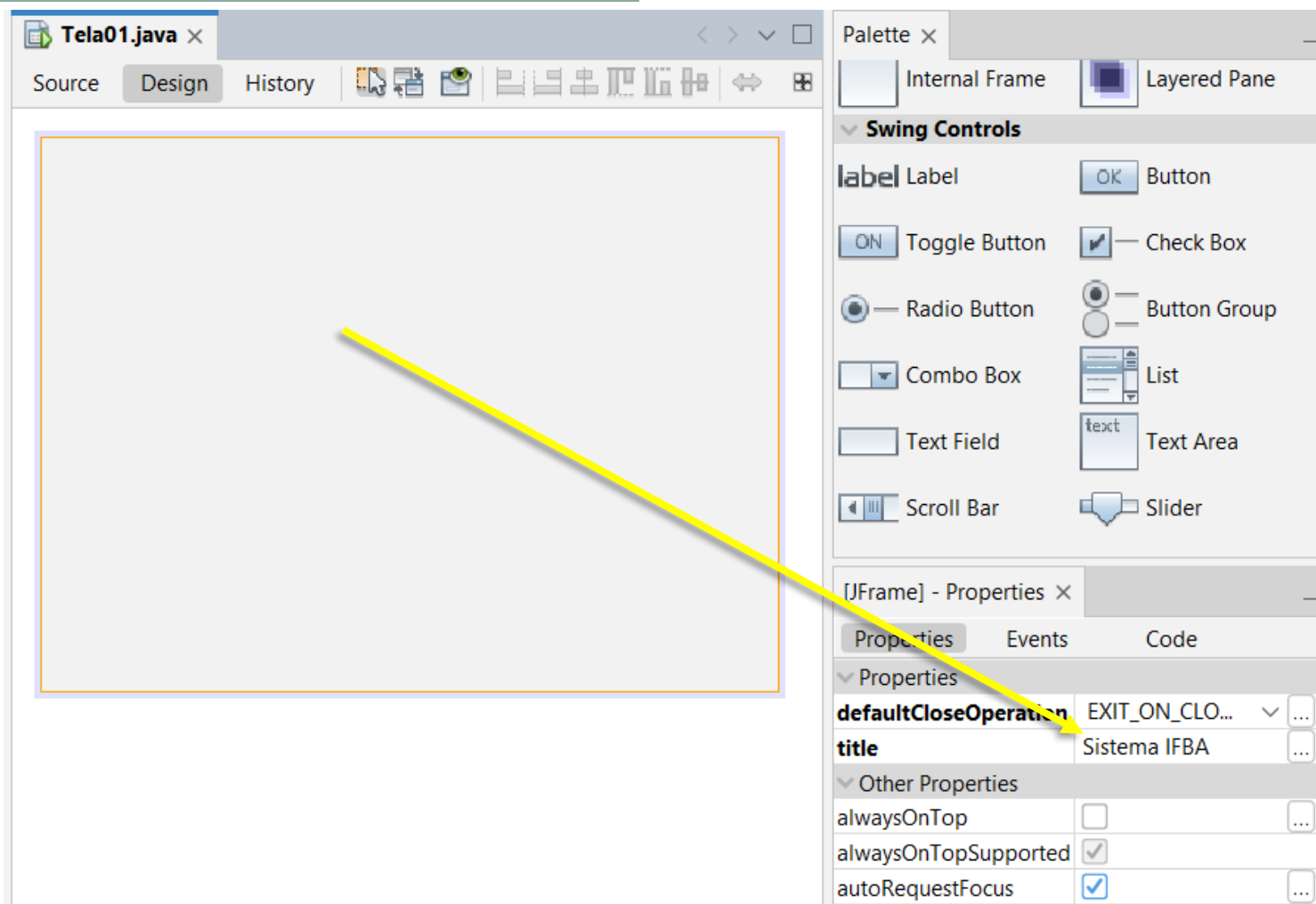
Netbeans: Centralizar JFrame

```
public Tela01 () {  
    initComponents () ;  
    this.setLocationRelativeTo (null) ;  
}
```

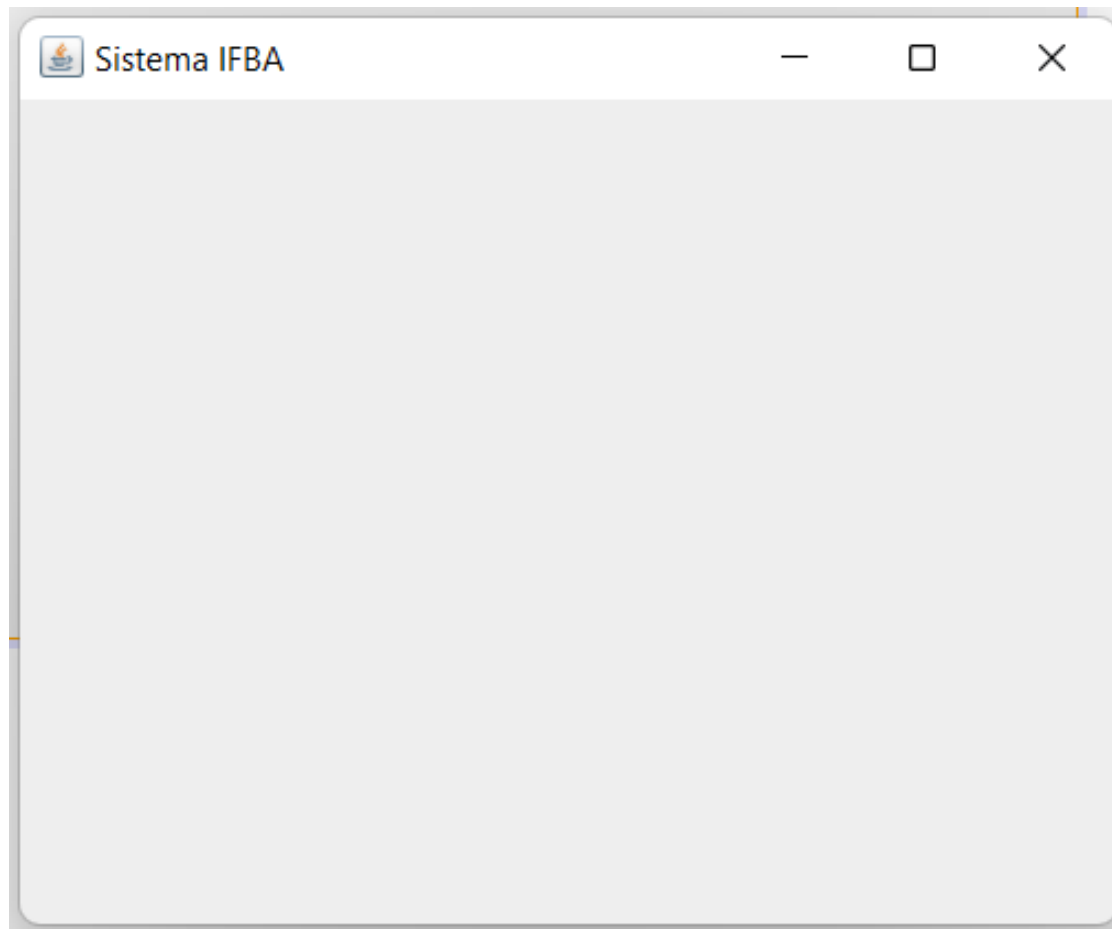




Netbeans: Título da Janela (JFrame)



Netbeans: Título da Janela (JFrame)



Netbeans: Desabilitar Redimensionar

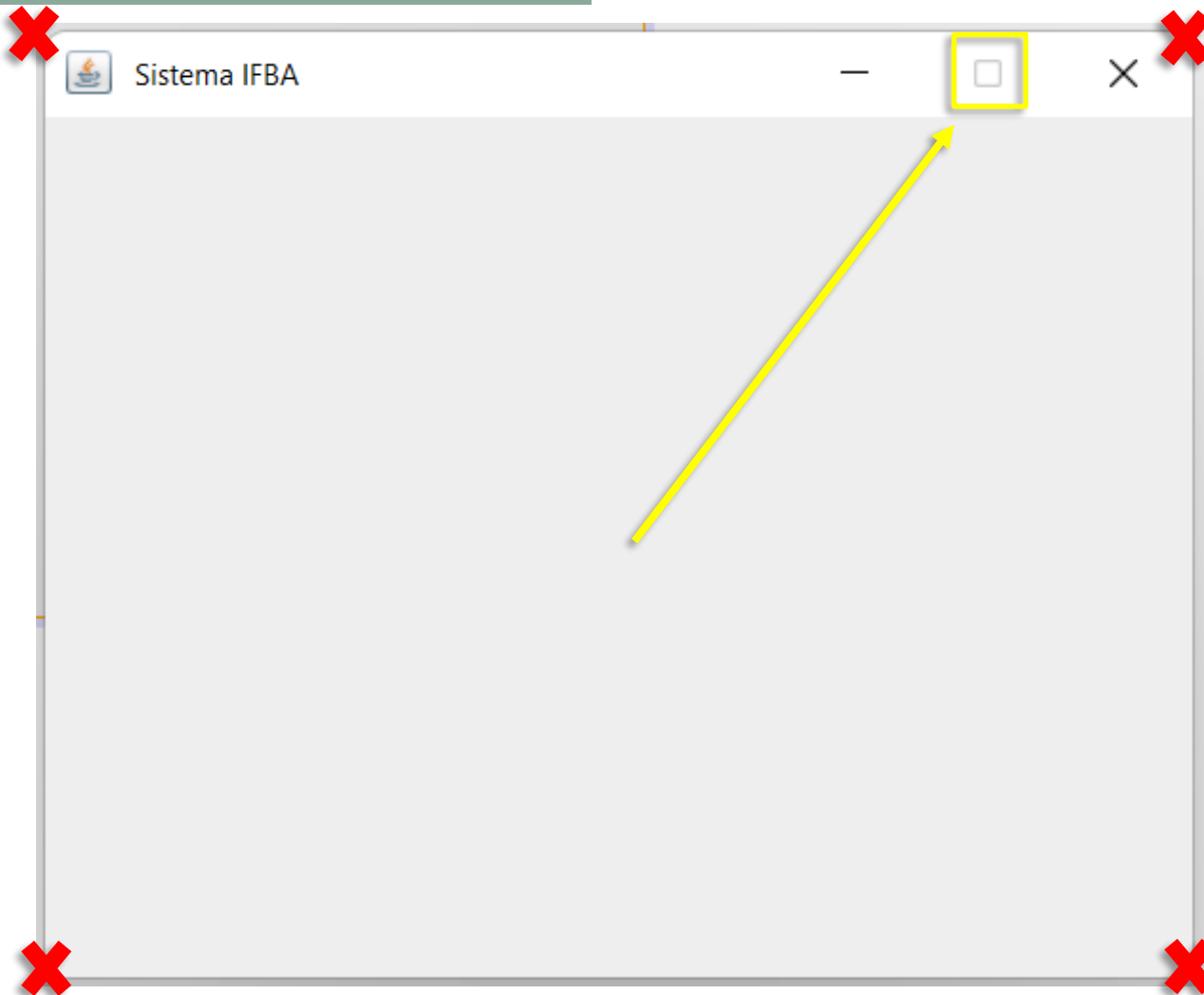
[JFrame] - Properties X

Properties	Events	Code
preferredSize		[395, 300]
resizable	<input type="checkbox"/>	
shape		<none>
size		<Not Set>
state		0
type		NORMAL

False



Netbeans: Desabilitar Redimensionar

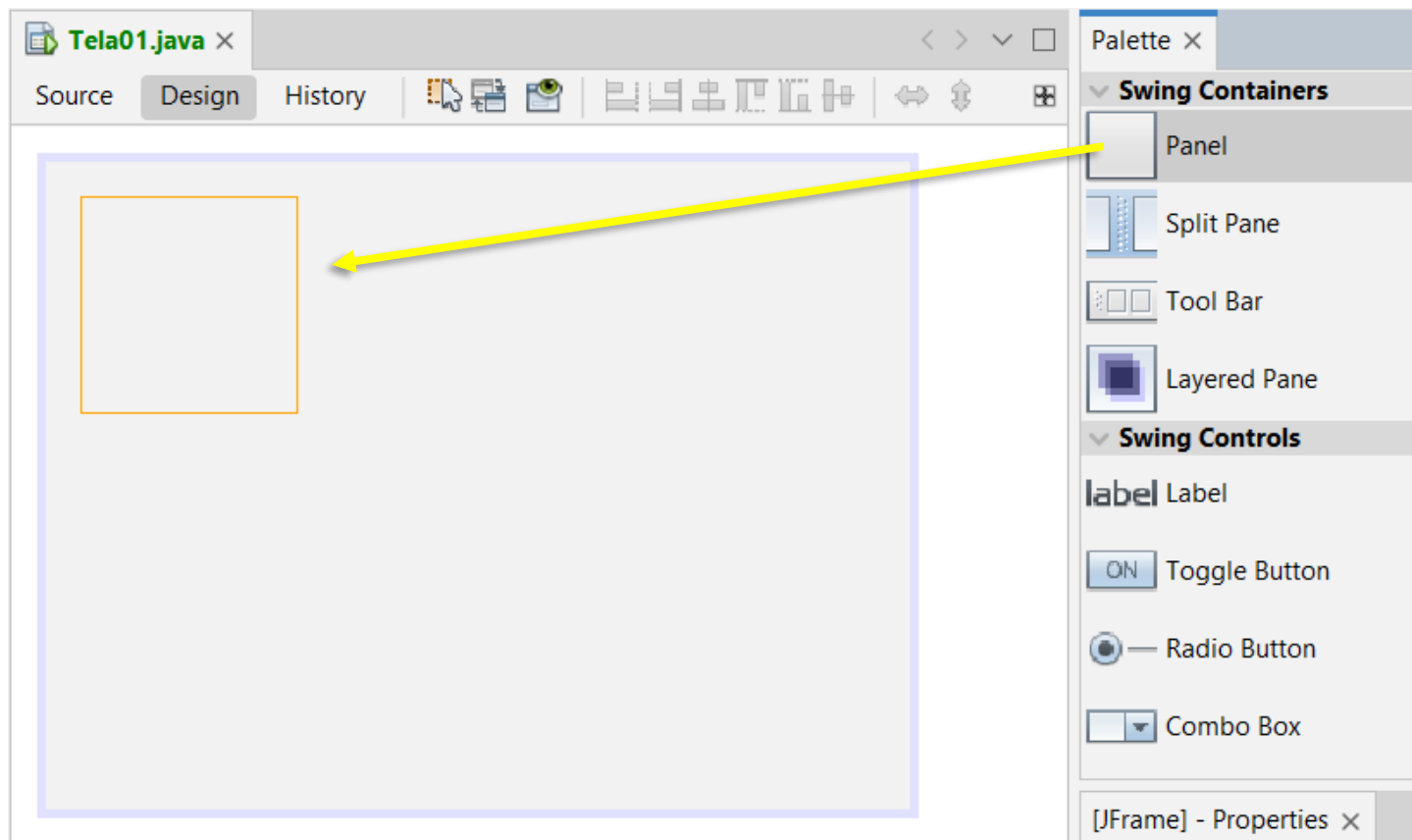


Java Swing – JPanel

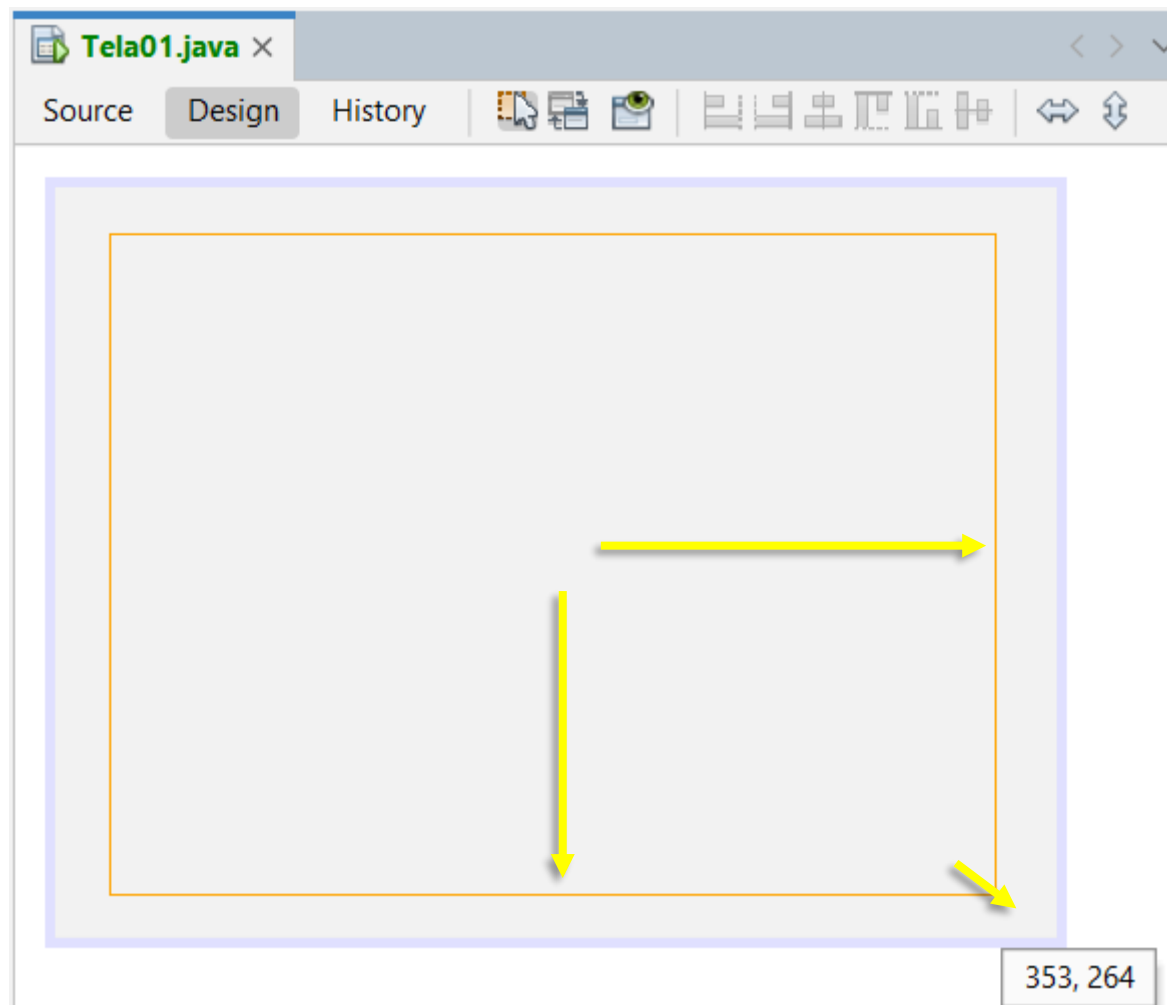
- O **JPanel** é um **contêiner flexível** no Java Swing, utilizado para **agrupar e organizar componentes** em interfaces gráficas.
- Ele **oferece opções de personalização**, como **layout e cor de fundo**, permitindo criar **interfaces atraentes e adaptáveis**.



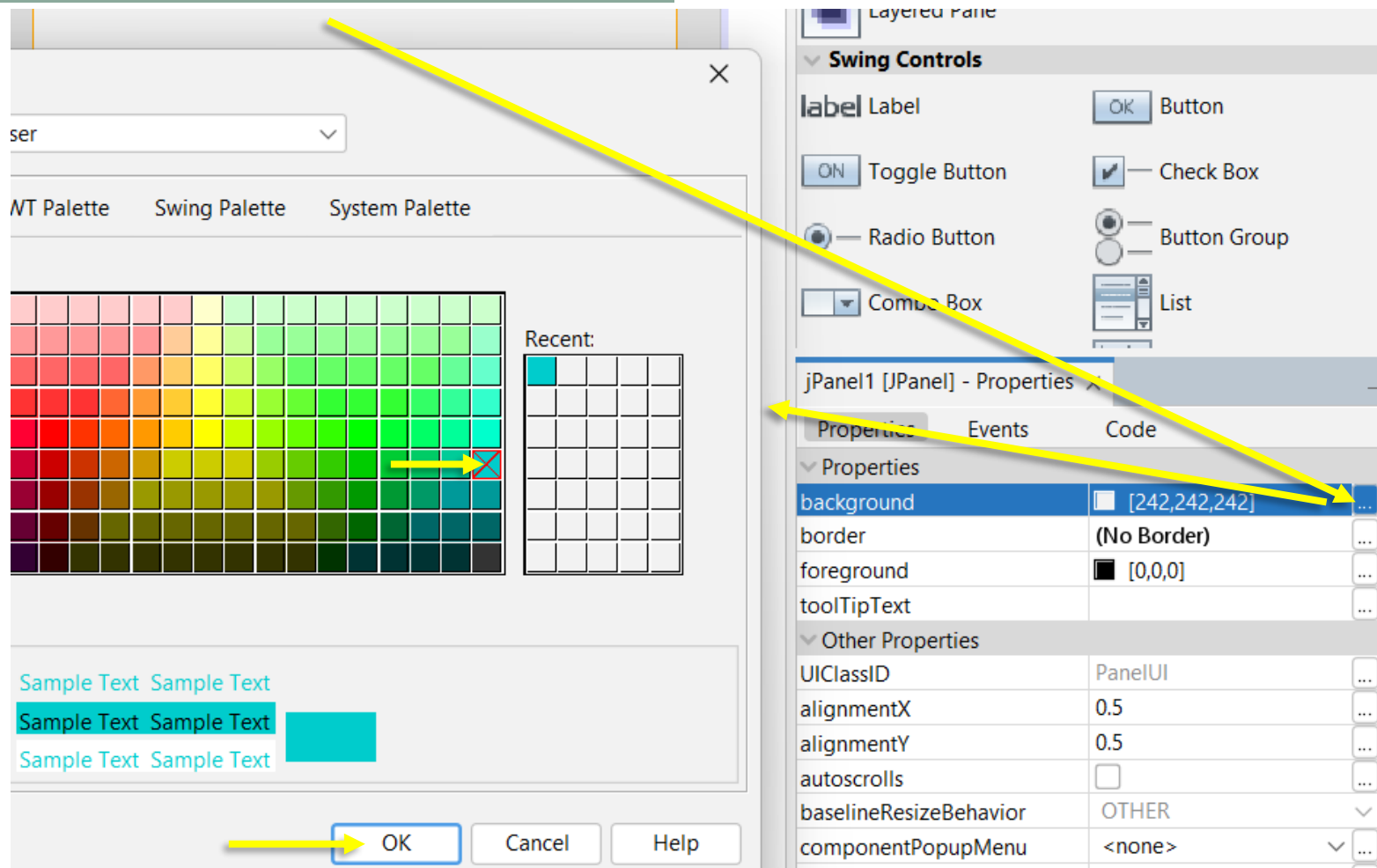
Java Swing – JPanel - Adicionar



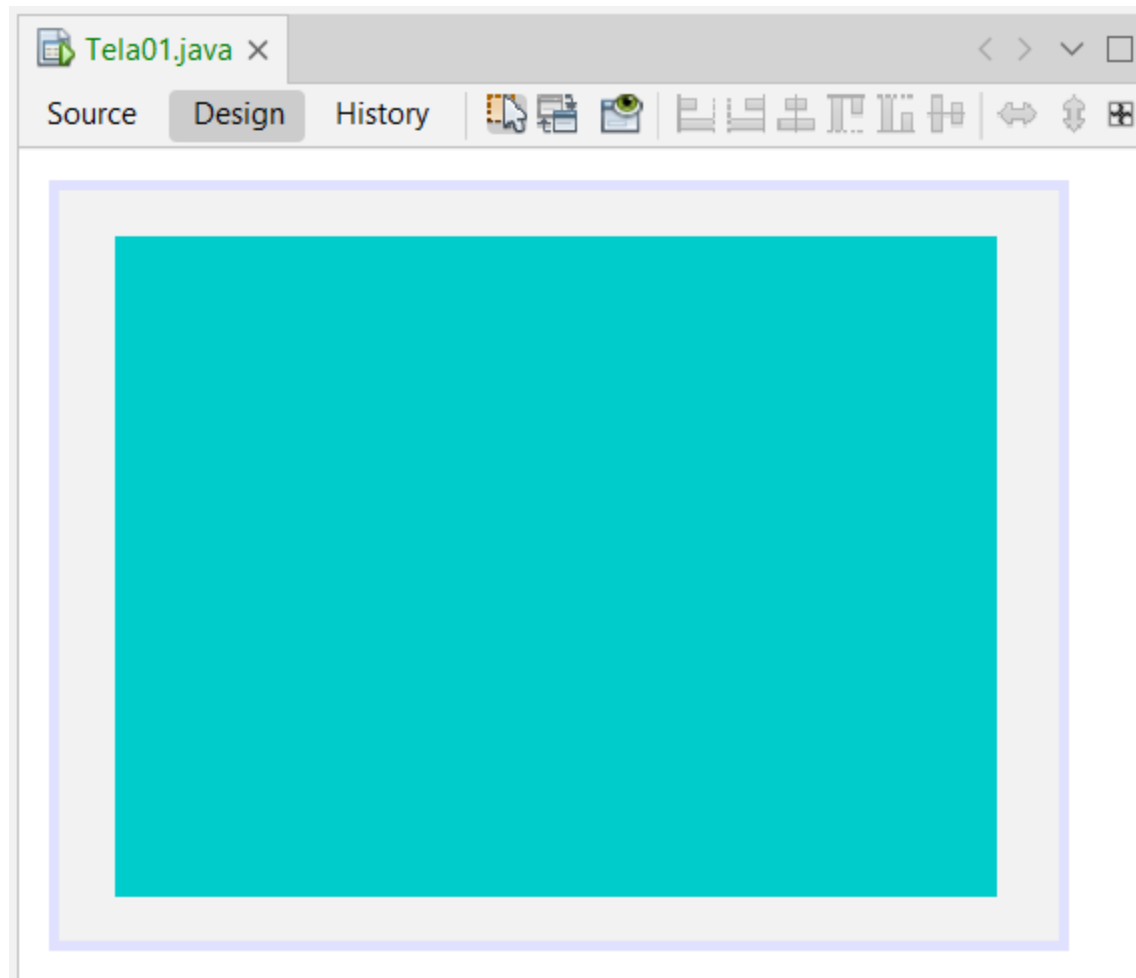
Java Swing – JPanel - Redimensionar



Java Swing – JPanel – Mudar Cor Fundo



Java Swing – JPanel – Mudar Cor Fundo

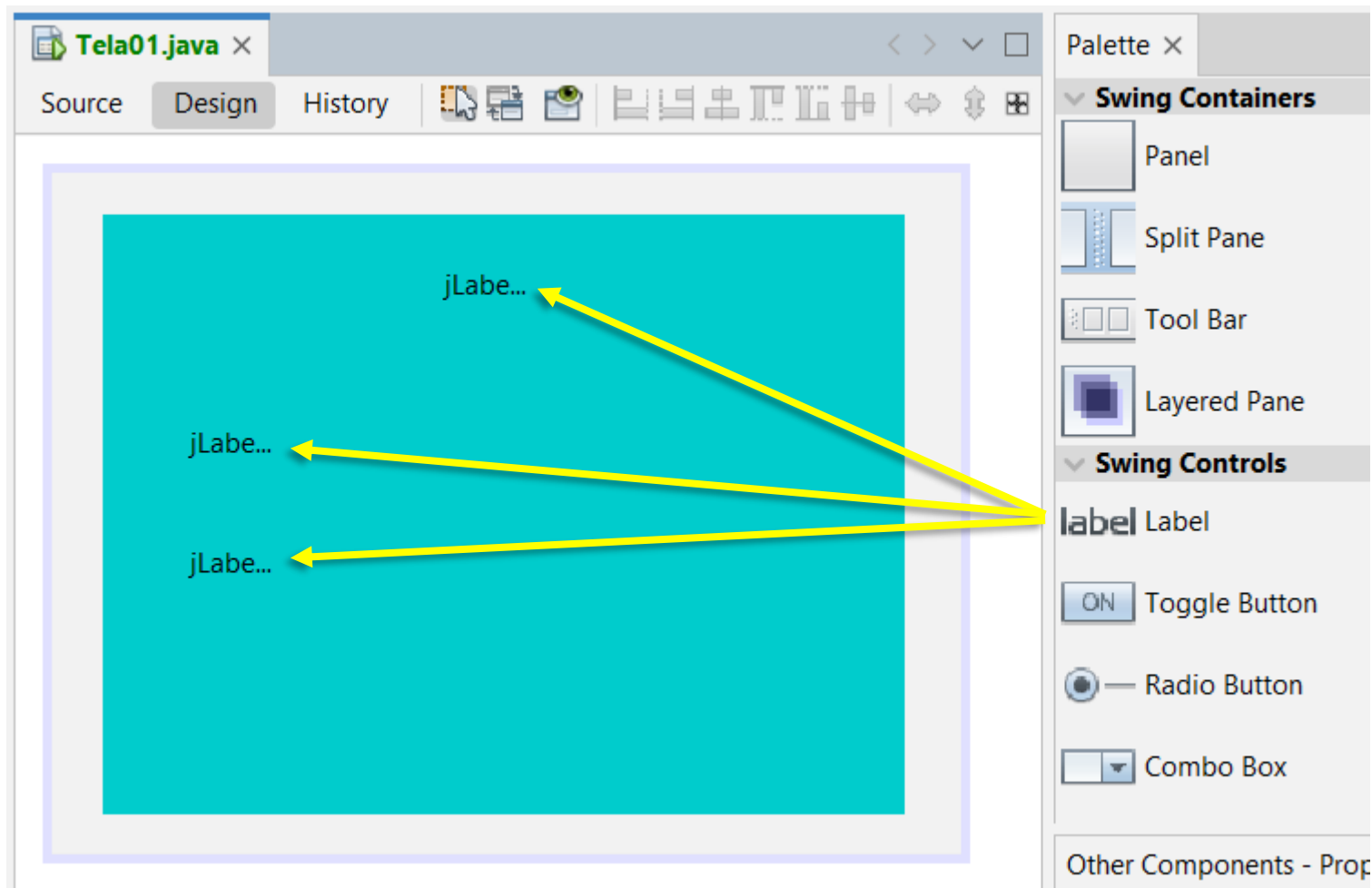


Java Swing – JLabel

- A classe **JLabel** permite **definir um texto** que **pode ser adicionado** a um **outro componente** (frame, painel etc.).
- Podem ser definidas várias **propriedades** para esse texto, tais como **alinhamento**, **tipo de letra**, **tamanho**, **cor** etc.



Java Swing – JLabel – Adicionar



Java Swing – JLabel – Editor Texto

The screenshot shows the Java Swing IDE with the following components:

- Design View:** A teal rectangular area representing a window. Inside, there is a grey rectangular component labeled "Tela 01" with a yellow border and a yellow arrow pointing to it from the Properties window. Below it, there are two labels labeled "jLabe...".
- Palette:** A panel on the right side of the IDE showing various Swing components. It is divided into two sections: "Swing Containers" and "Swing Controls".
- Swing Containers:** Includes Panel, Split Pane, Tool Bar, Layered Pane, Tabbed Pane, Scroll Pane, and Internal Frame.
- Swing Controls:** Includes Label and Button.
- Properties Window:** A window titled "jLabel1 [JLabel] - Properties" with three tabs: Properties, Events, and Code. The Properties tab is active, showing the following properties:

Properties	Events	Code
icon		<none>
labelFor		<none>
text		Tela 01
toolTipText		



Java Swing – JLabel – Editor Texto

The screenshot shows the Java Swing IDE with the following components:

- Design View:** A cyan rectangular area representing the GUI. Inside, there is a grey rectangular component labeled "Tela 0...". Below it, a smaller grey component is labeled "Mens" and "jLabe...". A yellow arrow points from the "text" property in the Properties window to the "Mens" label.
- Palette:** A panel on the right side of the IDE showing various Swing components. It is divided into two sections:
 - Swing Containers:** Panel, Split Pane, Tool Bar, Layered Pane, Tabbed Pane, Scroll Pane, Internal Frame.
 - Swing Controls:** Label, Button.
- Properties Window:** A window titled "jLabel2 [JLabel] - Properties" with three tabs: Properties, Events, and Code. The Properties tab is active, showing the following properties:

Properties	Events	Code
icon		<none>
labelFor		<none>
text		Mensagem 1:
toolTipText		



Java Swing – JLabel – Editor Texto

The screenshot shows an IDE window titled 'Tela01.java' with tabs for Source, Design, and History. The Design tab is active, showing a visual representation of a Java Swing window. The window has a cyan background and a grey rectangular area. Inside this area, there are two labels, both labeled 'Mens...', one above the other. A yellow arrow points from the second 'Mens...' label to the 'text' property field in the 'jLabel3 [JLabel] - Properties' window. The 'Palette' window on the right shows 'Swing Containers' and 'Swing Controls'. The 'Swing Controls' section includes 'Label' and 'Button'.

Swing Containers

- Panel
- Split Pane
- Tool Bar
- Layered Pane
- Tabbed Pane
- Scroll Pane
- Internal Frame

Swing Controls

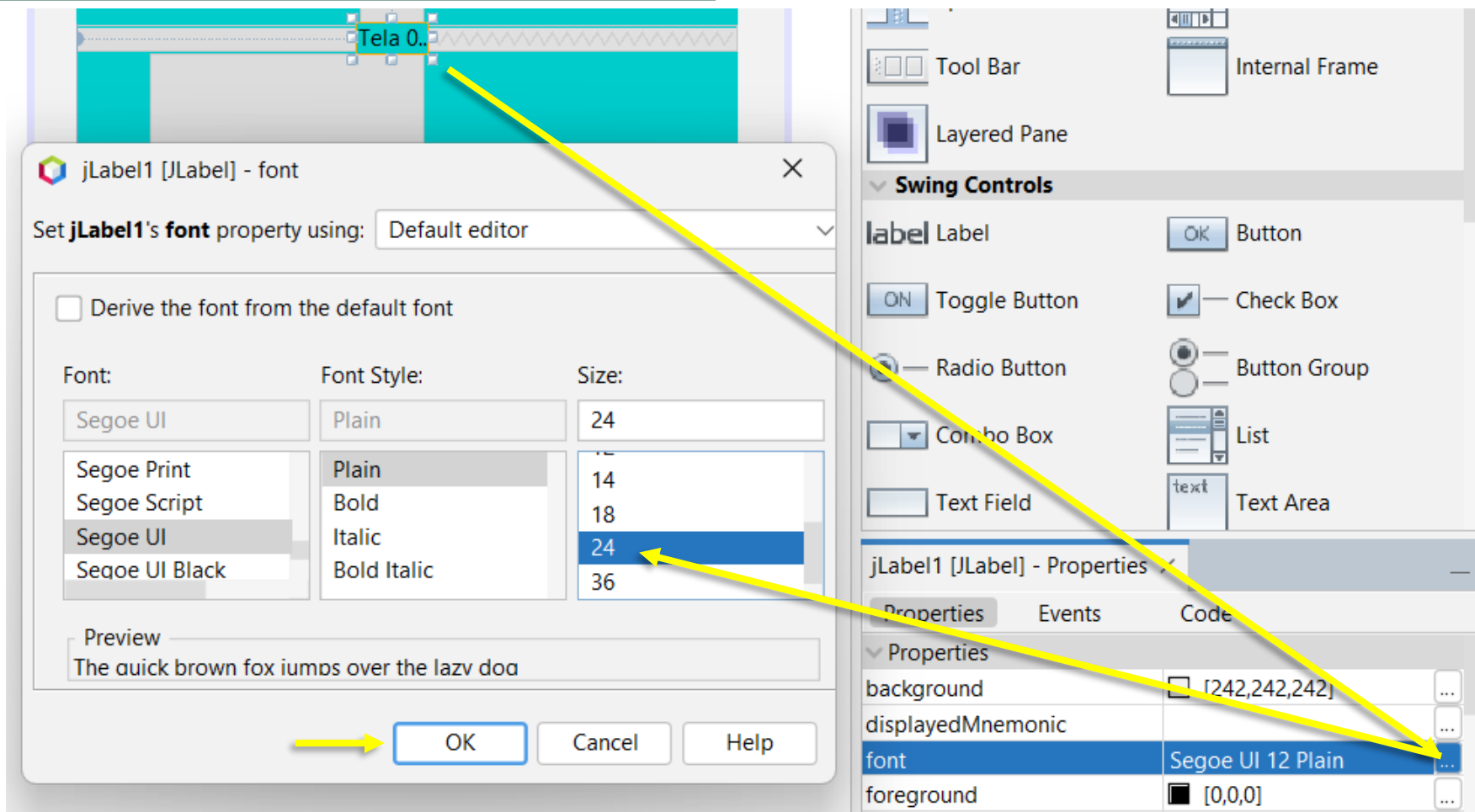
- Label
- Button

jLabel3 [JLabel] - Properties

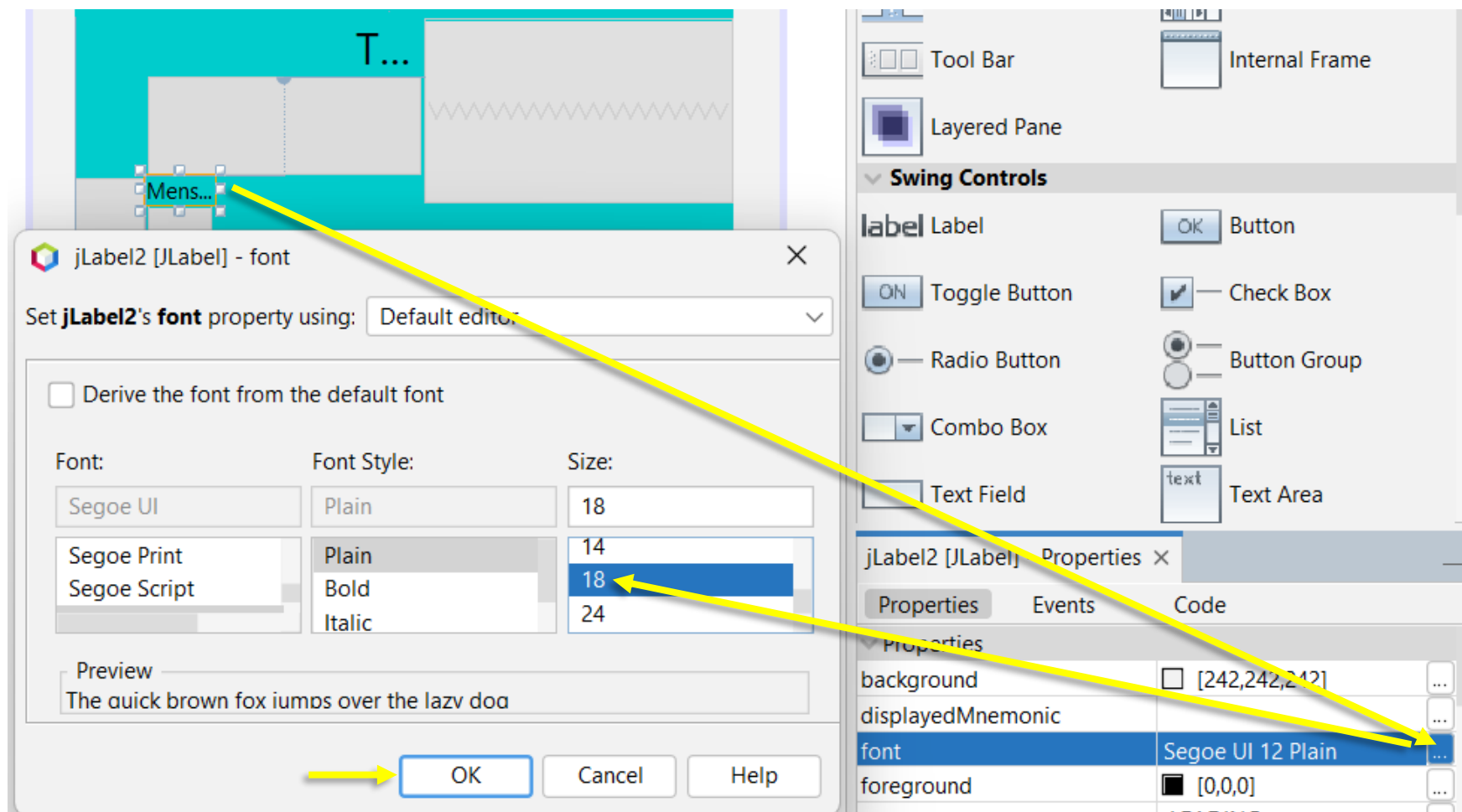
Properties	Events	Code
icon		<none>
labelFor		<none>
text		Mensagem 2:
toolTipText		



Java Swing – JLabel – Aumentar Fonte



Java Swing – JLabel – Aumentar Fonte



Java Swing – JLabel – Aumentar Fonte

The image shows a Java Swing IDE interface with two main windows. The top window is a preview of a JLabel component with the text "Mens...". Below it is a dialog box titled "jLabel3 [JLabel] - font". The dialog box has a dropdown menu set to "Default editor". It contains three sections: "Font:" with a list box showing "Segoe UI" and "Segoe Print"; "Font Style:" with a list box showing "Plain" and "Bold"; and "Size:" with a list box showing "18" and "24". A yellow arrow points from the "18" in the "Size:" list box to the "font" property in the "Properties" tab of the "jLabel3 [JLabel] - Properties" window. The "Properties" tab shows a table of properties for the JLabel component. The "font" property is highlighted, and its value is "Segoe UI 12 Plain". A yellow arrow points from the "font" property in the "Properties" tab to the "font" property in the "Properties" tab of the "jLabel3 [JLabel] - Properties" window. The "Properties" tab also shows other properties like "background", "displayedMnemonic", "foreground", "horizontalAlignment", and "icon".

Font settings for JLabel3:

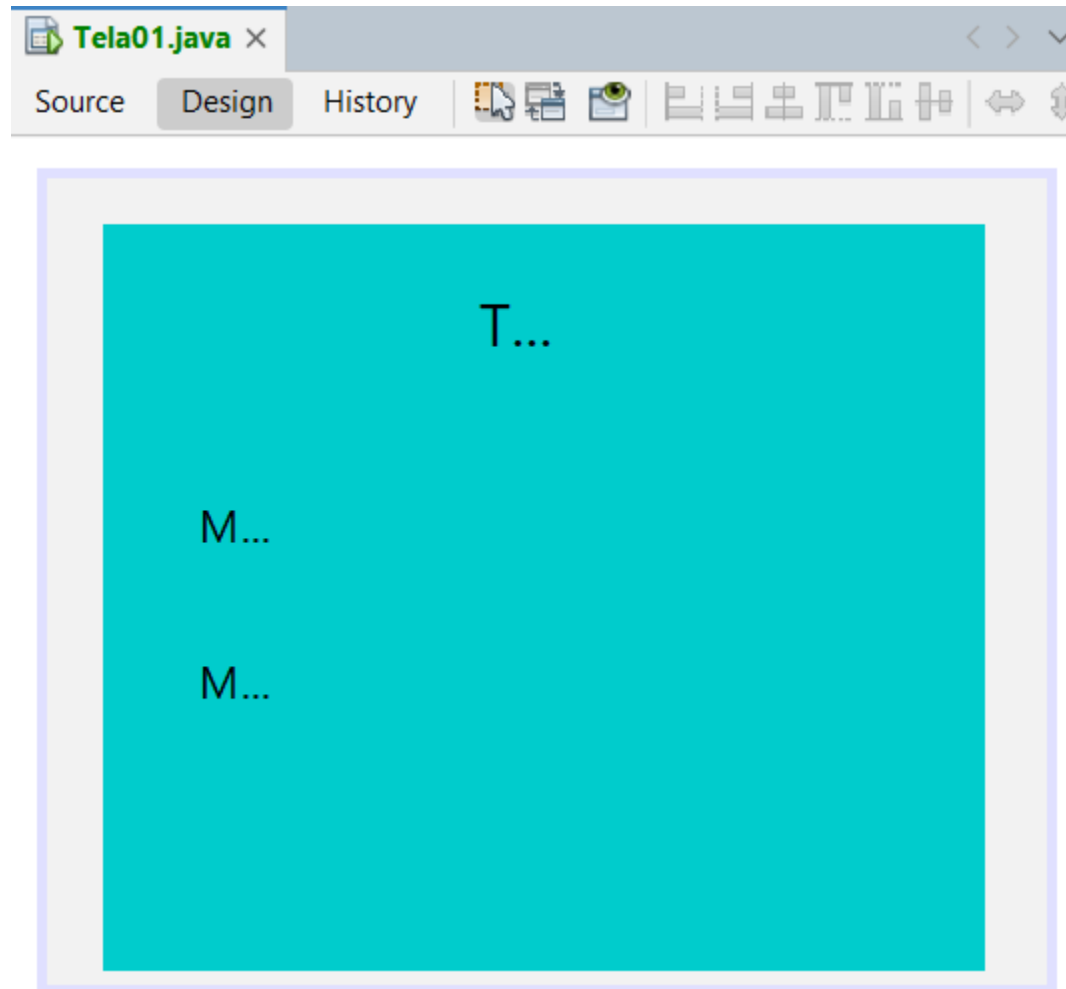
- Font: Segoe UI
- Font Style: Plain
- Size: 18

Properties for JLabel3:

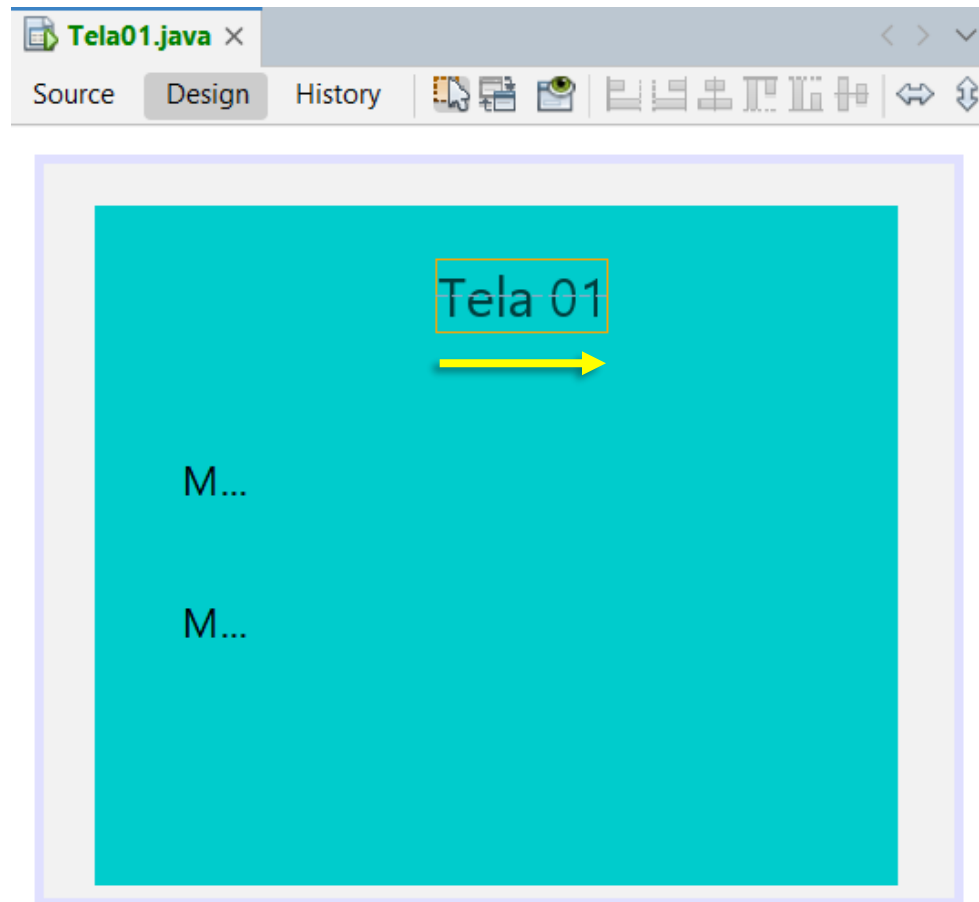
Property	Value
background	[242,242,242]
displayedMnemonic	
font	Segoe UI 12 Plain
foreground	[0,0,0]
horizontalAlignment	LEADING
icon	<none>



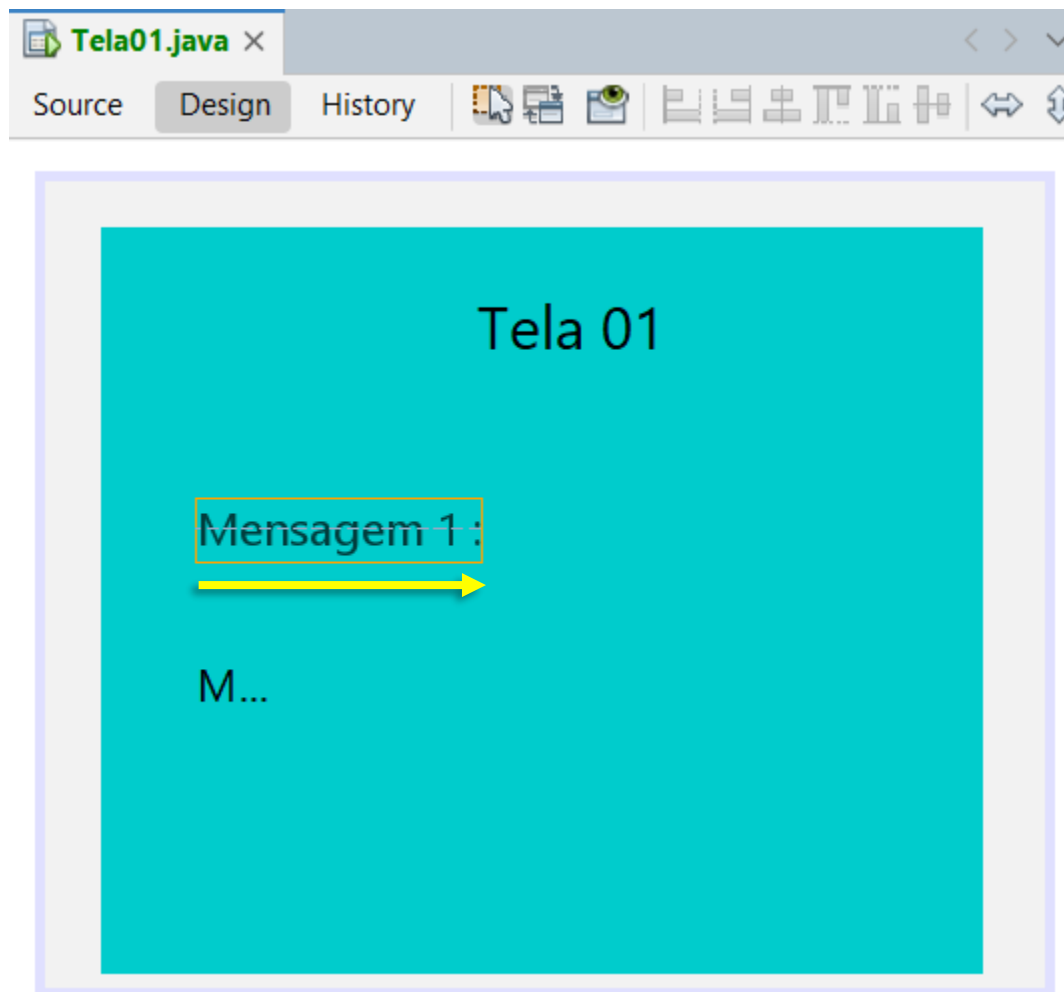
Java Swing – JLabel – Aumentar Fonte



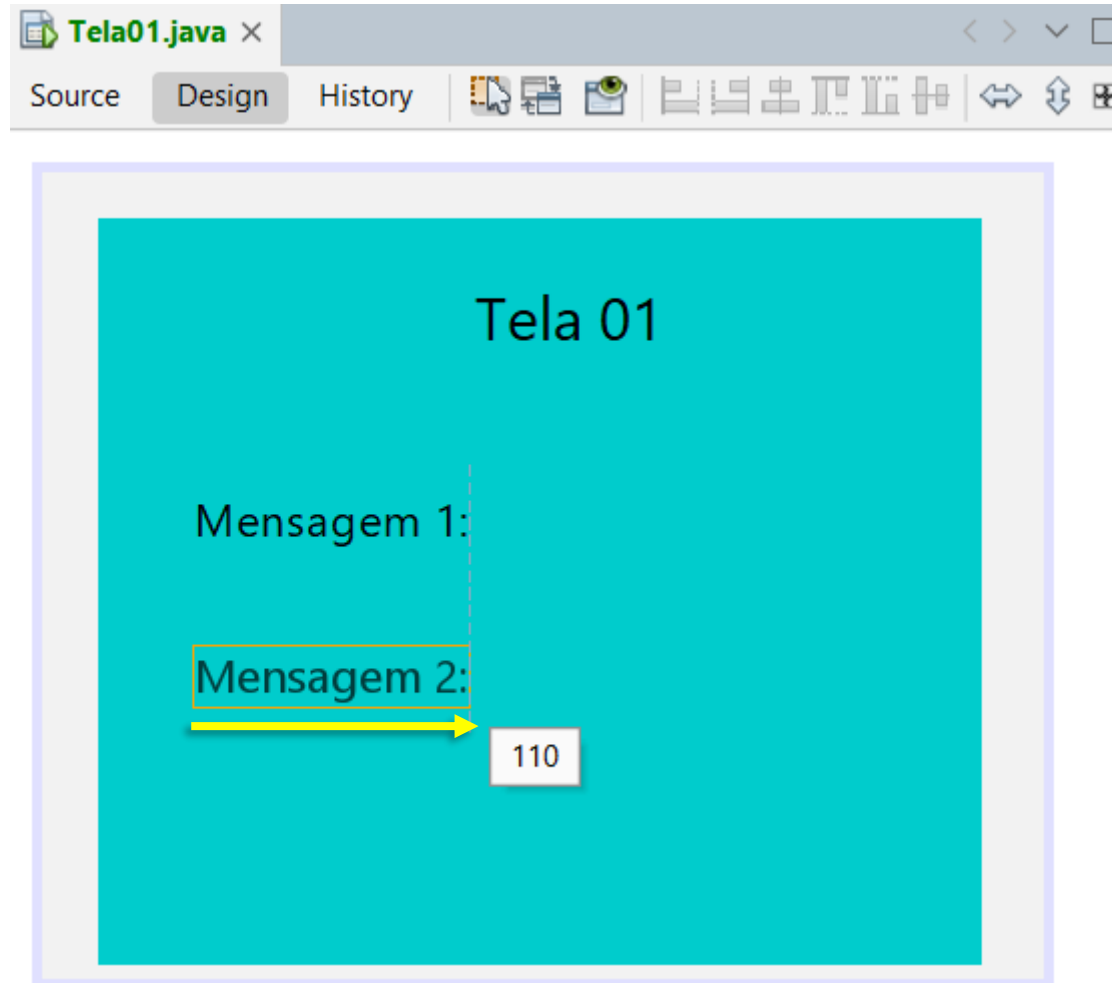
Java Swing – JLabel – Redimensionar



Java Swing – JLabel – Redimensionar



Java Swing – JLabel – Redimensionar

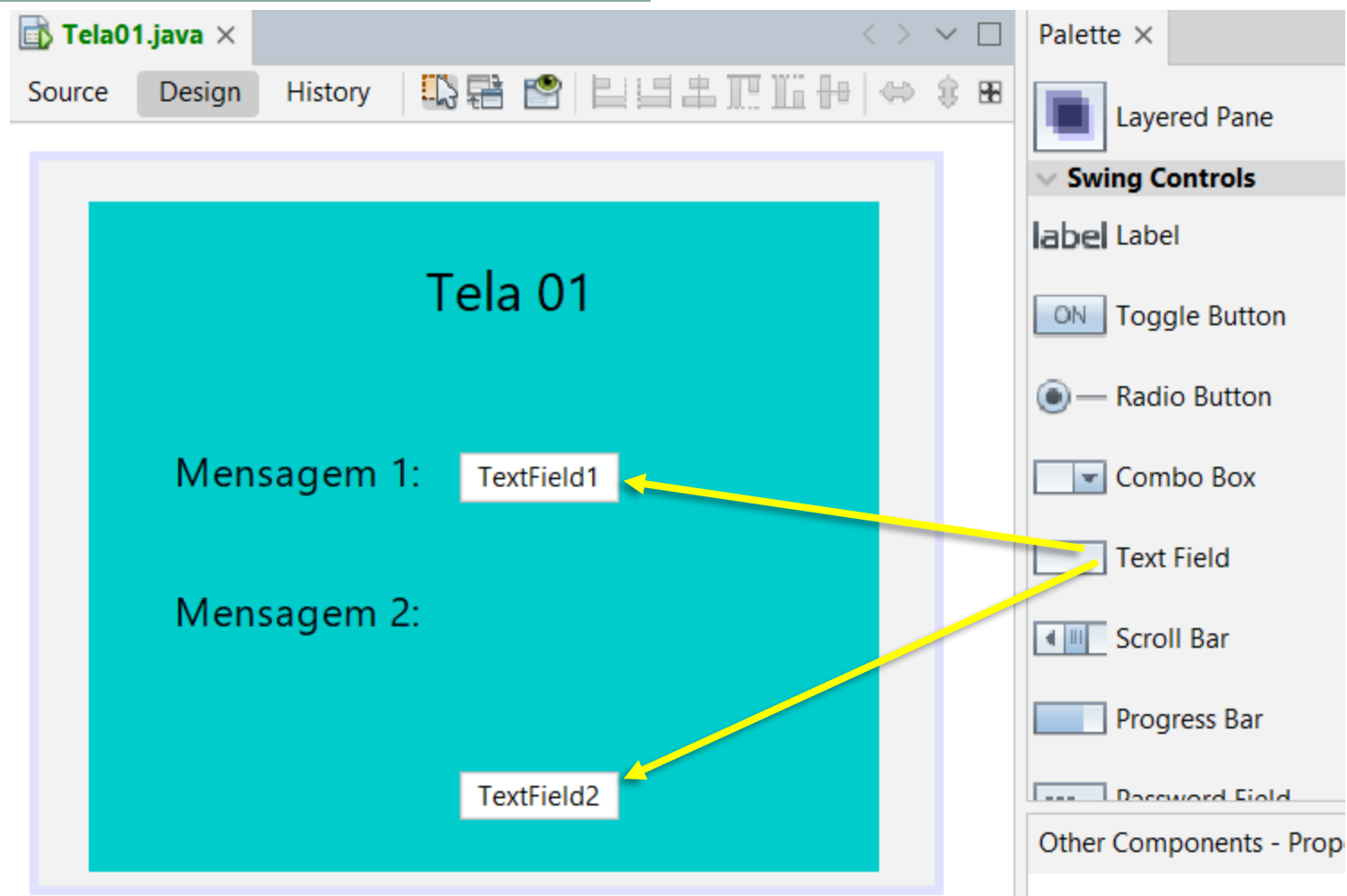


Java Swing – JTextField

- A classe **JTextField** permite criar uma **caixa de texto gráfica** na qual o usuário pode **digitar dados**.
- Existem **diversas propriedades** cujos **conteúdos podem ser modificados**.



Java Swing – JTextField – Adicionar



Java Swing – JTextField – Editor Texto

The image shows a screenshot of an IDE window titled 'Tela01.java' in Design mode. The main window contains a cyan background with a grey rectangular area labeled 'Tela 01'. Inside this area, there are two labels: 'Mensagem 1:' and 'Mensagem 2:'. A text field, labeled 'TextField2' at its bottom, is positioned to the right of 'Mensagem 1:'. A yellow arrow points from the text field to the 'Properties' tab of the 'jTextField1 [JTextField]' properties panel on the right.

Swing Controls Palette:

- Layered Pane
- Swing Controls
 - Label
 - Toggle Button
 - Radio Button
 - Combo Box
 - Button (OK)
 - Check Box
 - Button Group
 - List

jTextField1 [JTextField] - Properties

Properties	Events	Code
foreground		[0,0,0]
horizontalAlignment		LEADING
text		
toolTipText		
Other Properties		
UI		<default>



Java Swing – JTextField – Editor Texto

The screenshot shows an IDE window titled 'Tela01.java' with tabs for Source, Design, and History. The Design view displays a window titled 'Tela 01' with a cyan background. It contains two labels, 'Mensagem 1:' and 'Mensagem 2:', and a text field. A yellow arrow points from the text field to the 'text' property in the Properties window.

Palette

- Layered Pane
- Swing Controls**
 - Label
 - Toggle Button
 - Radio Button
 - Combo Box
 - Button
 - Check Box
 - Button Group
 - List

jTextField2 [JTextField] - Properties

Properties	Events	Code
foreground		[0,0,0]
horizontalAlignment		LEADING
text		
toolTipText		
Other Properties		
UI		<default>



Java Swing – JTextField – Renomear Var.

The screenshot shows an IDE window titled 'Tela01.java' with tabs for Source, Design, and History. The Design view displays a window titled 'Tela 01' with a cyan background. It contains two labels, 'Mensagem 1:' and 'Mensagem 2:', each followed by a text field. A yellow arrow points from the 'Mensagem 1:' label to the 'txtMensagem1 [JTextField] - Properties' panel on the right. The panel shows the 'Code' tab with the following table:

Code Generation	
Bean Class	class javax.swing.JTextField
Variable Name	txtMensagem1
Variable Modifiers	private
Type Parameters	
Use Local Variable	<input type="checkbox"/>



Java Swing – JTextField – Renomear Var.

The image shows a screenshot of an IDE interface. On the left, a window titled 'Tela01.java' is in 'Design' mode. It displays a window titled 'Tela 01' with a cyan background. Inside, there are two labels: 'Mensagem 1:' and 'Mensagem 2:'. Below 'Mensagem 1:', there is a text field. A yellow arrow points from this text field to the 'Code' tab of the 'jTextField2 [JTextField] - Properties' window on the right. The 'Code' tab shows the 'Variable Name' set to 'txtMensagemFinal'.

Palette ×

Layered Pane

Swing Controls

- Label
- Toggle Button
- Radio Button
- Combo Box
- Button
- Check Box
- Button Group
- List

jTextField2 [JTextField] - Properties ×

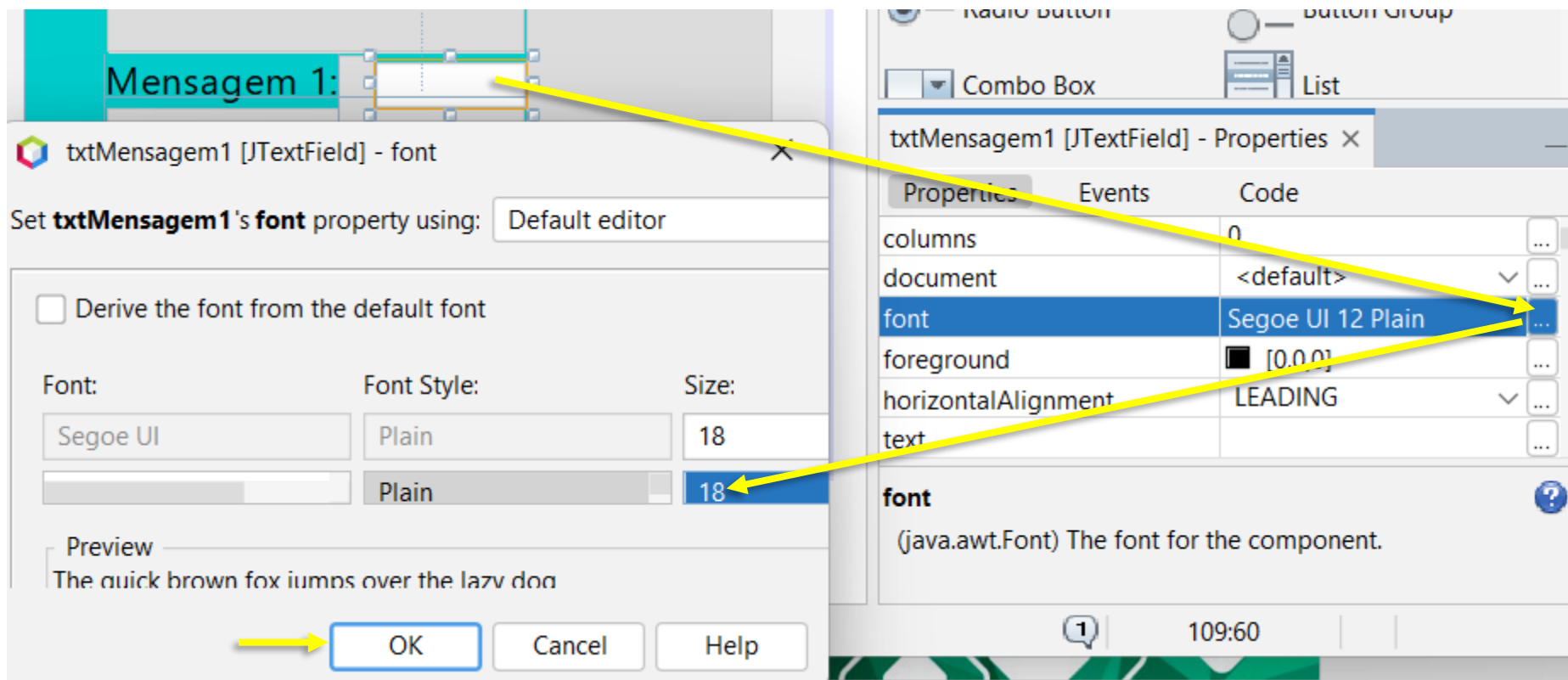
Properties Events Code

Code Generation

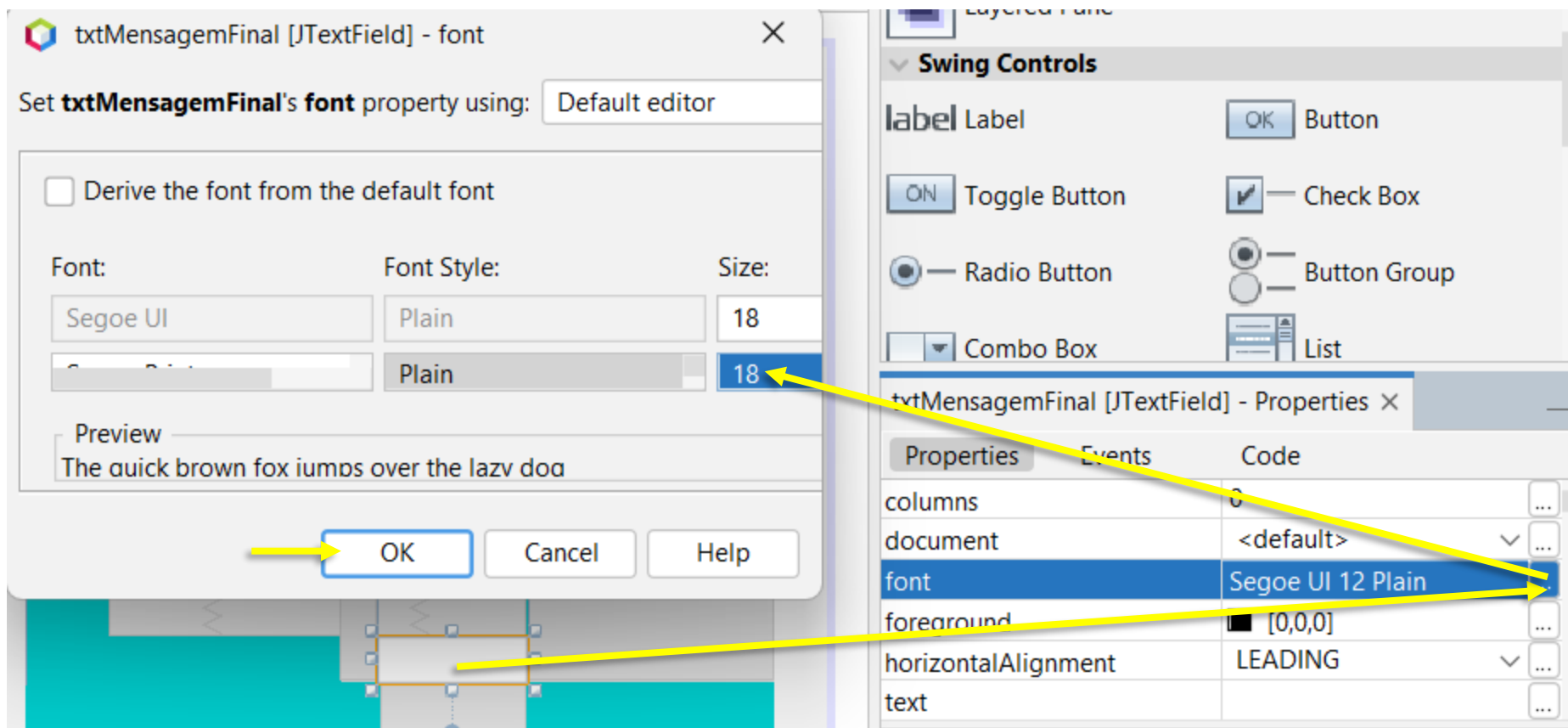
Bean Class	class javax.swing.JTextField
Variable Name	txtMensagemFinal
Variable Modifiers	private
Type Parameters	
Use Local Variable	<input type="checkbox"/>



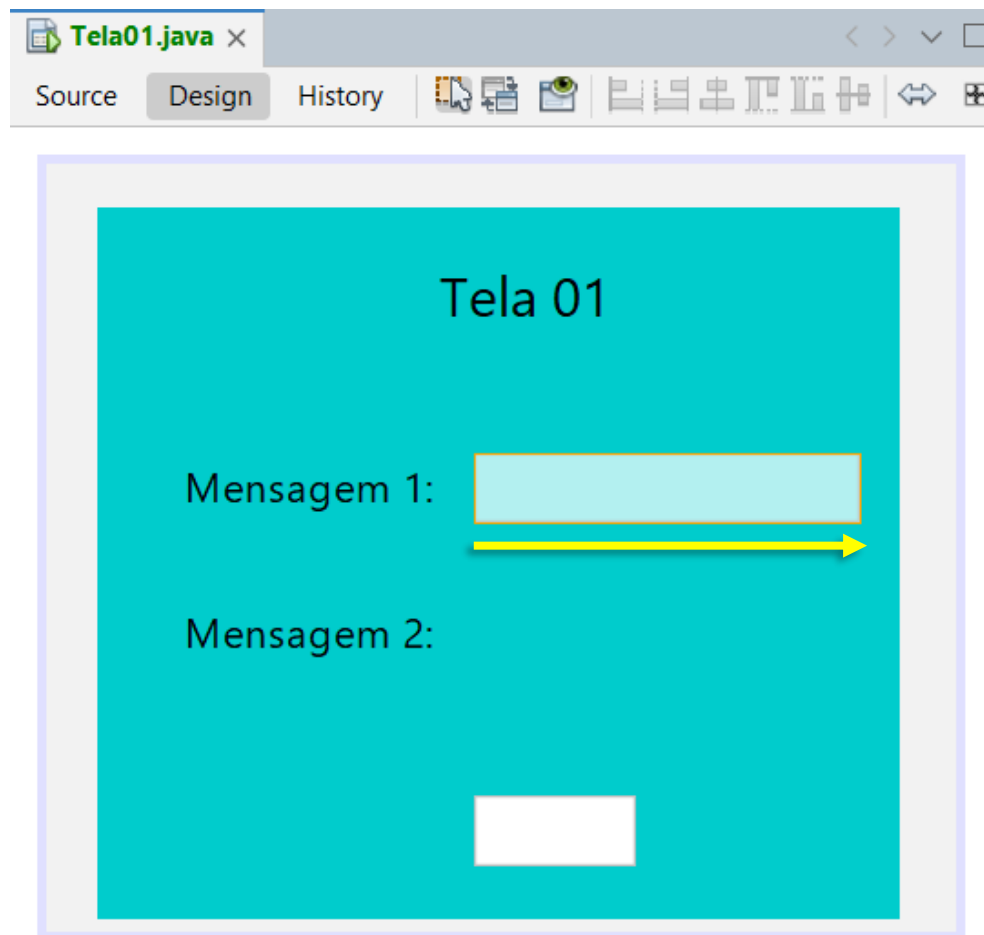
Java Swing – JTextField – Aument. Fonte



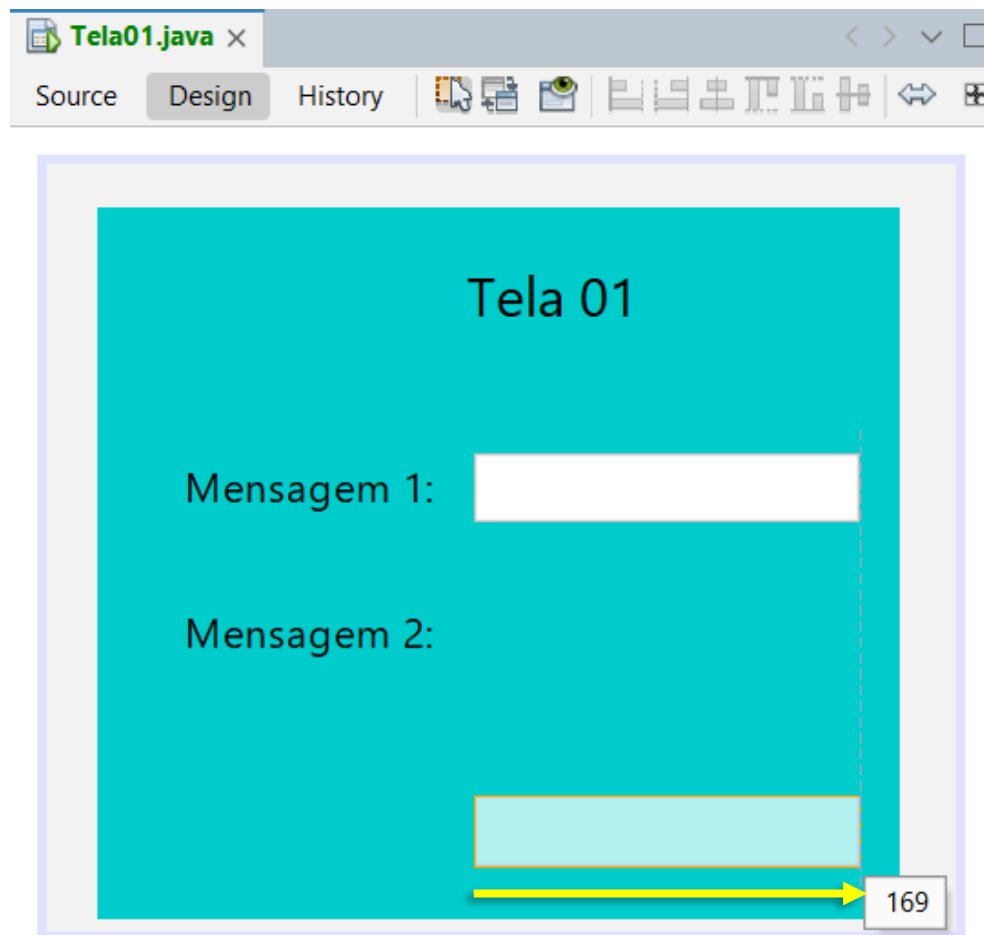
Java Swing – JTextField – Aument. Fonte



Java Swing – JTextField – Redimens.



Java Swing – JTextField – Redimens.



Java Swing – JTextField – Não Editável

The screenshot shows an IDE window titled 'Tela01.java' with tabs for Source, Design, and History. The Design view displays a window titled 'Tela 01' with a cyan background. It contains two labels, 'Mensagem 1:' and 'Mensagem 2:', each followed by a text field. The text field for 'Mensagem 2:' is highlighted with a yellow box, and a yellow arrow points from it to the 'editable' property in the Properties panel, which is set to false.

Palette:

- Radio Button
- Button Group
- Combo Box
- List
- Text Field
- Text Area
- Scroll Bar
- Slider
- Progress Bar
- Formatted Field
- Password Field
- Spinner
- Separator
- Text Pane

txtMensagemFinal [JTextField] - Properties

Properties	Events	Code
editable		<input type="checkbox"/>
background		<input type="checkbox"/> [242,242,242]
columns		0

editable

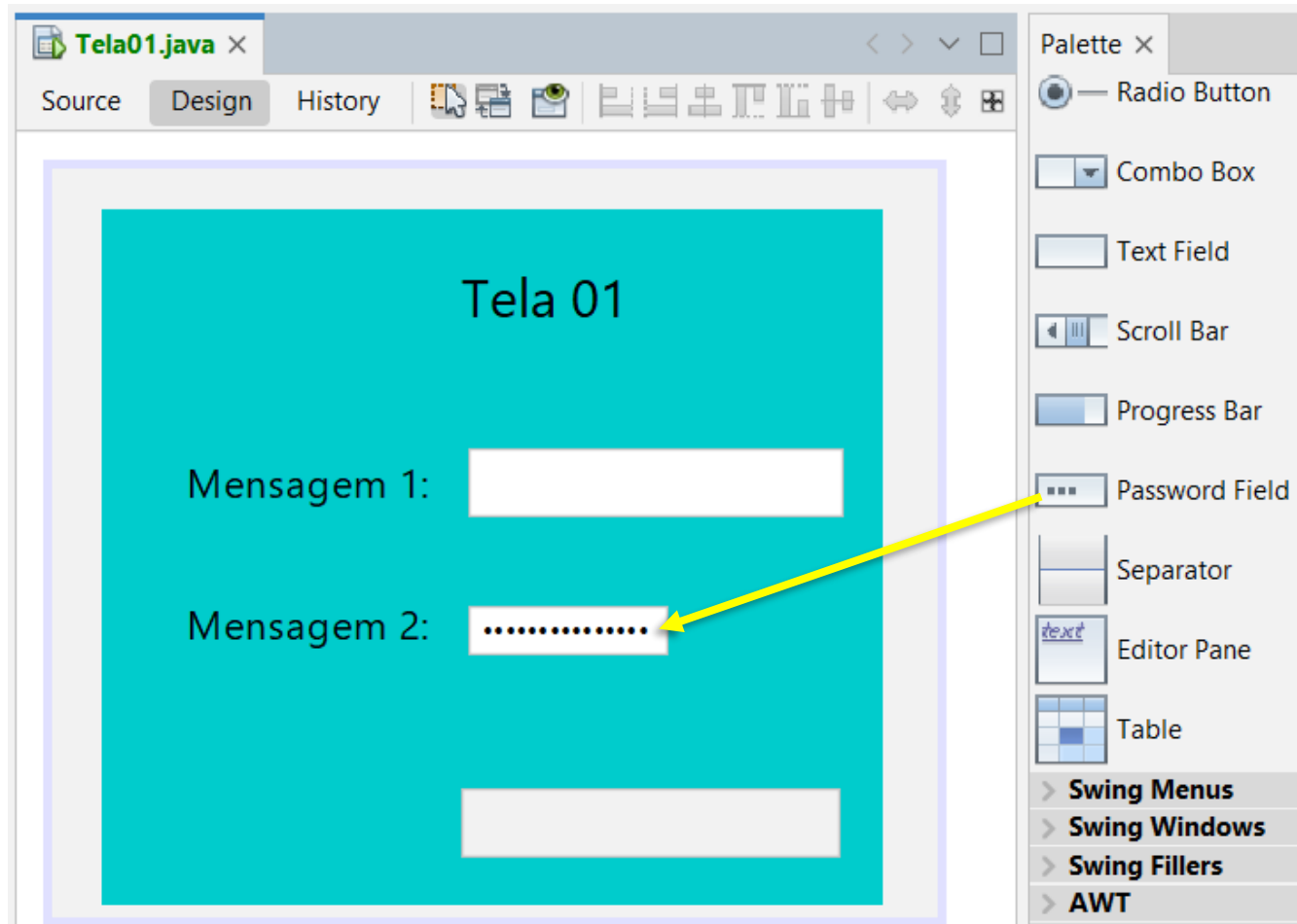


Java Swing – JPasswordField

- A classe **JPasswordField** permite criar uma **caixa de texto gráfica** na qual o usuário pode **digitar dados**, porém eles são substituídos (visualmente) por outro caractere (*).



Java Swing – JPasswordField – Adic.



Java Swing – JPasswordField – Editor T.

The screenshot displays the NetBeans IDE interface for a Java Swing application. The main window, titled 'Tela 01', is in the 'Design' view. It features a cyan background with two labels, 'Mensagem 1:' and 'Mensagem 2:', each followed by a text input field. The second input field is a JPasswordField, highlighted with a yellow box and a yellow arrow pointing to its properties. The 'Properties' tab of the 'jPasswordField1 [JPasswordField] - Properties' window is open, showing the 'text' property set to an empty string. The 'Palette' window on the right shows various Swing components, including 'Password Field'.

Properties

Properties	Events	Code
foreground		[0,0,0]
horizontalAlignment		LEADING
text		
toolTipText		

jPasswordField1 [JPasswordField]



Java Swing – JPasswordField – Ren. Var.

The screenshot shows an IDE window titled 'Tela01.java' with tabs for Source, Design, and History. The Design view displays a window titled 'Tela 01' with a cyan background. Inside the window, there are two labels: 'Mensagem 1:' and 'Mensagem 2:'. Below 'Mensagem 2:', there is a JPasswordField component. A yellow arrow points from the JPasswordField to the 'txpMensagem2 [JPasswordField] - Properties' window. The Properties window shows the 'Code' tab with the following information:

Code Generation	
Bean Class	class javax.swing.JPassword...
Variable Name	txpMensagem2
Variable Modifiers	private

Below the table, the text 'txpMensagem2 [JPasswordField]' is displayed.



Java Swing – JPasswordField – Aum. F.

The screenshot displays the Java Swing IDE interface. A dialog box titled "txpMensagem2 [JPasswordField] - font" is open, showing the font configuration for the selected component. The dialog has a "Font:" field set to "Segoe UI", a "Font Style:" field set to "Plain", and a "Size:" field set to "18". The "OK" button is highlighted with a yellow arrow. Below the dialog, a preview of the component shows the text "Mensagem 2:" and a password field. A yellow arrow points from the "OK" button to the "font" property in the "txpMensagem2 [JPasswordField] - Properties" window. The "font" property is set to "Segoe UI 12 Plain".

Font configuration details:

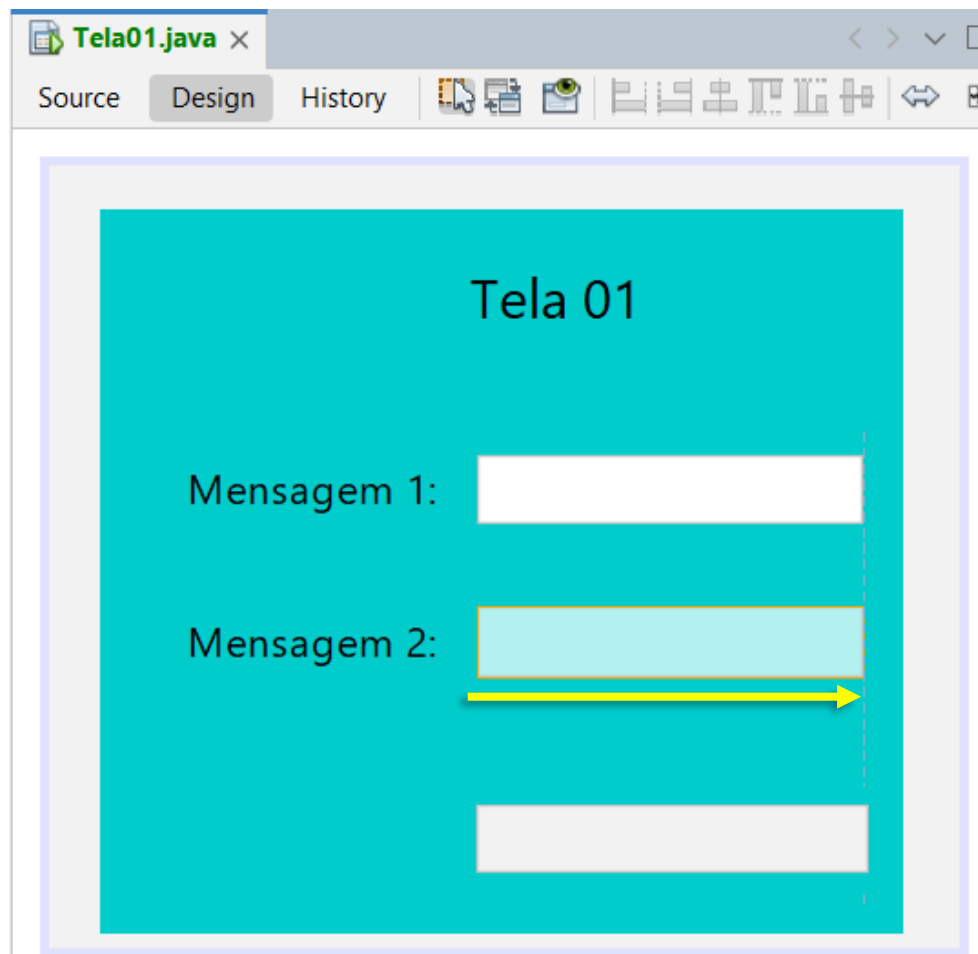
Font	Font Style	Size
Segoe UI	Plain	18

Properties window details:

Property	Value
columns	0
document	<default>
font	Segoe UI 12 Plain
foreground	[0,0,0]



Java Swing – JPasswordField – Redimen.

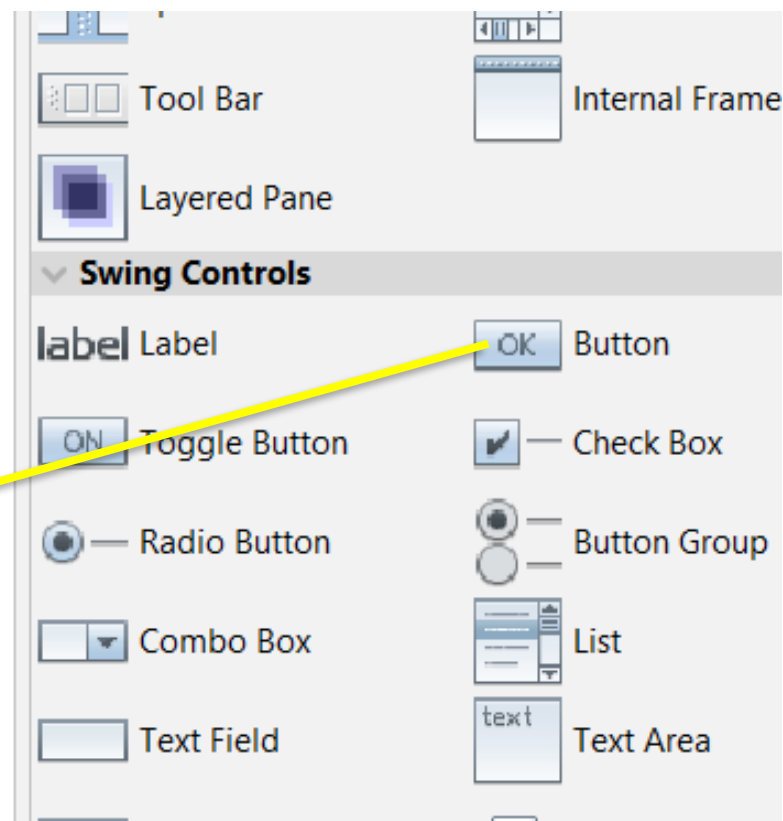
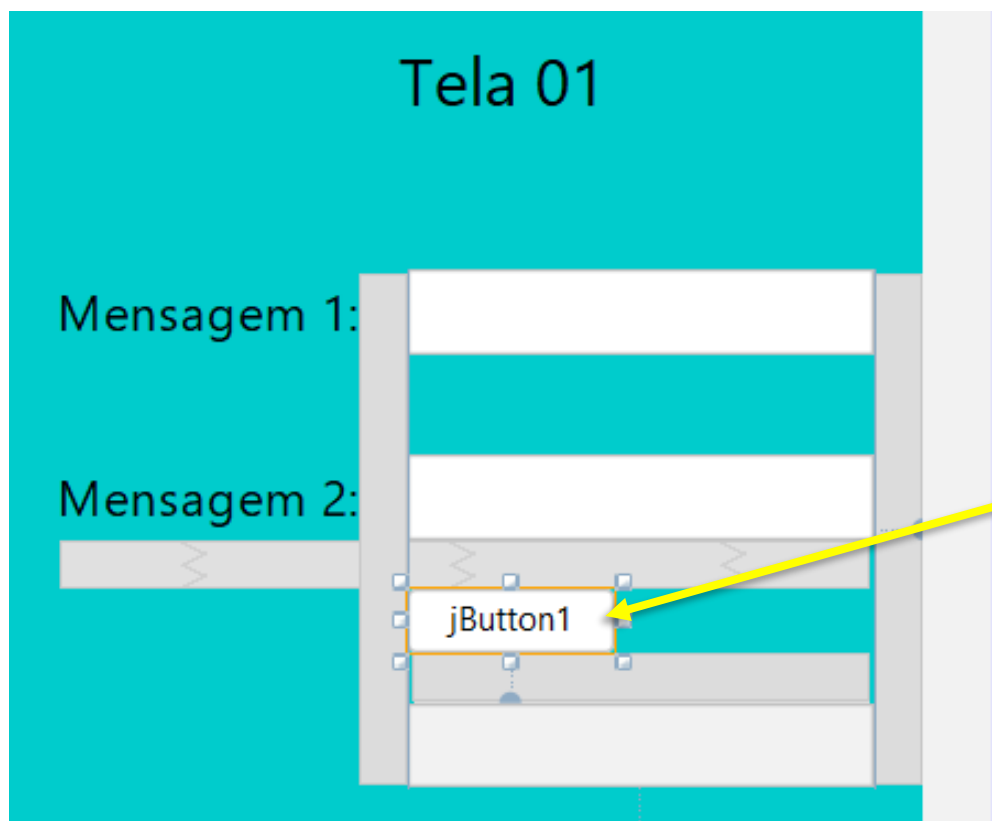


Java Swing – JButton

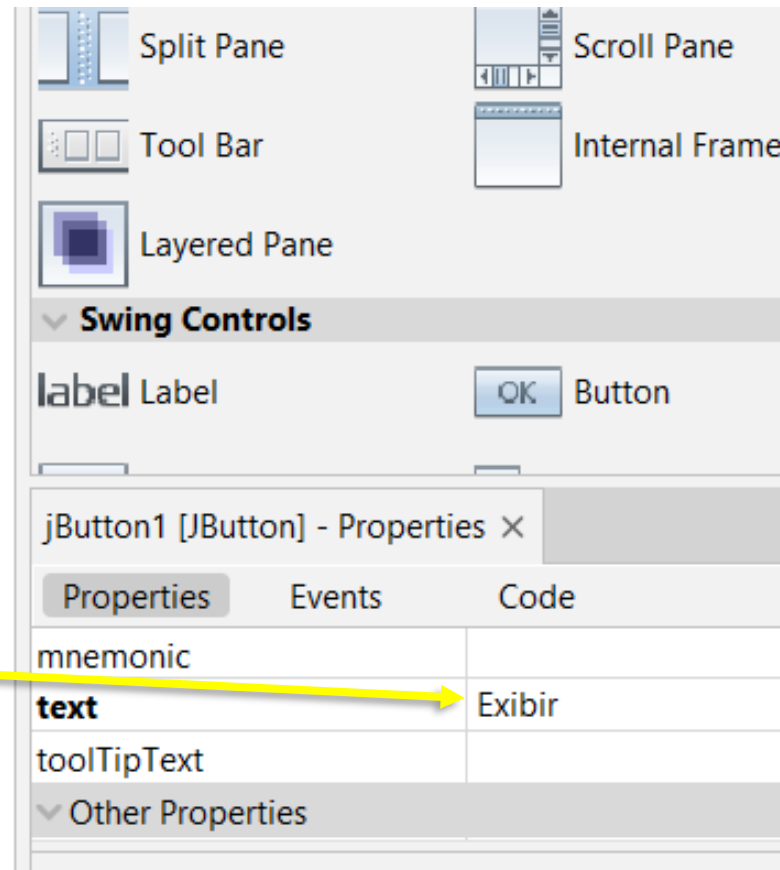
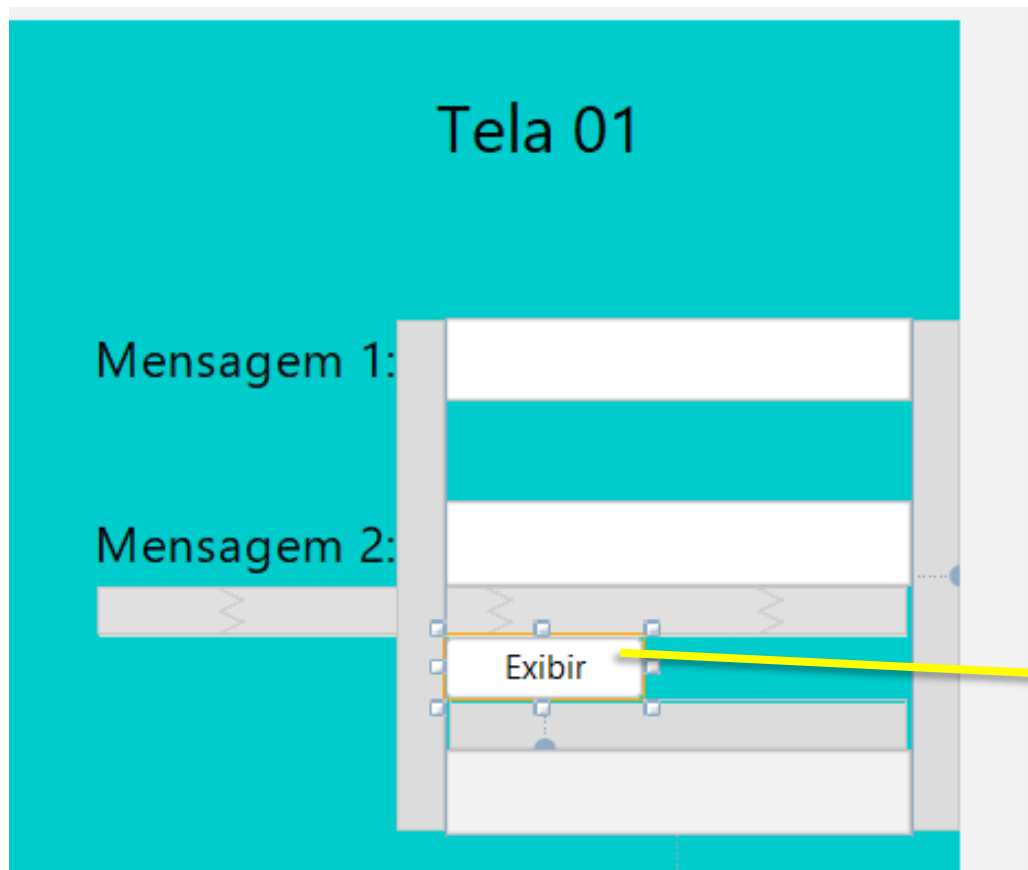
- A classe **JButton** permite **definir botões gráficos** que **pode ser adicionados** a um **outro componente** (frame, painel etc.).
- Um **botão** pode **ser criado** com apenas um **texto e/ou com ícones** para tornar o ambiente mais intuitivo.



Java Swing – JButton – Adicionar



Java Swing – JButton – Editor Texto



Java Swing – JButton – Aumentar Fonte

The image shows a Java Swing IDE interface with two main windows. The left window is titled "jButton1 [JButton] - font" and contains a "Default editor" tab. It has a checkbox "Derive the font from the default font" which is unchecked. Below this are three fields: "Font:" with "Segoe UI", "Font Style:" with "Plain", and "Size:" with "18". A "Preview" section shows the text "The quick brown fox jumps over the lazy dog" in the selected font. At the bottom are "OK", "Cancel", and "Help" buttons. A yellow arrow points from the "OK" button to the "font" property in the right window. The right window is titled "jButton1 [JButton] - Properties" and has three tabs: "Properties", "Events", and "Code". The "Properties" tab is active, showing a list of properties. The "font" property is highlighted in blue, and its value is "Segoe UI 12 Plain". A yellow arrow points from the "font" property value to the "font" field in the left window. The background shows a partial view of a Swing container with a button labeled "Exibir".

Set **jButton1**'s **font** property using: Default editor

☐ Derive the font from the default font

Font: Segoe UI Font Style: Plain Size: 18

Preview
The quick brown fox jumps over the lazy dog

OK Cancel Help

Swing Containers

- Panel
- Split Pane
- Tool Bar
- Layered Pane
- Tabbed Pane
- Scroll Pane
- Internal Frame

Swing Controls

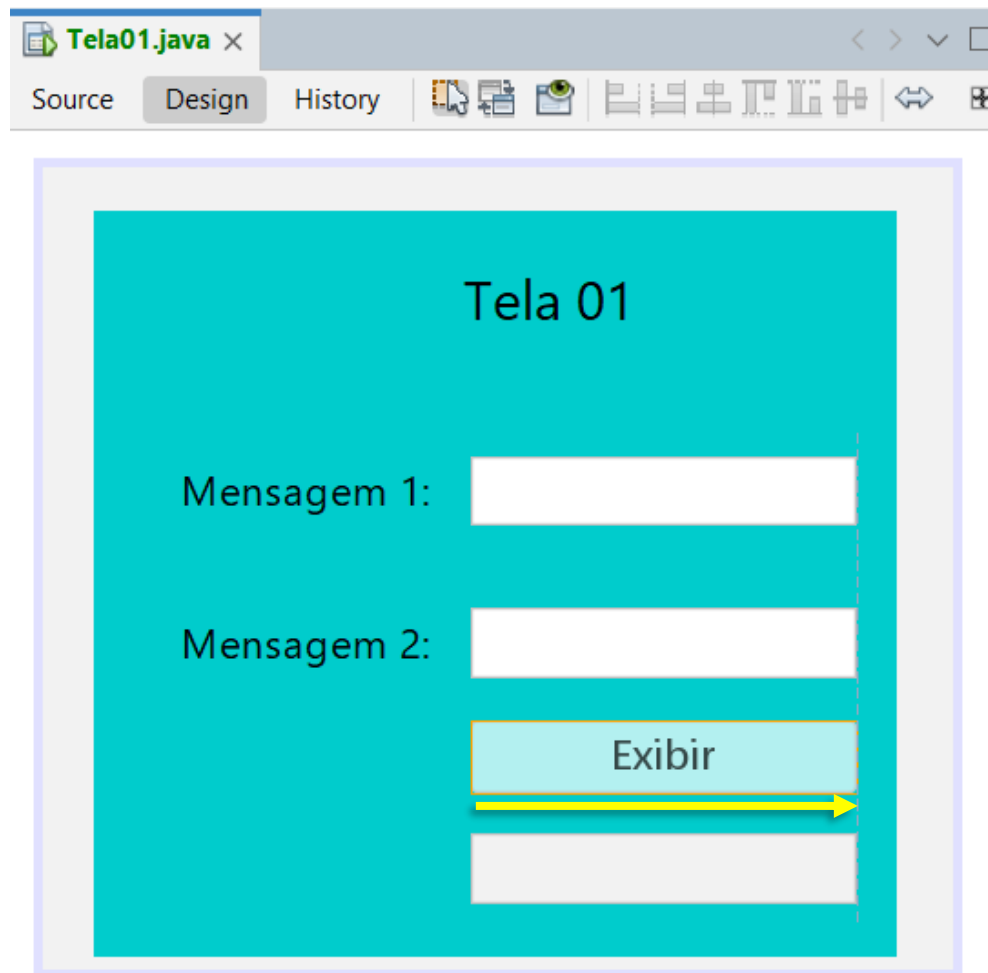
- Label
- Button

jButton1 [JButton] - Properties

Properties	Events	Code
font		Segoe UI 12 Plain
foreground		[0,0,0]
icon		<none>



Java Swing – JButton – Redimensionar



Java Swing – JButton – Renomear Var.

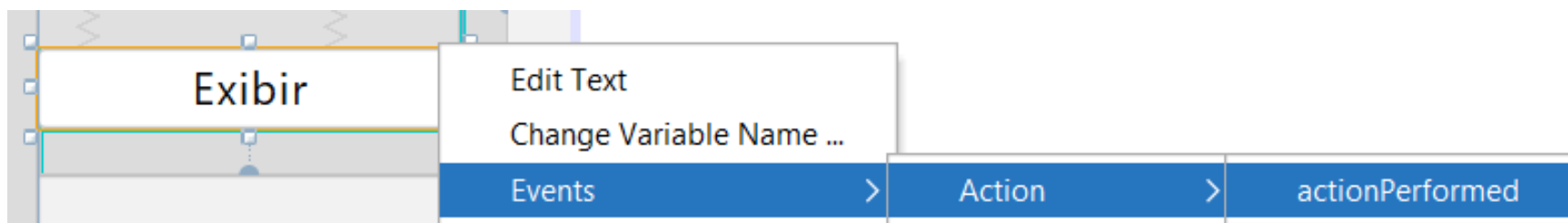
The image shows a Java Swing IDE interface. The main window, titled 'Tela 01', is in Design mode. It contains a cyan background with the text 'Tela 01'. Below this, there are two labels: 'Mensagem 1:' and 'Mensagem 2:'. To the right of these labels is a vertical stack of components. The bottom component is a JButton labeled 'Exibir'. A yellow arrow points from the 'Exibir' label to the 'Variable Name' field in the 'Properties' window.

The 'Properties' window for 'btnExibir [JButton]' is open, showing the 'Code' tab. The 'Code Generation' section is expanded, showing the following fields:

Field	Value
Bean Class	class javax.swing.JButton
Variable Name	btnExibir
Variable Modifiers	private



Java Swing – JButton – Adicionar Ação



Java Swing – JButton – Adicionar Ação

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

Java Swing – JButton – Adicionar Ação

```
import javax.swing.JOptionPane;
```



Java Swing – JButton – Adicionar Ação

```
private void btnExibirActionPerformed(java.awt.event.ActionEvent evt) {  
  
    String mensagem1 = txtMensagem1.getText();  
    String mensagem2 = String.valueOf(txpMensagem2.getPassword());  
  
    String mensagemFinal = mensagem1 + " " + mensagem2;  
  
    txtMensagemFinal.setText(mensagemFinal);  
  
    JOptionPane.showMessageDialog(this, mensagemFinal);  
  
}
```



Java Swing – JButton – Execução



Sistema IFBA

Tela 01

Mensagem 1:

Mensagem 2:



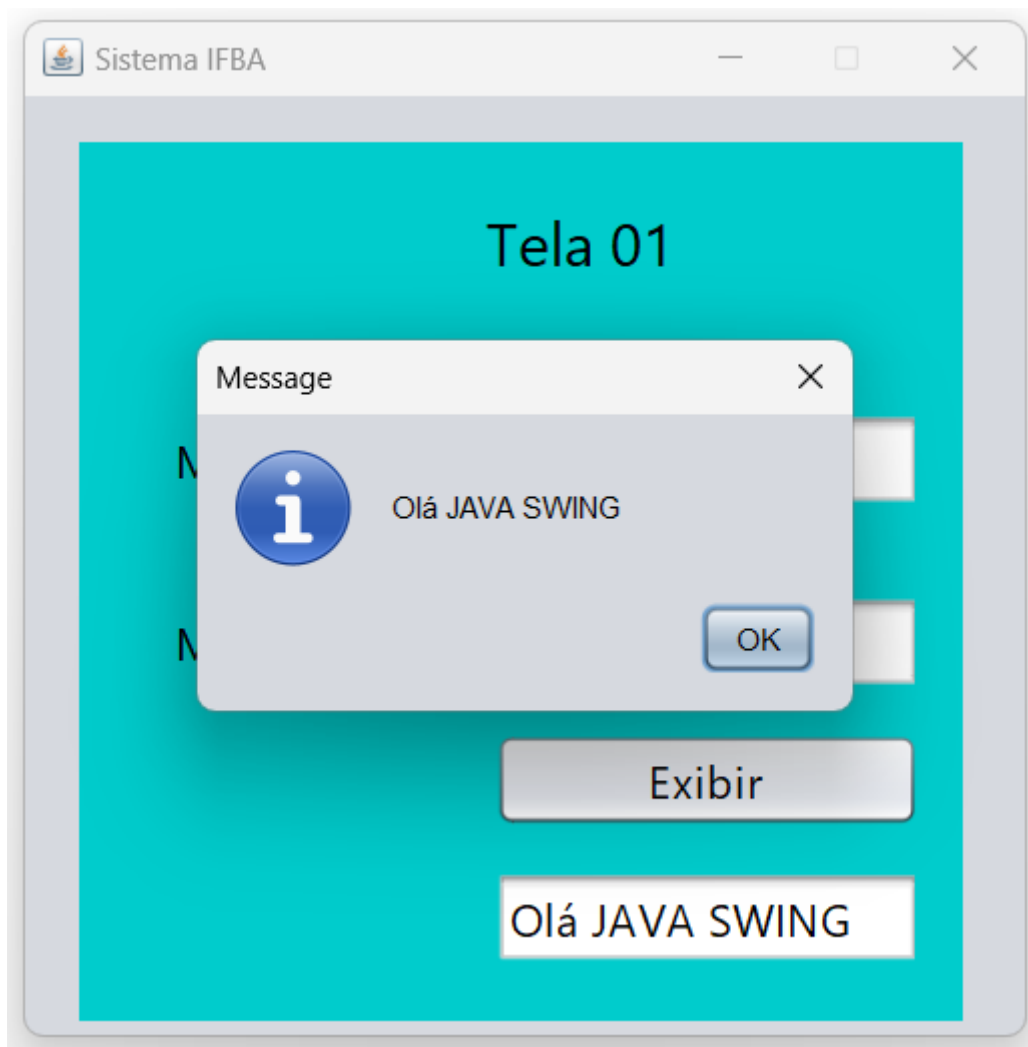
Java Swing – JButton – Execução



The screenshot shows a Java Swing window titled "Sistema IFBA". The window has a light gray title bar with standard minimize, maximize, and close buttons. The main content area has a cyan background. At the top center, the text "Tela 01" is displayed. Below this, there are two text input fields. The first field is labeled "Mensagem 1:" and contains the text "Olá". The second field is labeled "Mensagem 2:" and contains a series of asterisks "*****". Below the second field is a button labeled "Exibir". At the bottom of the cyan area, there is an empty text input field.



Java Swing – JButton – Execução



Java Swing – JButton – Execução



Java Swing – Exercício – Prog. Soma

The image shows a Java Swing window titled "Sistema IFBA". Inside the window is a form titled "Programa Soma". The form contains the following elements:

- lblValor1**: Label for the first input field.
- txtValor1**: Text input field for the first value.
- lblValor2**: Label for the second input field.
- txtValor2**: Text input field for the second value.
- btnSomar**: Button labeled "Somar" to perform the addition.
- lblResultado**: Label for the result field.
- txtResultado**: Text input field for the result.

Yellow arrows point from the labels in the boxes to the corresponding UI components in the window.

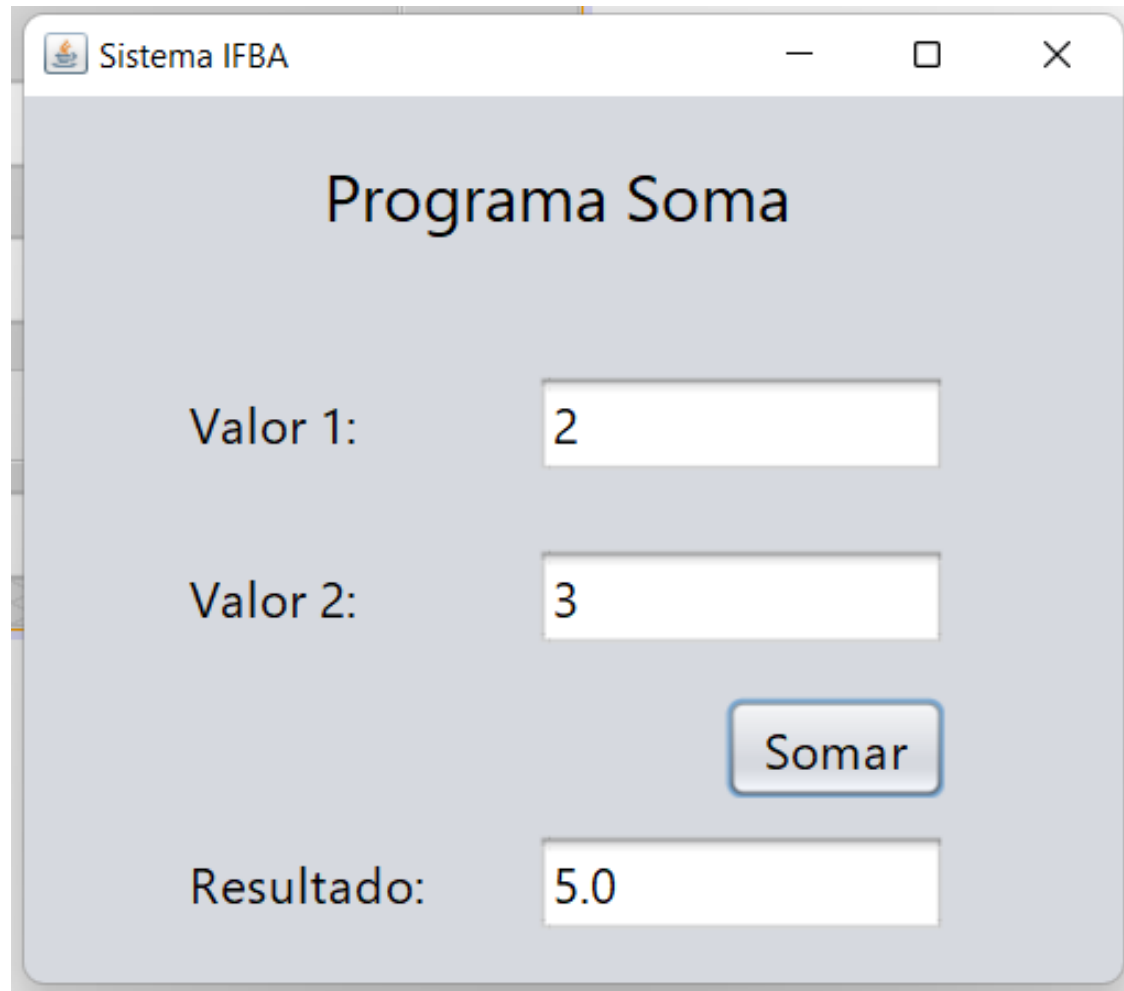
Java Swing – Exercício – Prog. Soma

```
private void btnSomarActionPerformed(java.awt.event.ActionEvent evt) {  
  
    //Lendo o texto de uma caixa de texto e atribundo-o a uma variável.  
    String strValor1 = txtValor1.getText();  
    //Converte um valor de string para double e o resultado a uma variável.  
    double valor1 = Double.parseDouble(strValor1);  
  
    //Lendo o texto de uma caixa de texto e atribundo-o a uma variável.  
    String strValor2 = txtValor2.getText();  
    //Converte um valor de string para double e o resultado a uma variável.  
    double valor2 = Double.parseDouble(strValor2);  
  
    //Soma duas variáveis do tipo double e atribui a uma terceira variável.  
    double resultado = valor1 + valor2;  
    //Converte um valor de double para string e o resultado a uma variável.  
    String strResultado = String.valueOf(resultado);  
  
    //Atribuindo o texto de uma variável a uma caixa de texto.  
    txtResultado.setText(strResultado);  
}
```



Java Swing – Exercício – Prog. Soma

Execução



Sistema IFBA

Programa Soma

Valor 1: 2

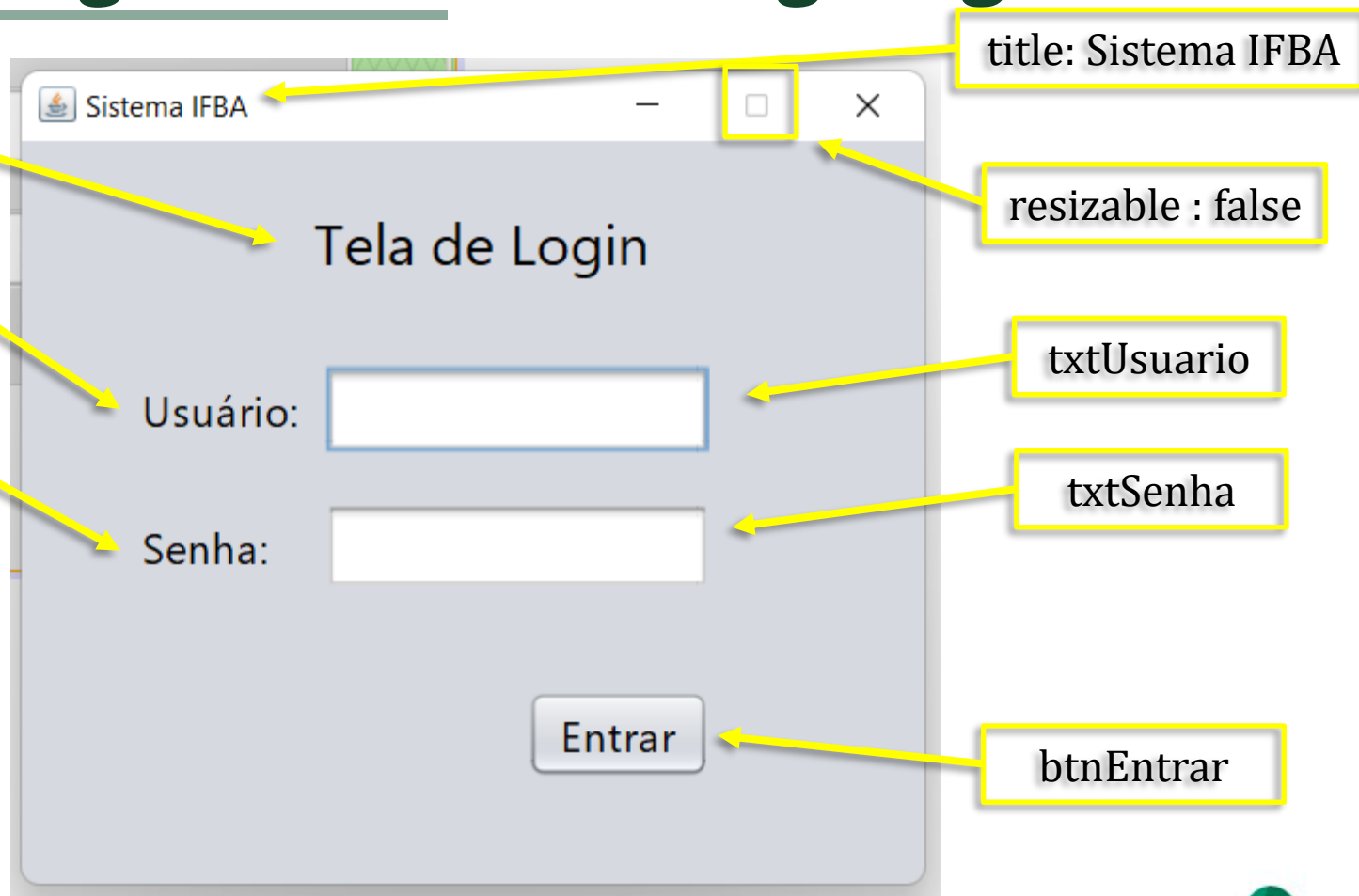
Valor 2: 3

Somar

Resultado: 5.0



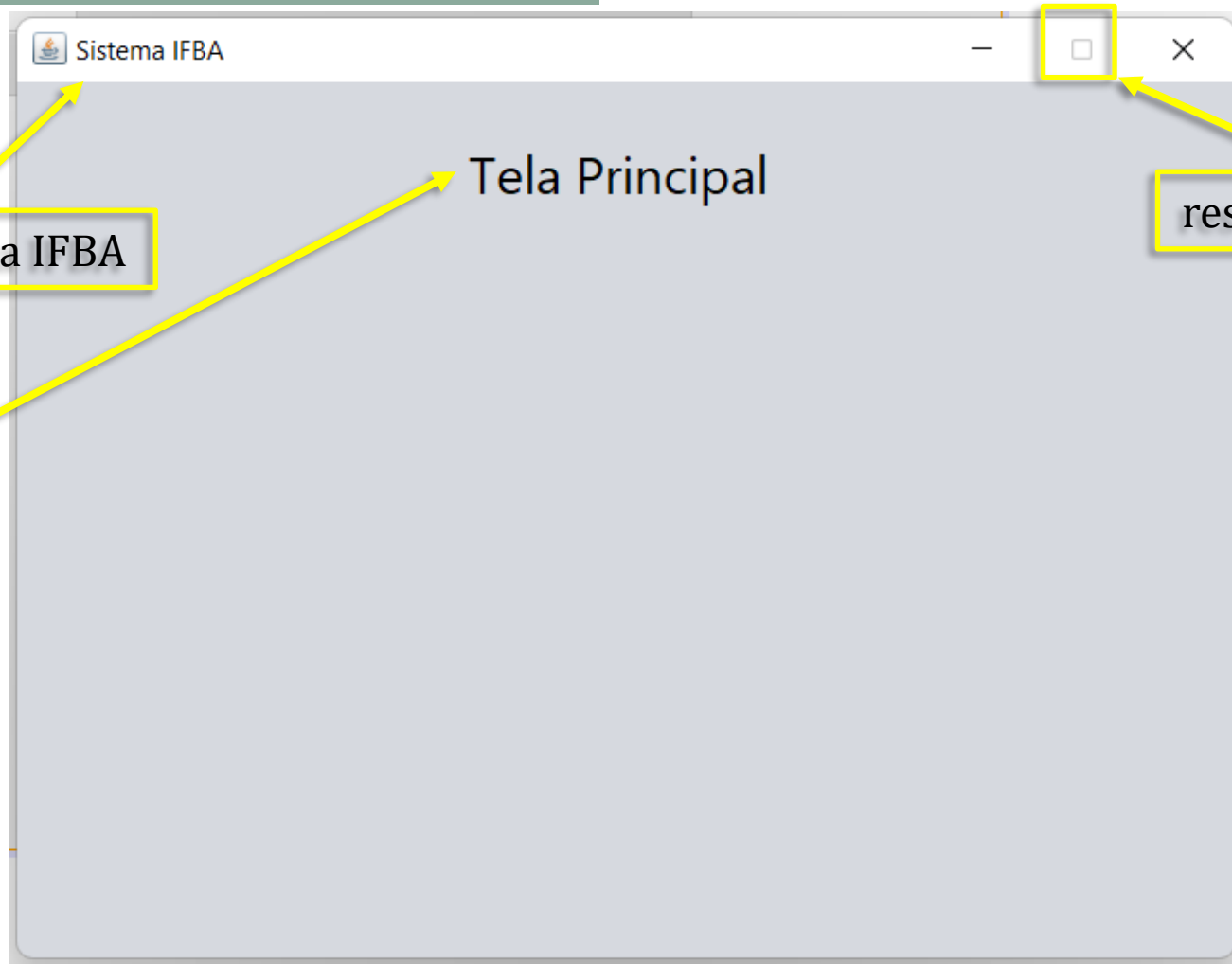
Java Swing – Exercício – Prog. Login



Java Swing – Exercício – Prog. Login

```
private void btnEntrarActionPerformed(java.awt.event.ActionEvent evt) {  
    //Lê caixa de texto e atribui resultado a variável  
    String usuario = txtUsuario.getText();  
    //Lê caixa de senha e (converte) atribui resultado a variável  
    String senha = String.valueOf(txpSenha.getPassword());  
  
    //abre nova janela (tela) se usuário e senha estiverem corretos  
    if(usuario.equals("ifba") && senha.equals("1234"))  
    {  
        //Cria um objeto, a tela Principal  
        TelaPrincipal telaPrincipal = new TelaPrincipal();  
        //Torna a tela Principal visível  
        telaPrincipal.setVisible(true);  
        //Descarta objeto (tela Login), "fecha"  
        this.dispose();  
    }  
    //Caso contrário  
    else  
    {  
        //Exibe caixa de diálogo com mensagem de acesso negado  
        JOptionPane.showMessageDialog(this, "Acesso negado!");  
    }  
}
```

Java Swing – Exercício – Prog. Principal



title: Sistema IFBA

lblTitulo

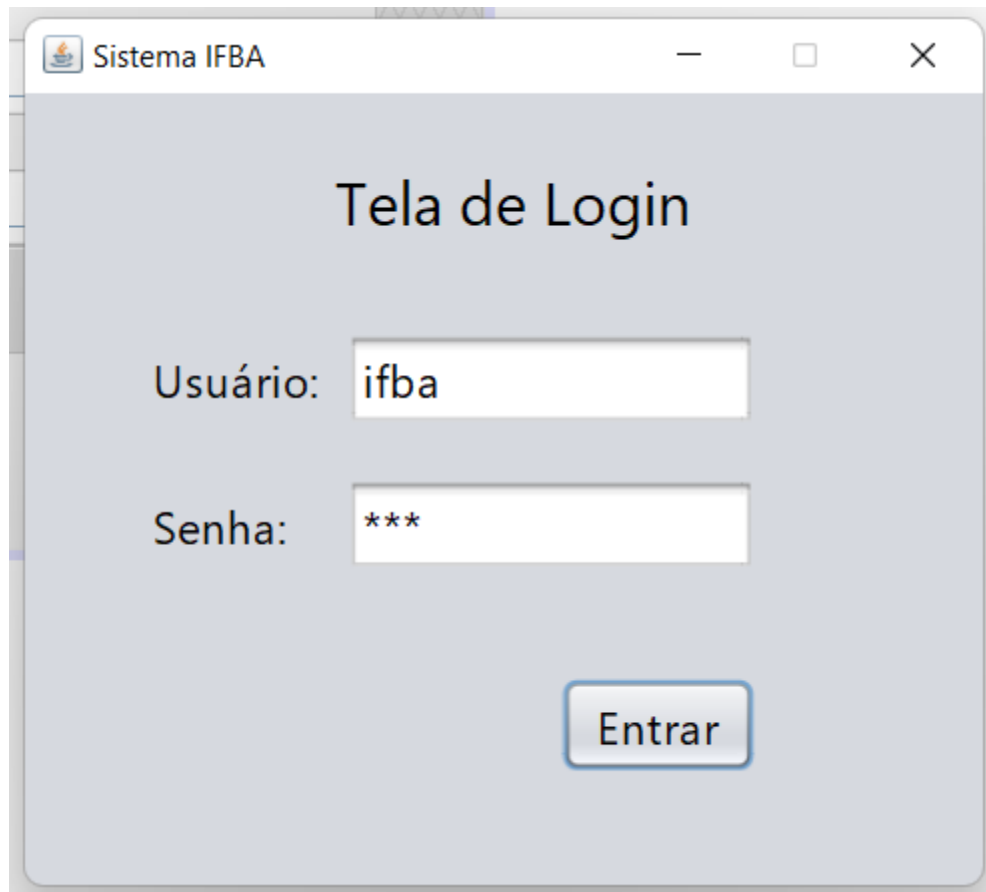
Tela Principal

resizable : false



Java Swing – Exercício – Prog. Login

Execução

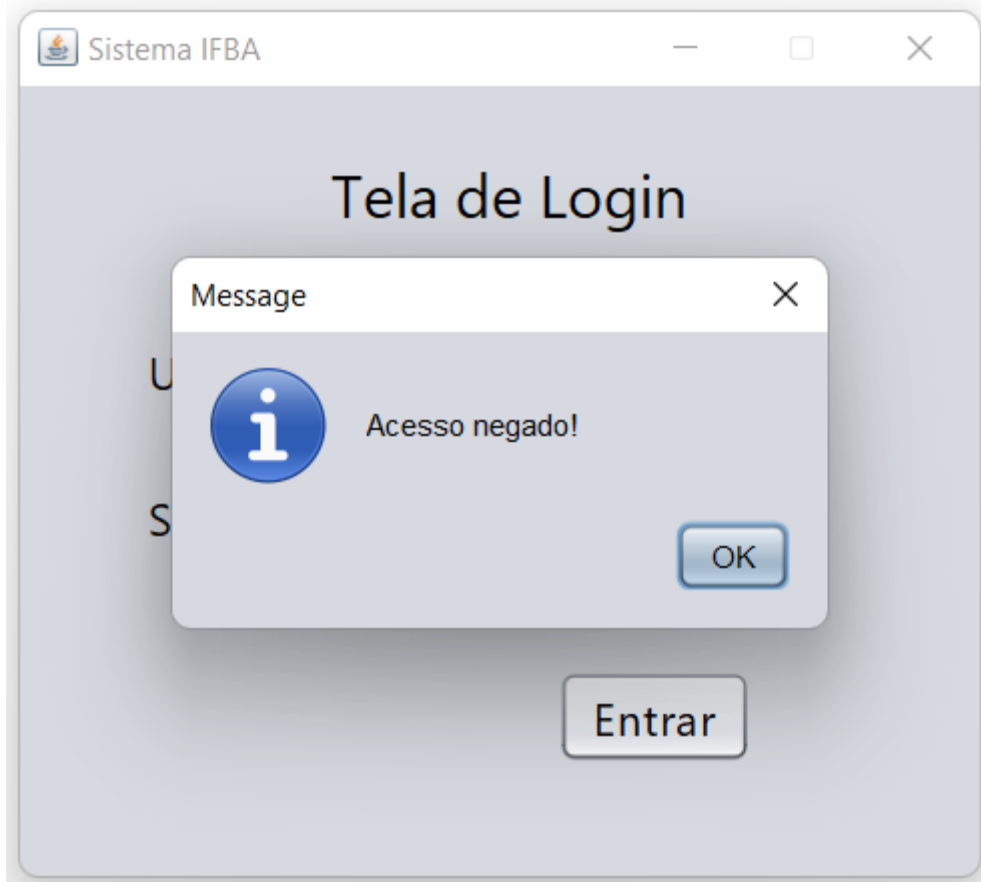


The screenshot shows a Java Swing window titled "Sistema IFBA". Inside the window, the text "Tela de Login" is centered. Below this, there are two input fields. The first is labeled "Usuário:" and contains the text "ifba". The second is labeled "Senha:" and contains three asterisks "***". At the bottom right of the window, there is a button labeled "Entrar".



Java Swing – Exercício – Prog. Login

Execução



Java Swing – Exercício – Prog. Login

Execução

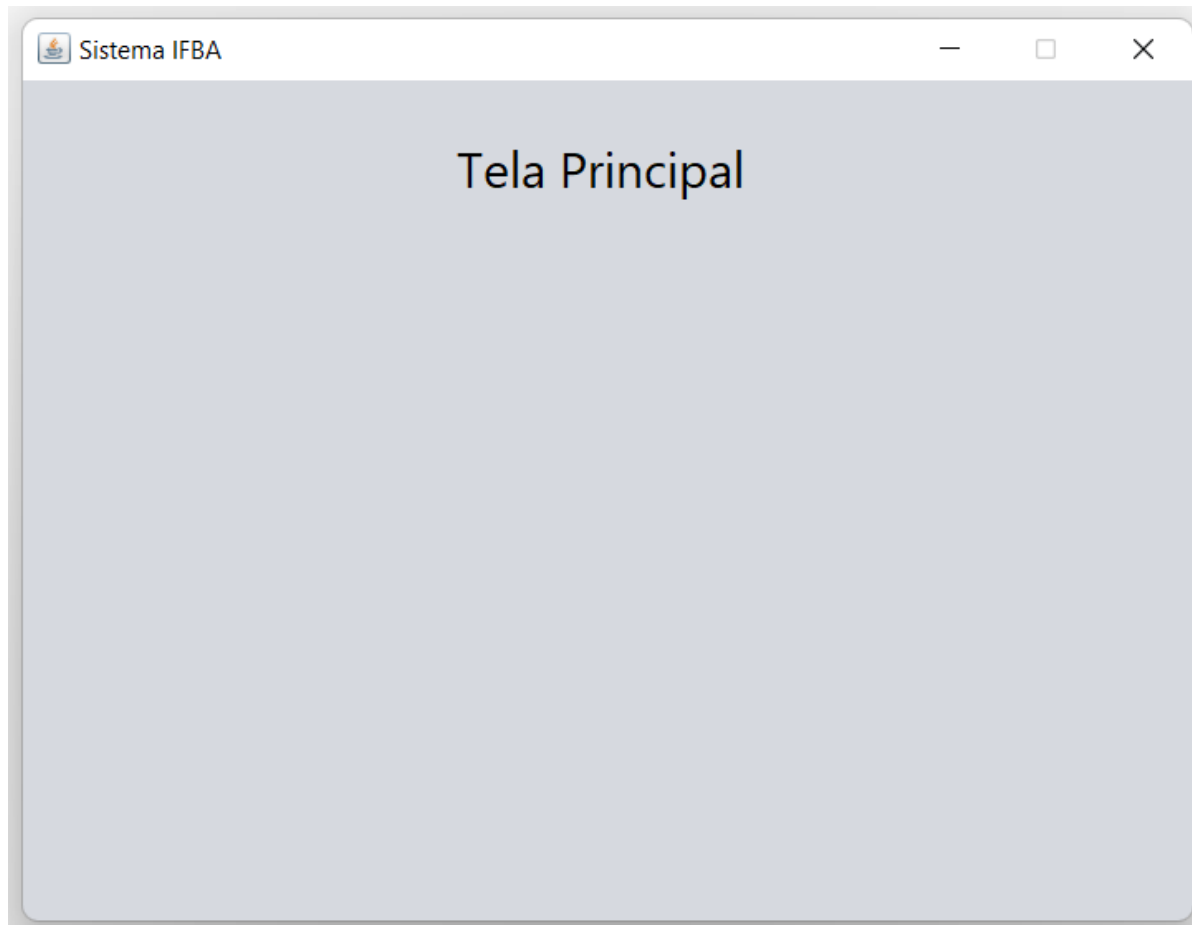


The screenshot shows a Java Swing window titled "Sistema IFBA" with a light gray background. The window contains the text "Tela de Login" centered at the top. Below this, there are two input fields: "Usuário:" followed by a text box containing "ifba", and "Senha:" followed by a password box containing four asterisks "****". At the bottom center of the window is a button labeled "Entrar". The window has standard OS window controls (minimize, maximize, close) in the title bar.



Java Swing – Exercício – Prog. Login

Execução



Referências

- Junior, Peter Jandl. Java Guia do Programador - 4ª Edição. Novatec Editora.
- SÉRGIO FURGERI. Java Ensino Didático: Desenvolvimento e Implementação de Aplicações. Editora Érica.
- <https://www.youtube.com/playlist?list=PLwQkYMetu0OYFOU71txhtvSYTVUCHjJYD>
- https://www.youtube.com/playlist?list=PLWd_VnthxxLfeEUK08gB7G3316OS5xIT3



Obrigado!

- Canais de Comunicação;
- Horário de Atendimento.

