

JAVA

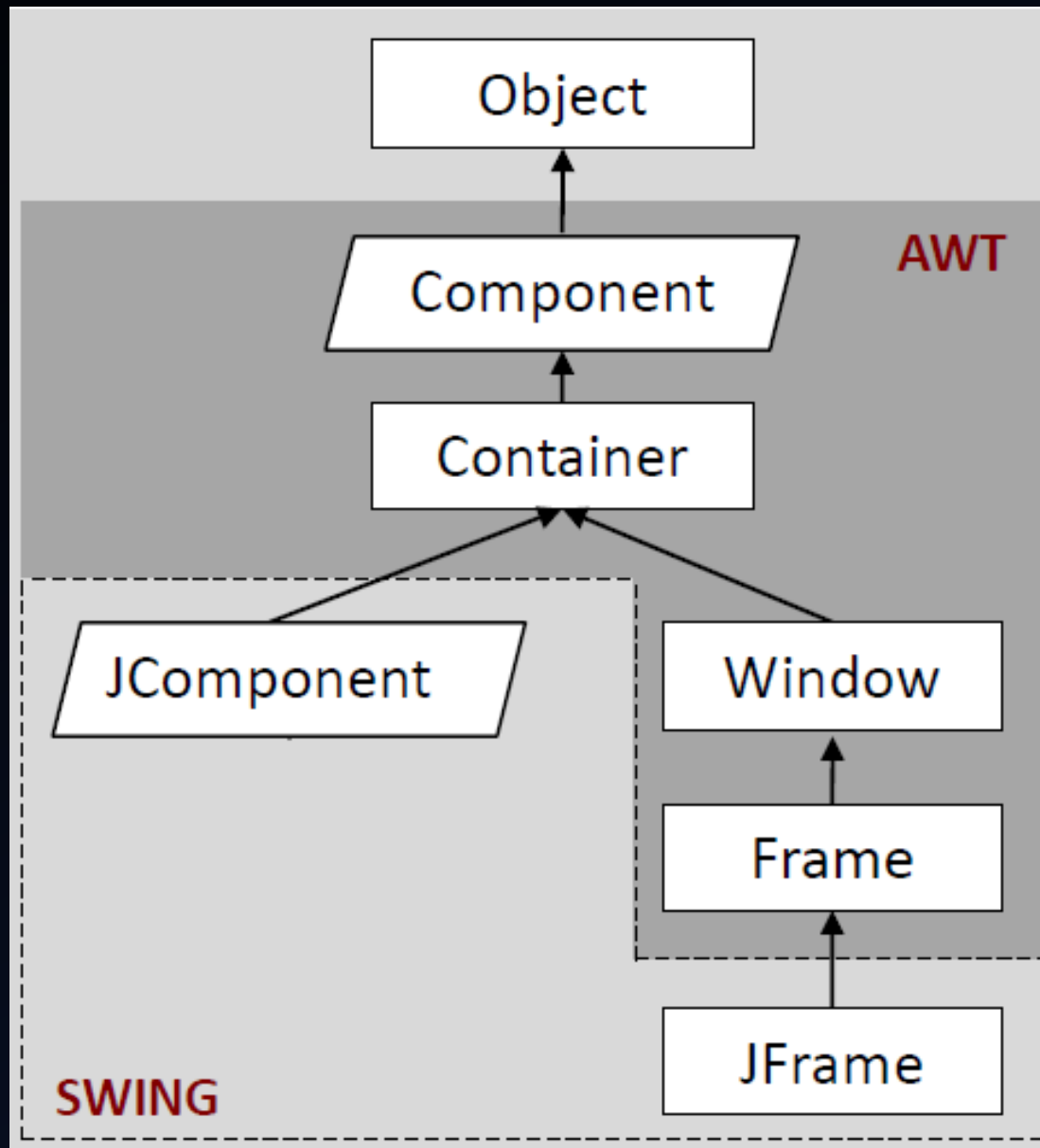
LINGUAGEM DE PROGRAMAÇÃO
INTERFACE GRÁFICA (GUI)

JFrame

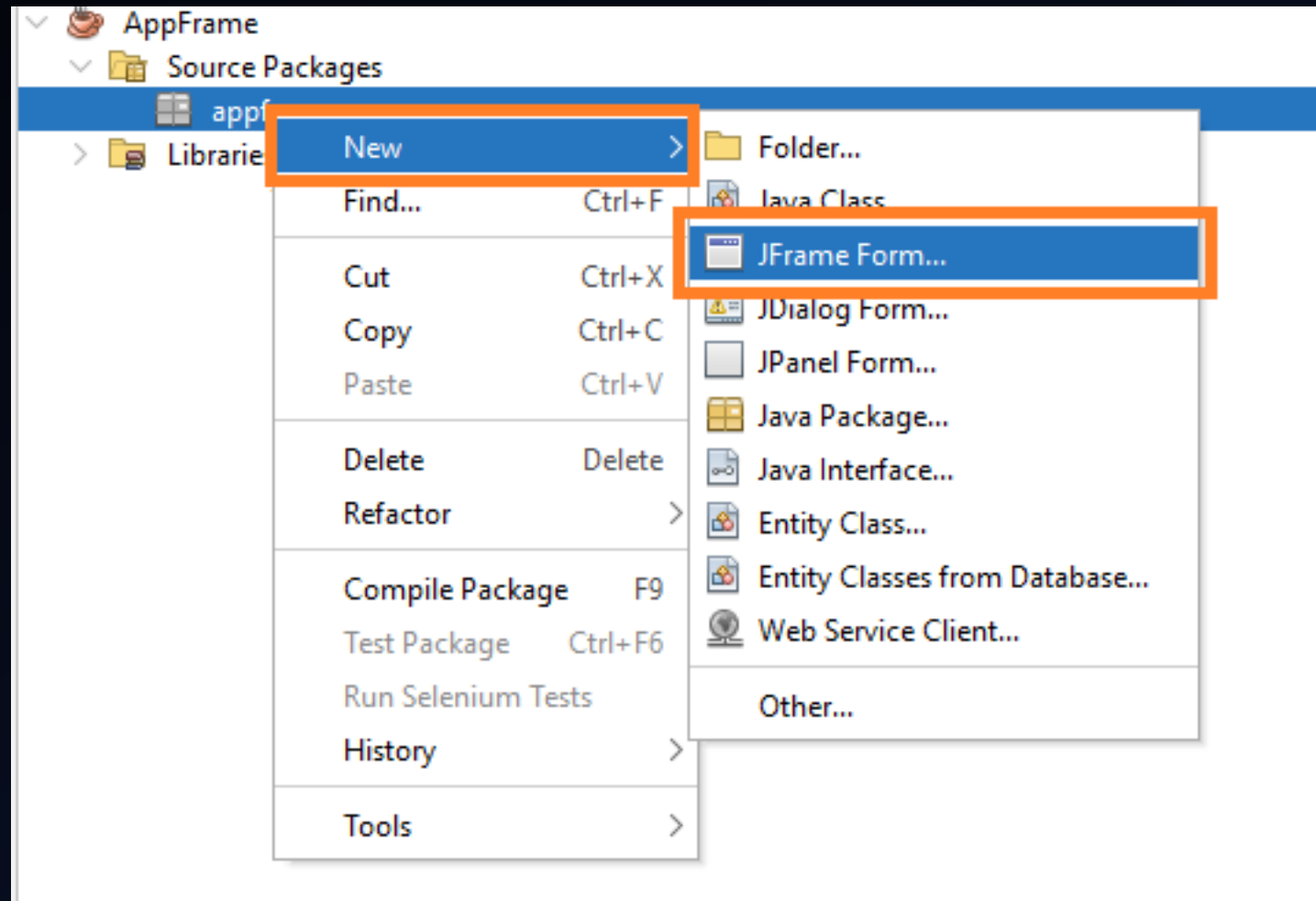
LINGUAGEM DE PROGRAMAÇÃO JAVA
INTERFACE GRÁFICA (GUI)

JFRAME

Como regra, um aplicativo independente com uma GUI baseada em Swing tem pelo menos uma hierarquia de contenção com a **JFRAME** como raiz.



Como criar um JFrame?



New JFrame Form



Steps

1. Choose File Type
2. **Name and Location**

Name and Location

Class Name:

NewJFrame

Project:

AppFrame

Location:

Source Packages

Package:

appframe

Created File:

C:\Users\Maicon\Documents\NetBeansProjects\AppFrame\src\appframe\New

Superclass:

Browse...

Interfaces:

Browse...

< Back

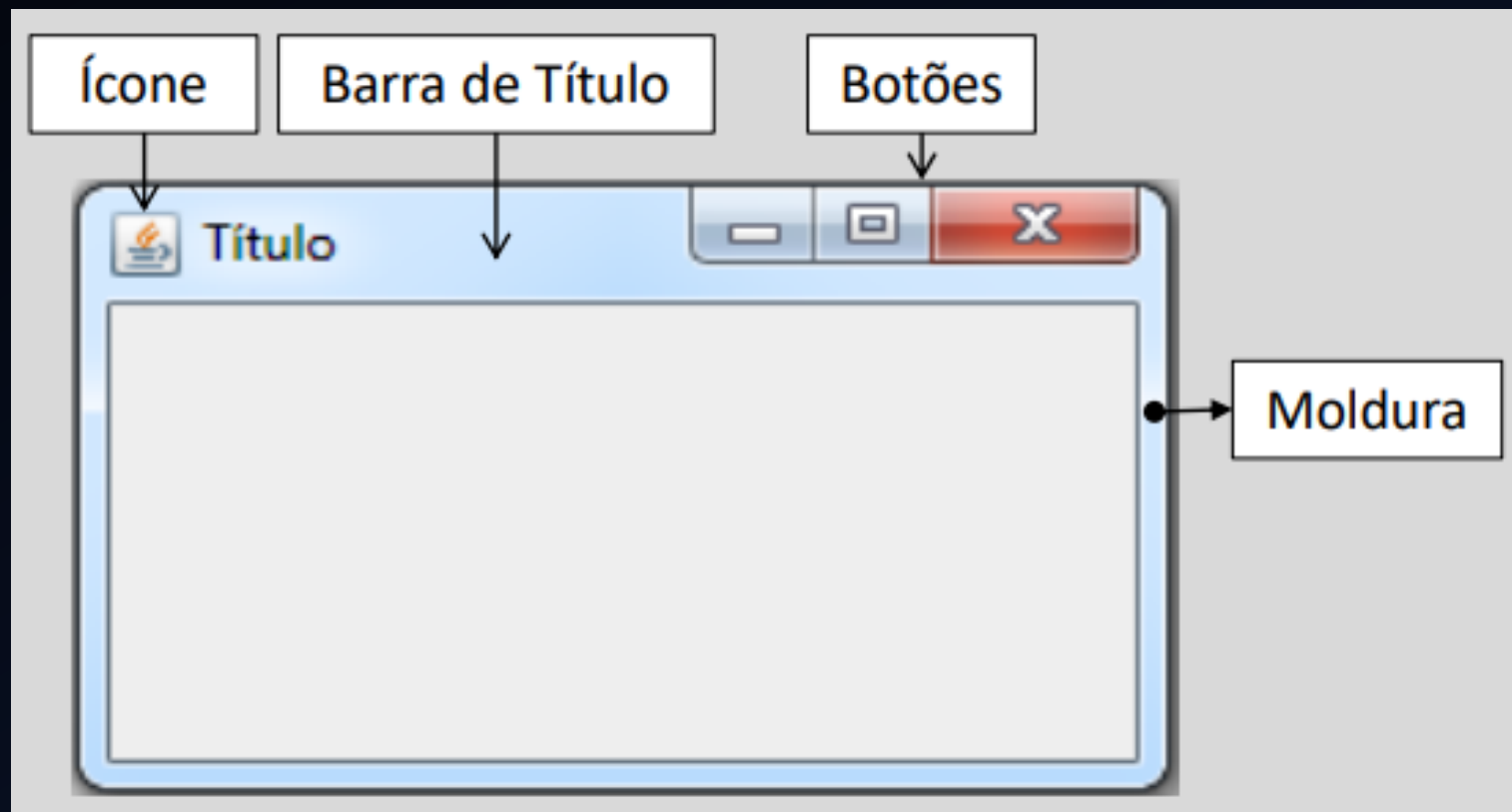
Next >

Finish

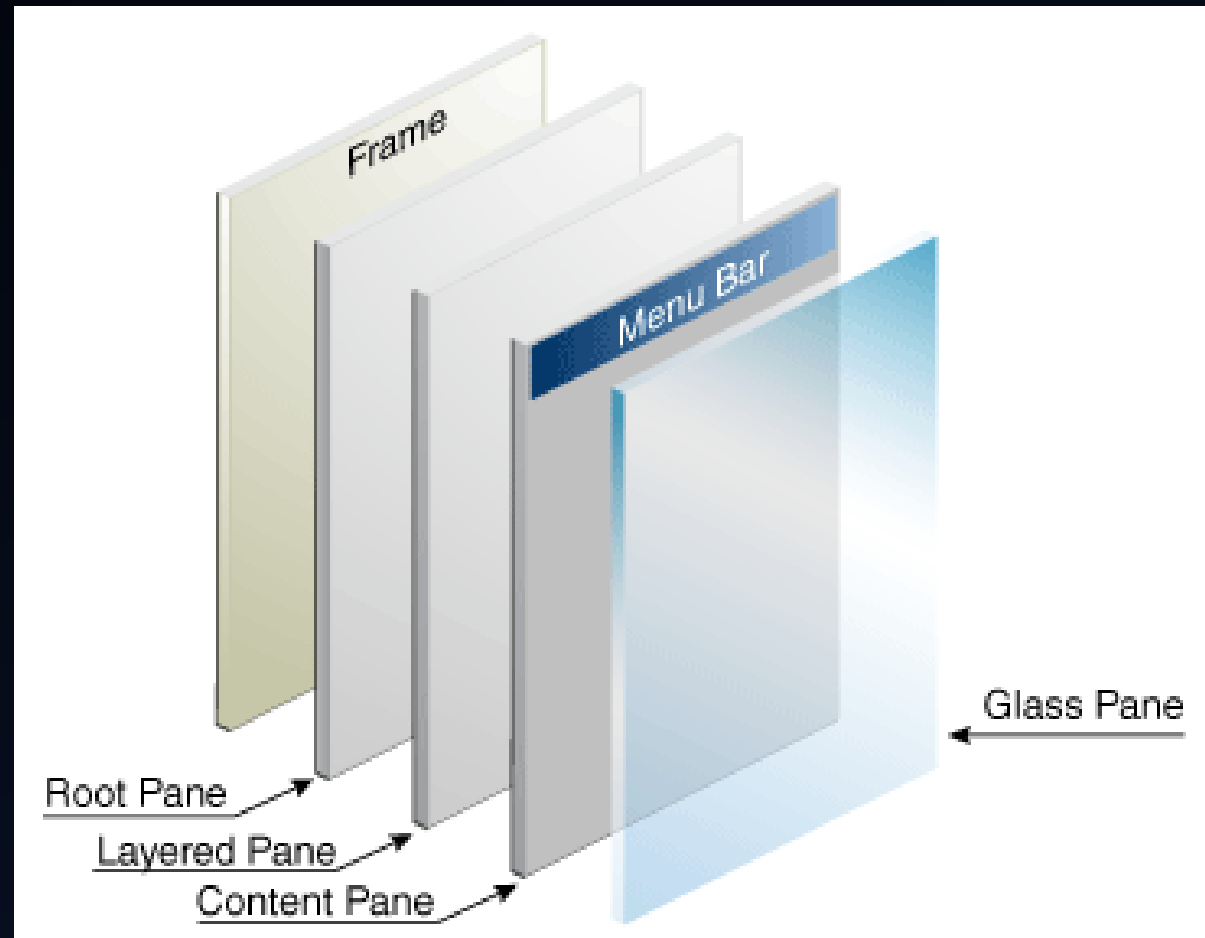
Cancel

Help

JFRAME



JFRAME



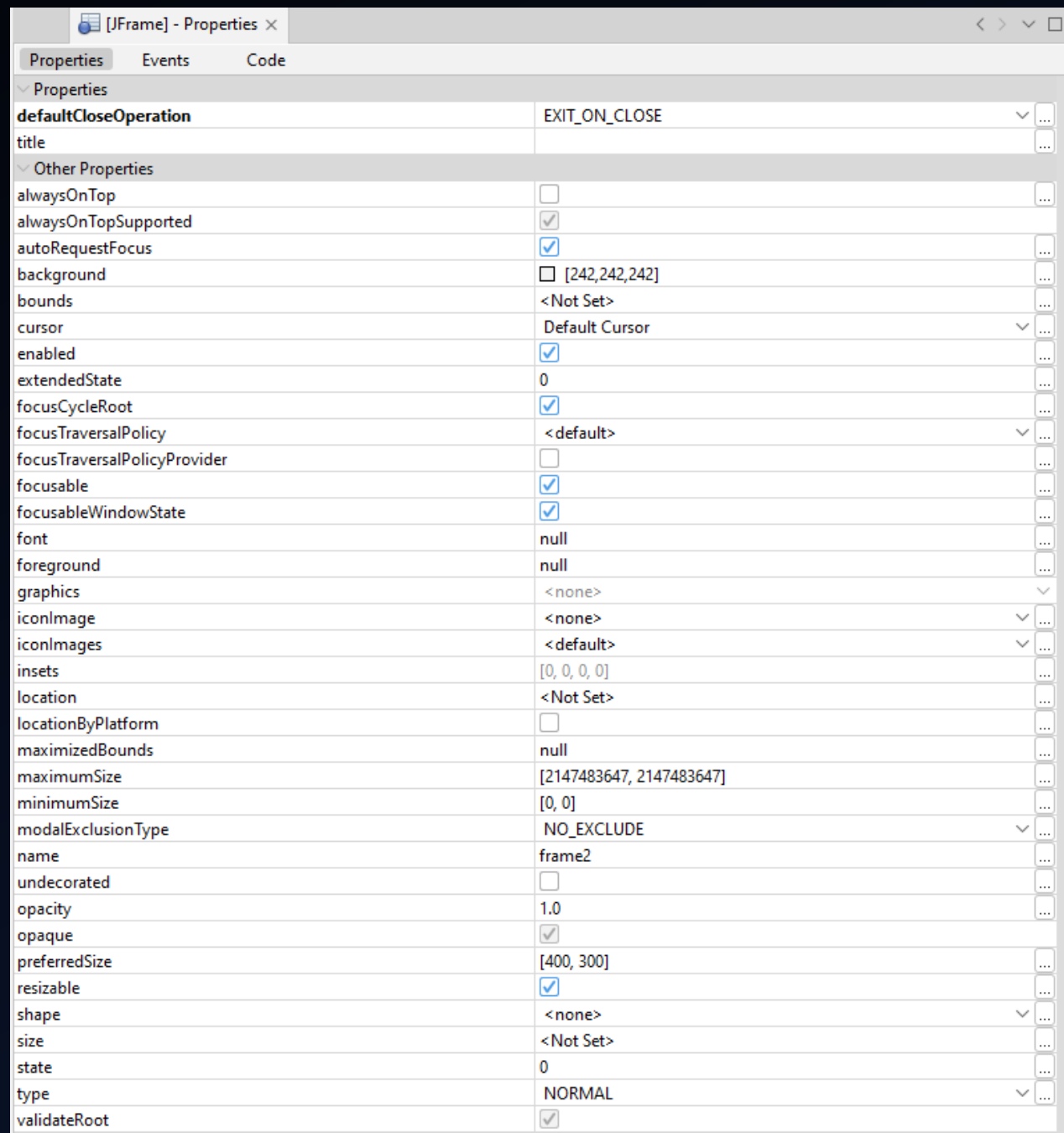
JFRAME

Classe **JFRAME**:

- A maioria das janelas é uma instância ou subclasse dessa classe.
- Fornece a barra de título.
- Já existe um container para ser utilizado.
- Fornece botões para minimizar, maximizar e fechar a aplicação.

JFRAME

Algumas
propriedades
da Classe
JFRAME



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

```
    public NewJFrame() {  
        initComponents();  
    }
```

```
@SuppressWarnings("unchecked")
```

```
Generated Code
```

```
public static void main(String args[]) {
```

```
    /* Set the Nimbus look and feel */
```

```
    Look and feel setting code (optional)
```

```
    /* Create and display the form */
```

```
    java.awt.EventQueue.invokeLater(new Runnable() {
```

```
        public void run() {
```

```
            new NewJFrame().setVisible(true);
```

```
        }
```

```
    });
```

```
}
```

```
// Variables declaration - do not modify
```

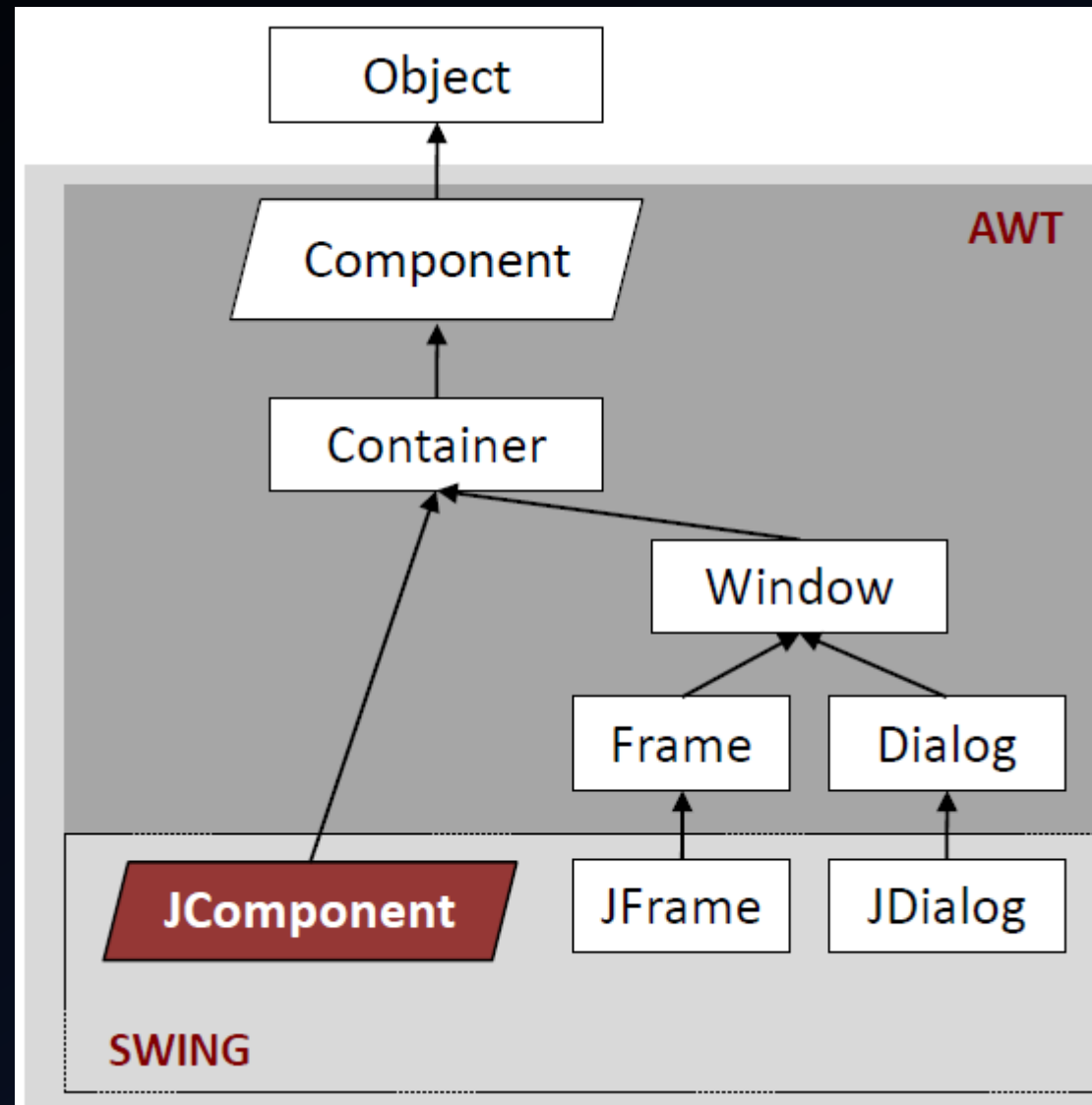
```
// End of variables declaration
```

```
}
```

NA PRÁTICA....

ATIVIDADE 1

- Crie um JFrame em um projeto Java e rode o programa.
- Após, adicione um título ao JFrame.



JComponent

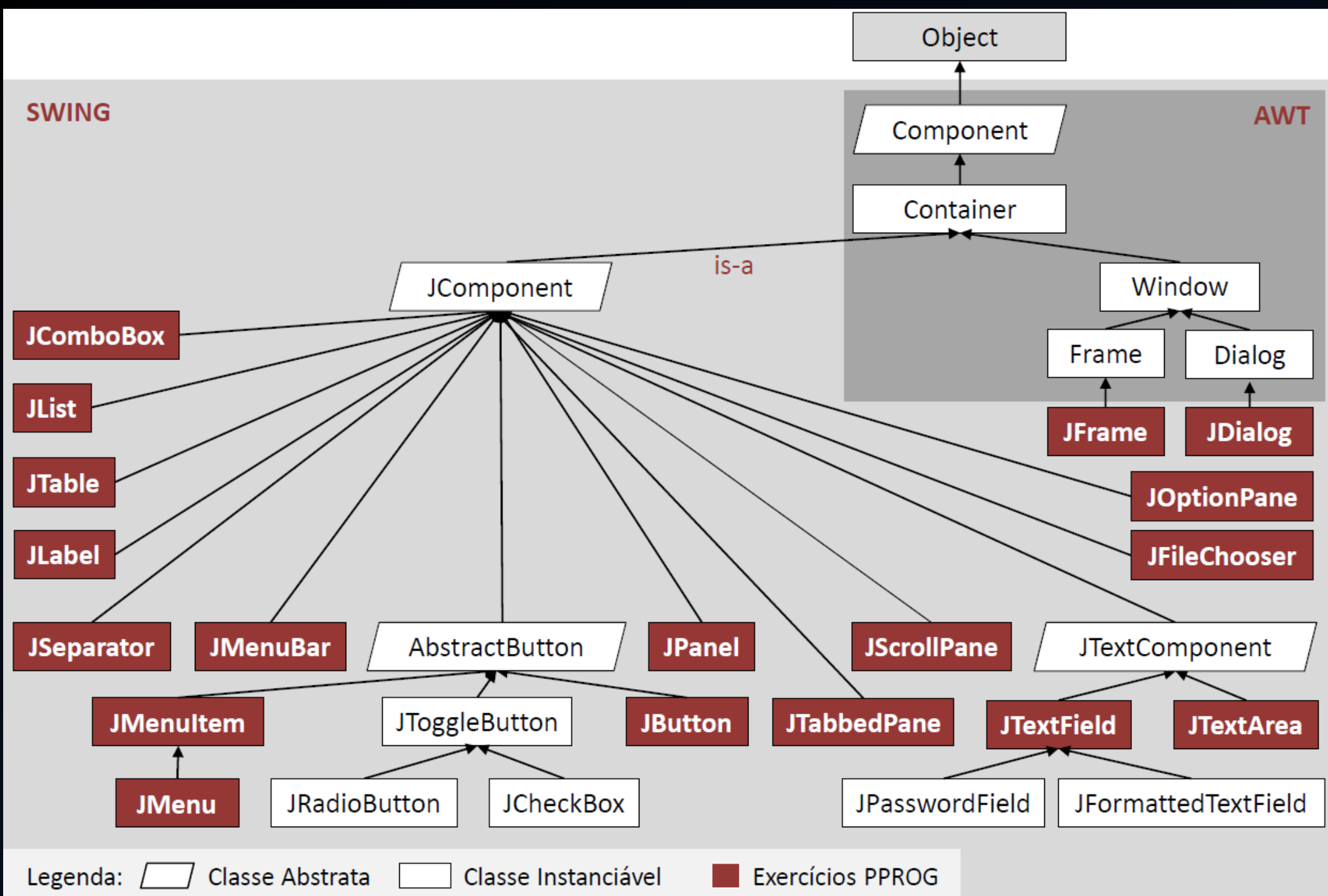
LINGUAGEM DE PROGRAMAÇÃO JAVA
INTERFACE GRÁFICA (GUI)

JComponent

Todos os componentes Swing cujos nomes começam com “J” descendem da classe **Jcomponent**.

A Classe **Jcomponent** estende do Container, que por sua vez estende aos objetos gráficos. As tabelas de API desta seção resumem os métodos mais usados de Componente Container, bem como de **JComponent**.

Mas quantas classes de
componentes existem???

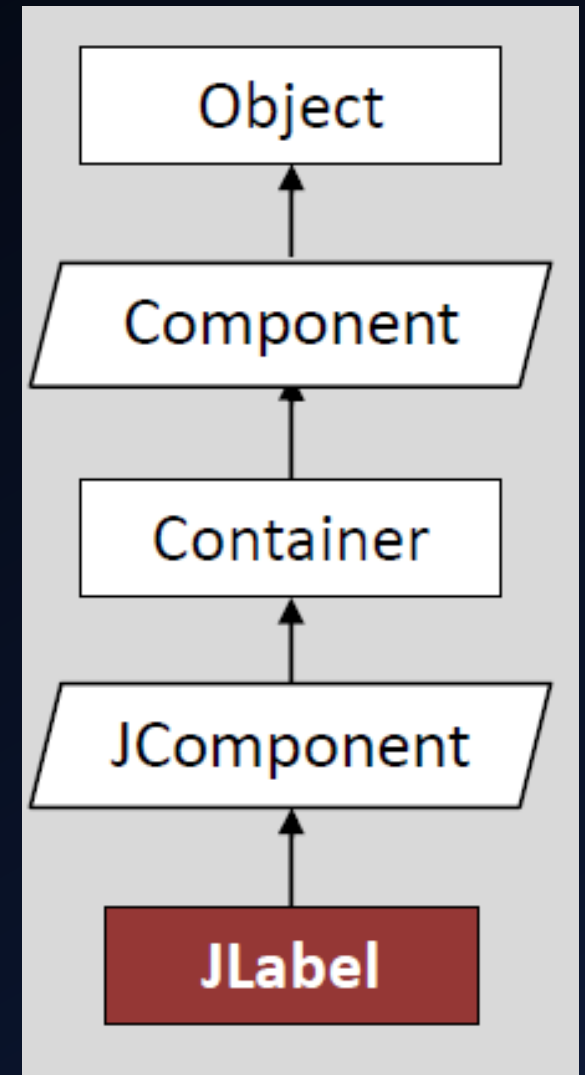


JLabel

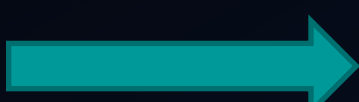
LINGUAGEM DE PROGRAMAÇÃO JAVA
INTERFACE GRÁFICA (GUI)

JLabel

Com a classe **JLabel**, podemos exibir texto e imagens. Se precisar criar um componente que exiba uma cadeia de caracteres, uma imagem ou ambos, poderá fazer isso usando o **JLabel**.



JLabel



```
JLabel label1, label2, label3;

ImageIcon icon = createImageIcon("images/middle.gif",
                                   "a pretty but meaningless splat");

//Create the first label.
label1 = new JLabel("Image and Text",
                    icon,
                    JLabel.CENTER);
//Set the position of its text, relative to its icon:
label1.setVerticalTextPosition(JLabel.BOTTOM);
label1.setHorizontalTextPosition(JLabel.CENTER);

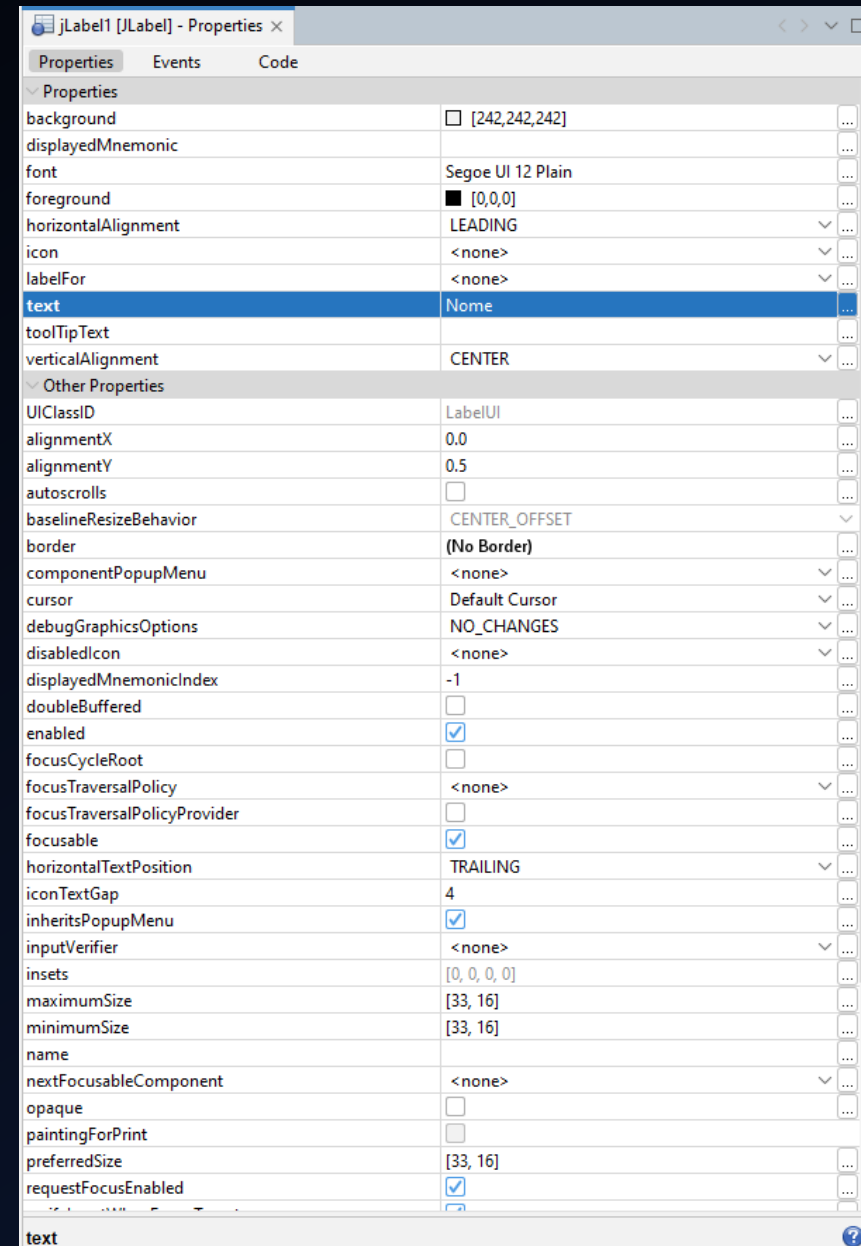
//Create the other labels.
label2 = new JLabel("Text-Only Label");
label3 = new JLabel(icon);

//Create tool tips, for the heck of it.
label1.setToolTipText("A label containing both image and text");
label2.setToolTipText("A label containing only text");
label3.setToolTipText("A label containing only an image");

//Add the labels.
add(label1);
add(label2);
add(label3);
```

JLabel

Algumas propriedades que podem ser modificadas e personalizadas para a Label (Texto personalizado).



Declaração	Funcional
<pre>public void setText(String text)</pre> <ul style="list-style-type: none">Ex: JLabel lblAgua = new JLabel(); lbl.setText("Água");	Especifica texto.
<pre>public String getText()</pre> <ul style="list-style-type: none">Ex: String texto = lbl.getText();	Retorna texto.
<pre>public void setIcon(Icon defaultIcon)</pre> <ul style="list-style-type: none">Ex: lbl.setIcon(new ImageIcon("agua.jpg"));	Especifica imagem.
<pre>public void setIconTextGap(int iconTextGap)</pre> <ul style="list-style-type: none">Ex: lbl.setIconTextGap(10);	Especifica intervalo entre

Declaração	Funcionalidade
<p>public void setVerticalAlignment(int alignment)</p> <ul style="list-style-type: none"> Exemplo 1: <pre>JLabel lblAguia = new JLabel("Águia"); lblAguia.setVerticalAlignment(SwingConstants.TOP);</pre> Exemplo 2: <pre>JLabel lblAguia = new JLabel("Águia"); lblAguia.setVerticalAlignment(JLabel.TOP);</pre> 	<p>Especifica alinhamento vertical do texto e imagem;</p> <p>Opções de alinhamento:</p> <ul style="list-style-type: none"> SwingConstants.CENTER SwingConstants.TOP SwingConstants.BOTTOM JLabel.CENTER JLabel.TOP JLabel.BOTTOM
<p>public void setHorizontalAlignment(int alignment)</p> <ul style="list-style-type: none"> Exemplo 1: <pre>JLabel lblAguia = new JLabel("Águia"); lblAguia.setHorizontalAlignment(SwingConstants.LEFT);</pre> Exemplo 2: <pre>JLabel lblAguia = new JLabel("Águia"); lblAguia.setHorizontalAlignment(JLabel.LEFT);</pre> 	<p>Especifica alinhamento horizontal do texto e imagem;</p> <p>Opções de alinhamento:</p> <ul style="list-style-type: none"> SwingConstants.CENTER SwingConstants.RIGHT SwingConstants.LEFT JLabel.CENTER JLabel.RIGHT JLabel.LEFT

NA PRÁTICA....

ATIVIDADE 2

- Crie um JLabel e adicione no JFrame em um projeto Java e rode o programa.
 - Após, adicione um novo texto ao JLabel.
- Agora, adicione um .icon ao JLabel e adicione um novo texto.

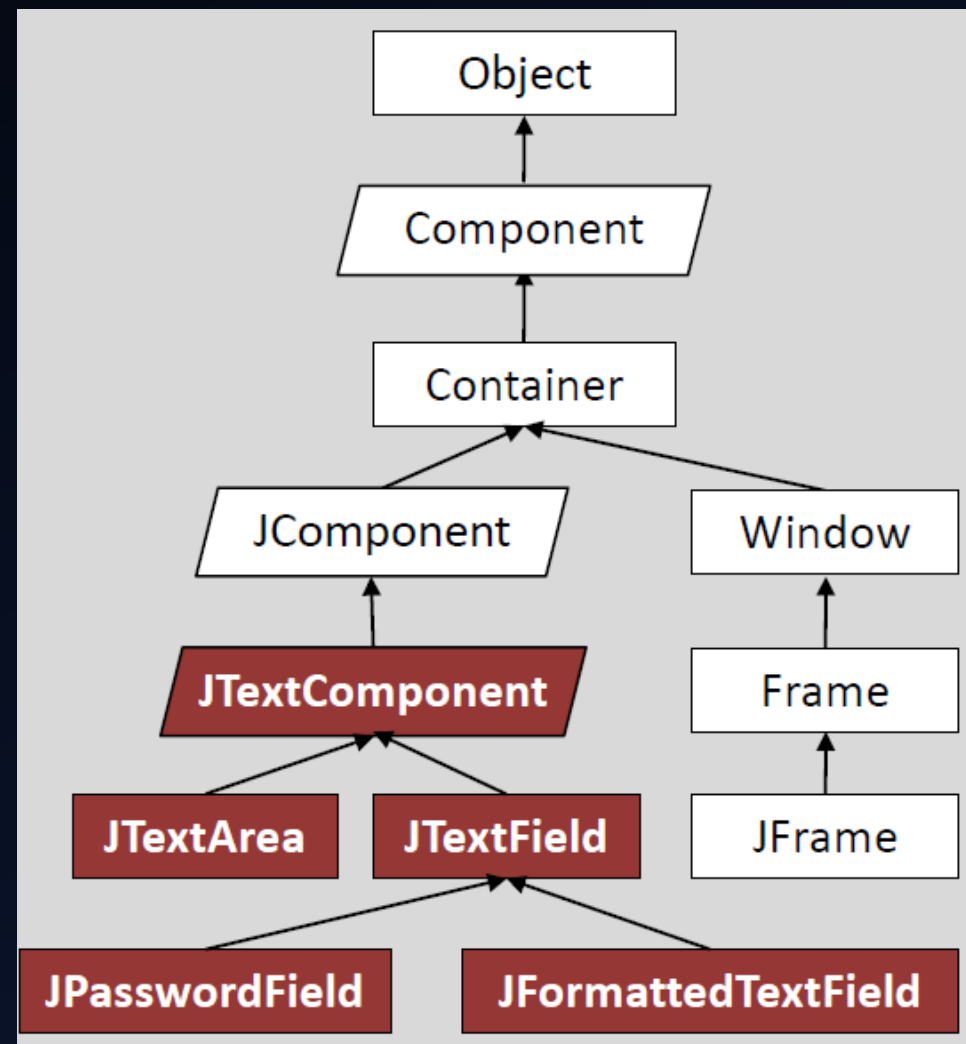
JTextComponent

LINGUAGEM DE PROGRAMAÇÃO JAVA
INTERFACE GRÁFICA (GUI)

JTextComponent

Um campo de texto é um controle de texto básico que permite ao usuário digitar uma pequena quantidade de texto. O campo de texto dispara um evento onde é possível tratar as informações.

A API do Swing fornece várias classes para componentes que são variedades de campos de texto ou que incluem campos de texto.



Tipos de TextComponent Swing

JTextField

- Aceita uma só linha de texto;

JTextArea

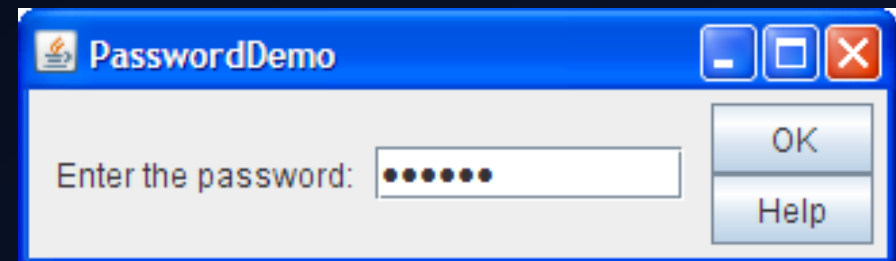
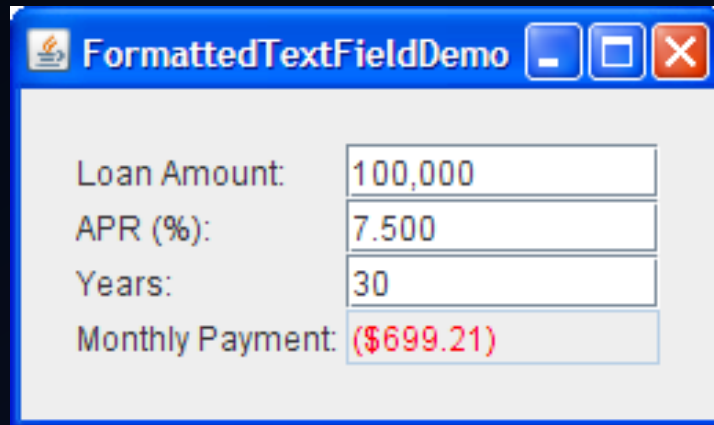
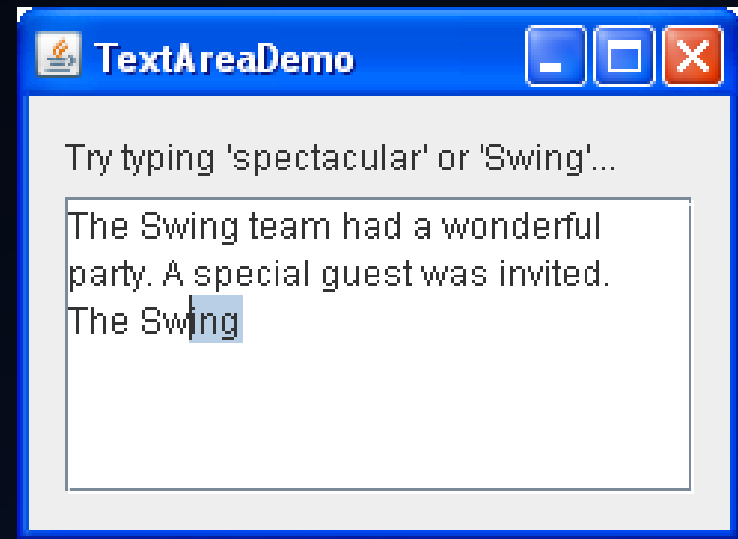
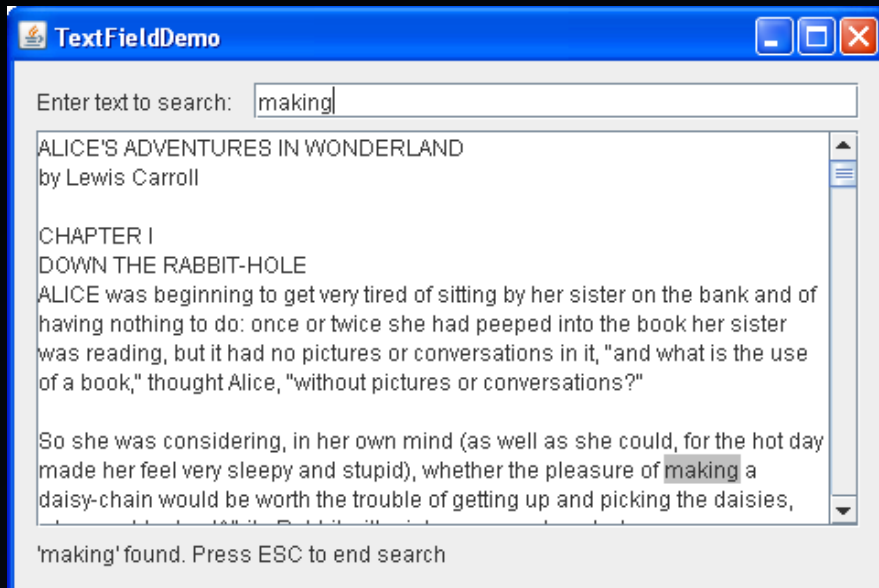
- Pode aceitar múltiplas linhas de texto;

JPasswordField

- Aceita só uma linha de texto, Esconde texto introduzido;

JFormattedTextField

- Para entradas específicas (Datas e Endereços IP);



Declaração	Funcionalidade
<code>public void setText(String t)</code>	Especifica texto t para o componente; <code>setText(null)</code> ou <code>setText("")</code> elimina texto do componente.
<code>public String getText()</code>	Retorna texto contido no componente.
<code>public void setEditable(boolean b)</code>	Especifica se o texto pode ou não ser editado.

Declaração	Funcionalidade
<code>public void setFont(Font f)</code>	Especifica tipo, estilo e tamanho das letras.
<code>public void setHorizontalAlignment(int alignment)</code>	Especifica alinhamento horizontal do texto; Opções: <ul style="list-style-type: none">▪ <code>TextField.LEFT</code> ou <code>TextField.LEADING</code>▪ <code>TextField.CENTER</code>▪ <code>TextField.RIGHT</code> ou <code>TextField.TRAILING</code>

Declaração	Funcionalidade
<code>public void setFont(Font f)</code>	Especifica tipo, estilo e tamanho das letras.
<code>public void setHorizontalAlignment(int alignment)</code>	Especifica alinhamento horizontal do texto; Opções: <ul style="list-style-type: none">▪ <code>TextField.LEFT</code> ou <code>TextField.LEADING</code>▪ <code>TextField.CENTER</code>▪ <code>TextField.RIGHT</code> ou <code>TextField.TRAILING</code>

NA PRÁTICA....

ATIVIDADE 3

- Crie dois JTextField (Nome e Sobrenome)
 - Crie um JTextArea (500 caracteres)
- Crie um JPasswordField (Senha tam. 10)
- Crie dois JFormattedTextField (Formatar cpf e cnpj)

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