

INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

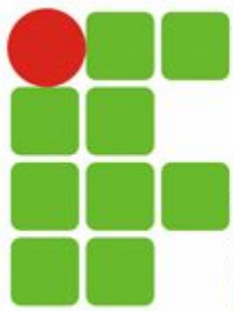
# PROGRAMAÇÃO ORIENTADA A OBJETOS

## Interface Gráfica

**Prof<sup>a</sup> Roberta B. Tôrres**

*Adaptação do Material de Prof<sup>a</sup> Giselle Teixeira de Almeida*

- ❑ Interface Gráfica em Java (GUI)
- ❑ Pacote java.awt
- ❑ Pacote javax.swing
- ❑ Principais Componentes de uma GUI
- ❑ Exercícios

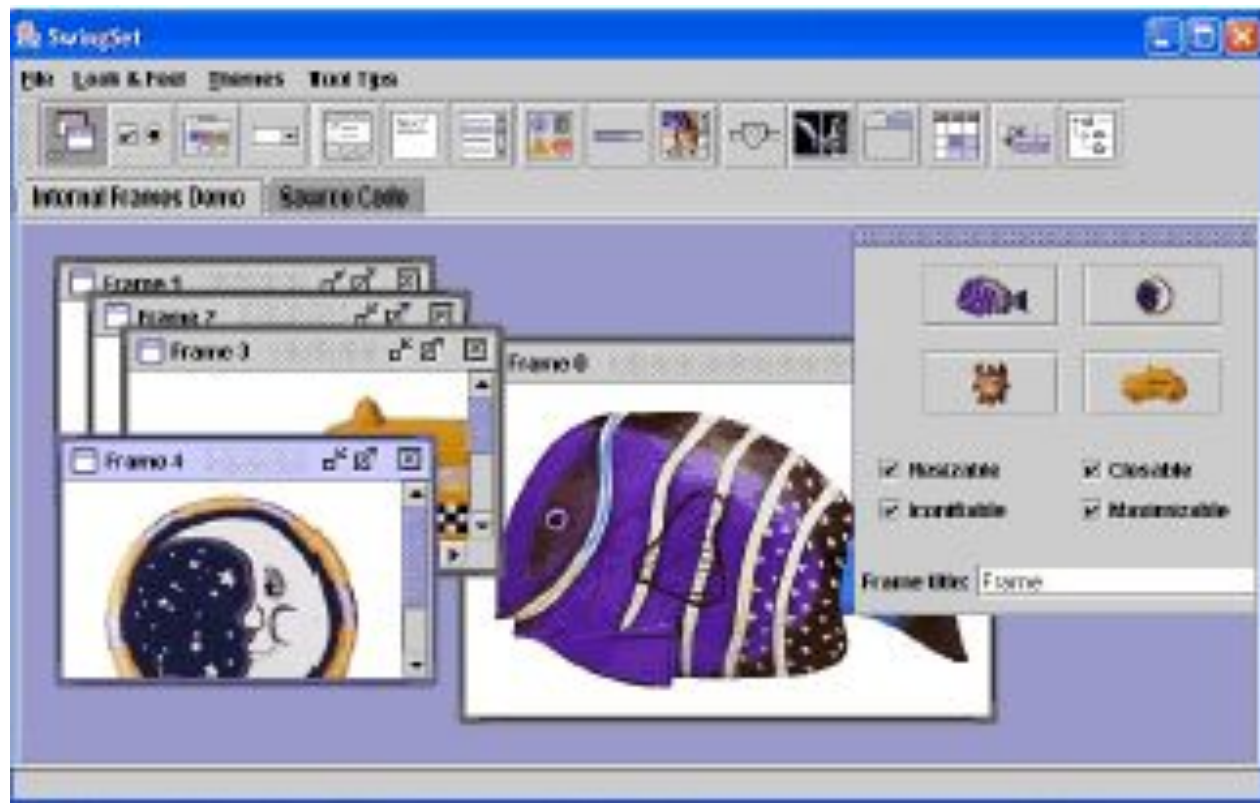


INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# INTERFACE GRÁFICA EM JAVA

## □ Considerações Gerais

A Interface Gráfica construída em Java é conhecida como **GUI** (*Graphical User Interface* - Interface Gráfica do Usuário).

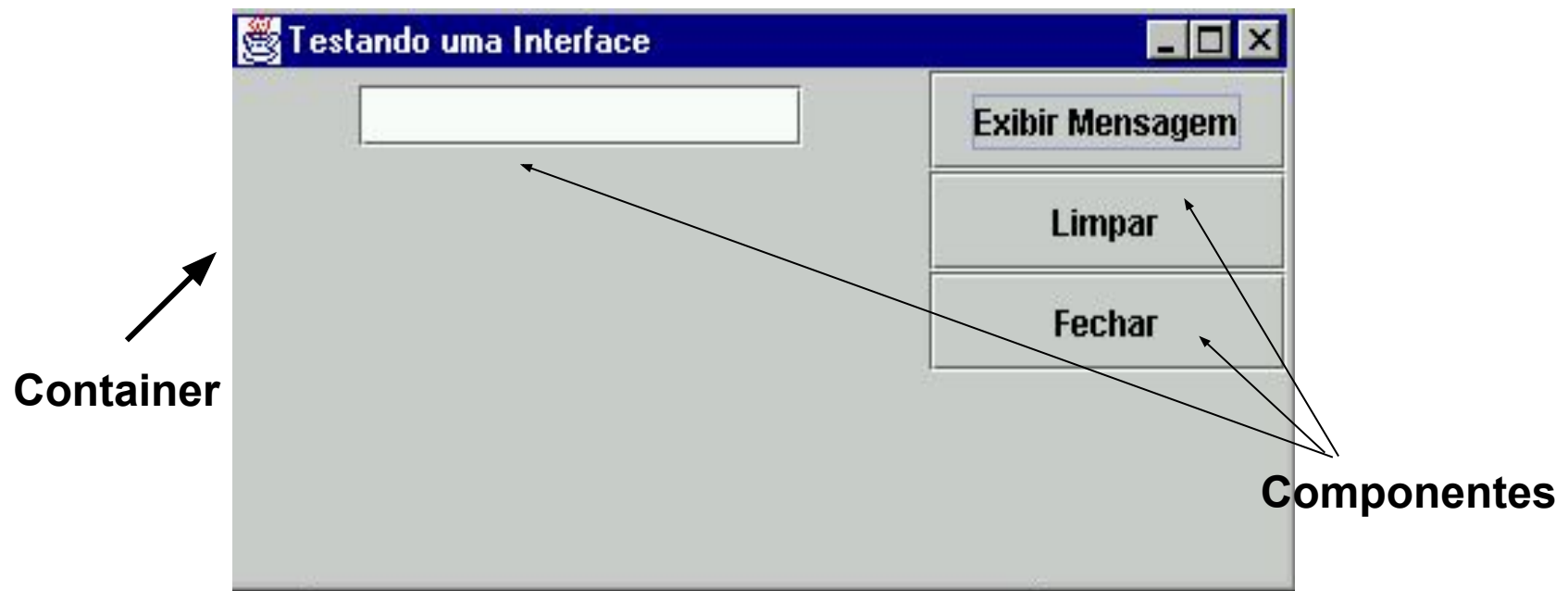


# INTERFACE GRÁFICA EM JAVA

## □ Considerações Gerais

Uma **GUI** é basicamente composta por: componentes, container, layout e eventos:

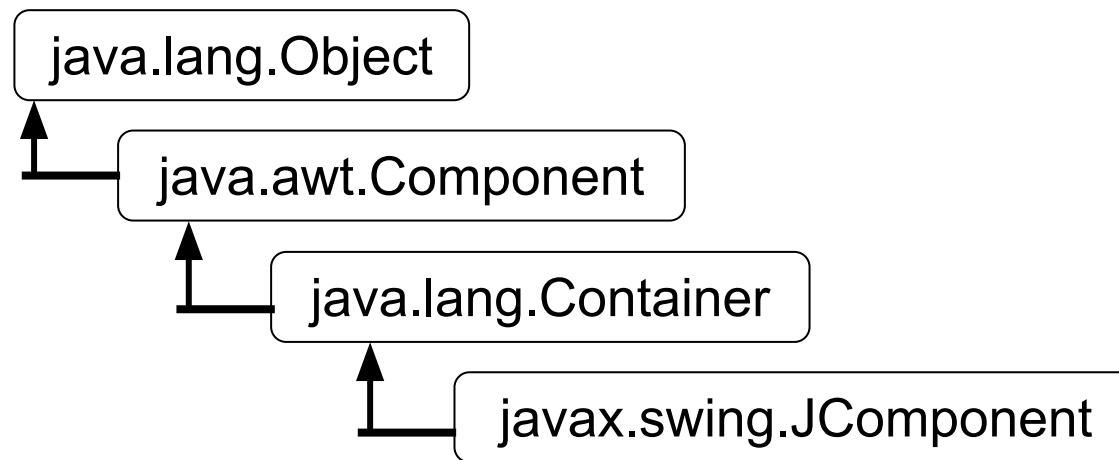
- **Componentes:** elementos como botões, caixas de texto, rótulos, menus etc.
- **Container:** representa uma coleção de componentes relacionados.
- **Layout:** auxilia na disposição e organização dos componentes.
- **Eventos:** são aplicados sobre os componentes.



# INTERFACE GRÁFICA EM JAVA

## □ Considerações Gerais

- Em Java, existem duas bibliotecas de classes principais que permitem a criação e manipulação de uma **GUI**: Pacote **java.awt** e Pacote **javax.swing**.
- Ao construir uma GUI, é preciso importar os pacotes `java.awt` e `javax.swing`.

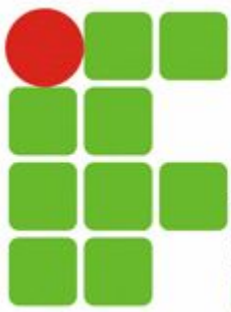


## □ Considerações Gerais

1. AWT significa ***Abstract Windows Toolkit*** (kit de ferramentas de janelas abstratas).
2. Oferece a infraestrutura mínima para lidar com uma GUI.
3. Permite especificar a localização e comportamento dos componentes, sendo adaptável a qualquer plataforma.
4. Apresenta limitações em aplicações com uma GUI mais elaborada, dependendo dos recursos gráficos da plataforma.
5. Insatisfação dos projetistas que diziam “escrever uma vez, depurar em todo lugar”.

## □ Considerações Gerais

1. Swing é o nome dado à coleção de componentes.
2. O Pacote javax.swing é mais robusto e tem mais recursos.
3. Não é um substituto completo do pacote AWT, apenas fornece uma coleção de componentes gráficos mais ricos.
4. Faz parte do JCF (Java Foundation Classes), que oferece uma variedade de API's.



# GUI

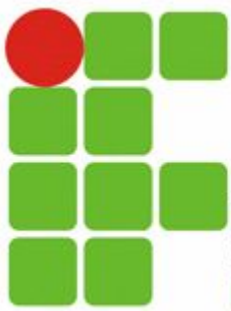
## PRINCIPAIS COMPONENTES

- Principais Componentes de uma GUI
  - JOptionPane
  - JFrame
  - ContentPane
  - JPanel
  - JLabel
  - JTextField
  - JPasswordField
  - JButton

---

Execute o código-fonte ***JanelaParte01.java*** para visualizar os principais componentes de uma GUI.





## ❏ Caixa de Diálogo - Componente JOptionPane

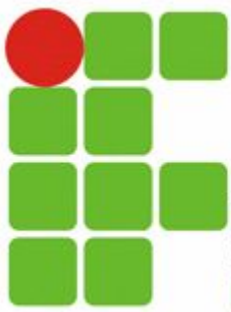
- Abra o arquivo ***03\_ExplicaçãoSimples\_JOptionPane.pdf***
- Execute o código-fonte ***TesteJOptionPane.java*** para ver um exemplo.

---

```
import javax.swing.JOptionPane;
public class TesteJanela {
    @SuppressWarnings("unused")
    public static void main(String[] args) {
        String num1 = JOptionPane.showInputDialog("Digite o Primeiro Número:");
        String num2 = JOptionPane.showInputDialog("Digite o Segundo Número:");

        int n1 = Integer.parseInt(num1);
        int n2 = Integer.parseInt(num2);
        int soma = n1 + n2;
        JOptionPane.showMessageDialog(null,"Primeiro Número: "+num1+" - Segundo
Número: "+num2,"Soma de Números",JOptionPane.INFORMATION_MESSAGE);

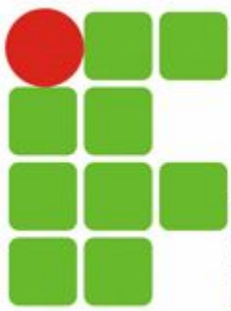
        JOptionPane.showMessageDialog(null,"Soma dos Números: "+soma,"Tela de
Resultados",JOptionPane.PLAIN_MESSAGE); } }
```



## ▣ Definição

1. É um **container** que serve de base para qualquer aplicação gráfica, freqüentemente utilizado para criar aplicativos baseados em GUI.
2. É uma **janela de nível mais alto**, sendo chamada de quadro (**frame**) em Java e pode conter outros componentes gráficos.
3. Fornece o espaço em que a GUI do Aplicativo é construída.
4. É um componente do Pacote javax.swing.
5. Criação de uma janela

***public class Janela extends JFrame***



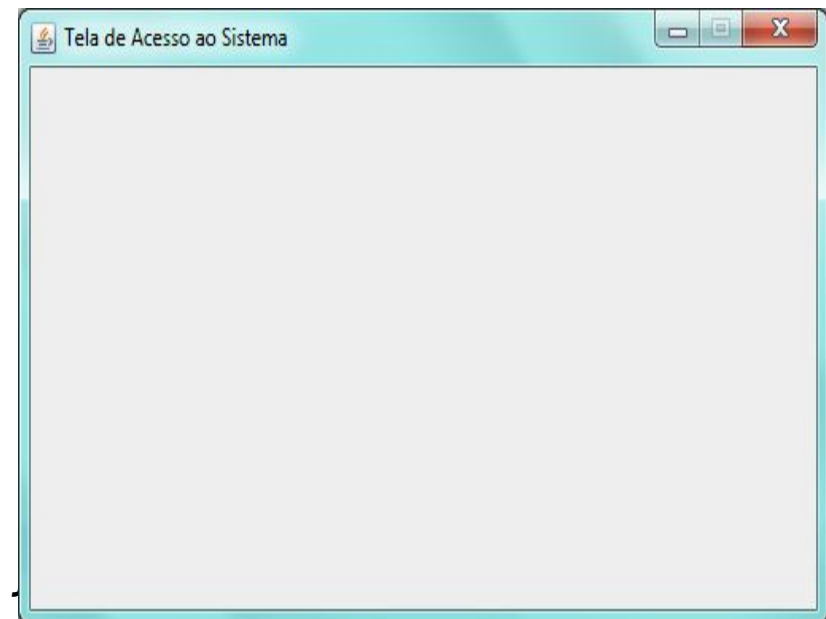
## ❑ JFrame – Código Exemplo:

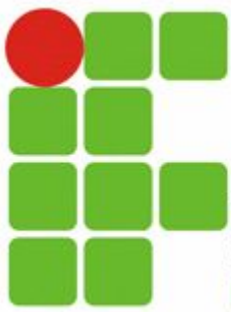
### - Criação da classe Janela

```
package testeJFrame;  
import javax.swing.JFrame;  
public class Janela extends JFrame  
{  
    public Janela()    //construtor da classe  
    {  
        this.setTitle("Tela de Acesso ao Sistema");  
        this.setSize(500,300);  
        this.setVisible(true);  
        this.setResizable(false);  
    }  
}
```

### - Instanciando a classe Janela

```
package testeJFrame;  
public class TesteJanela  
{  
    public static void main(String[] args)  
    {  
        Janela objTela = new Janela();  
    }  
}
```

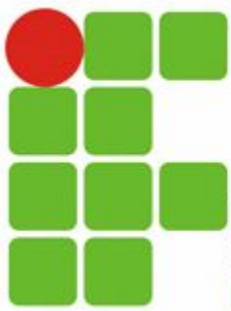




## □ Definição

1. Para acrescentar objetos em um **JFrame**, é preciso obter uma interface opaca chamada **ContentPane** (painel de conteúdo).
2. O **ContentPane** é um painel que cobre a área útil de **JFrame**.
3. O layout é definido no **ContentPane**.
4. Os componentes são adicionados ao **ContentPane**.
5. Cores e fontes devem ser definidas a partir do **ContentPane**.
6. Para obter o **ContentPane** do Frame use:

***Container c = this.getContentPane();***

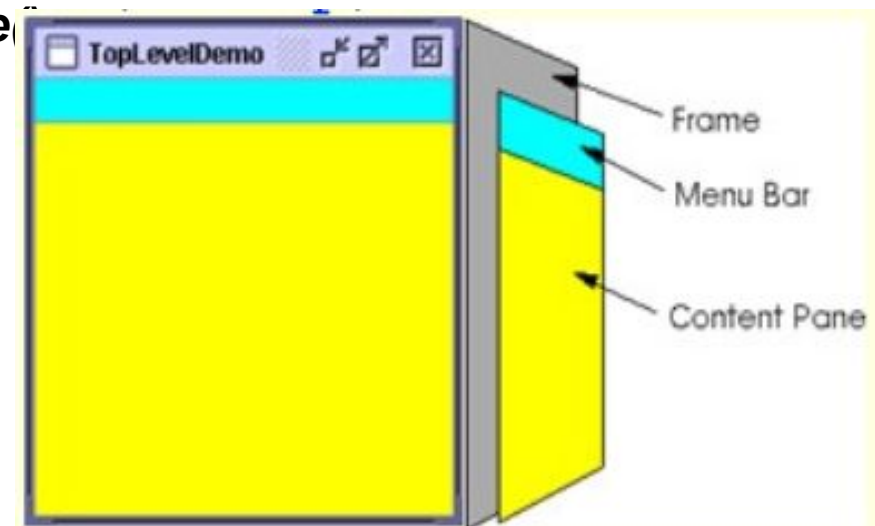


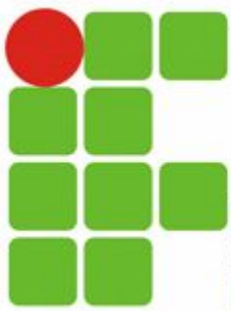
# ContentPane

## ContentPane – Código Exemplo:

```
package testeJFrame;  
import java.awt.Container;  
import javax.swing.JFrame;  
public class Janela extends JFrame  
{  
    public Janela()  
    {  
        this.setTitle("Tela de Acesso ao Sistema");  
        this.setSize(500,300);  
        this.setVisible(true);  
        this.setResizable(false);  
        Container c = this.getContentPane();  
    }  
}
```

```
package testeJFrame;  
public class TesteJanela  
{  
    public static void main(String[] args)  
    {  
        Janela objTela = new Janela();  
    }  
}
```

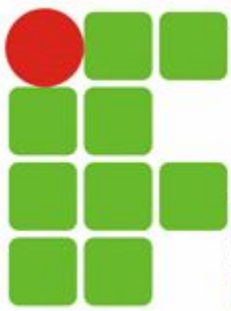




## □ Definição:

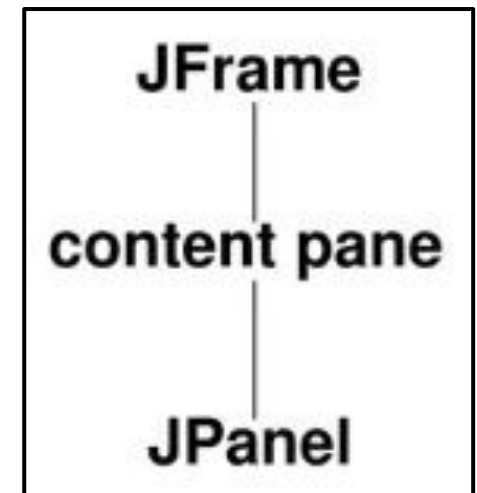
1. É um container de propósito geral.
2. Serve para agrupar outros componentes e permitir layout em camadas.
3. Facilita o posicionamento dos componentes.
4. Para adicionar um painel (JPanel) a um quadro (JFrame) usa-se:

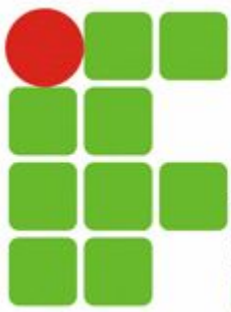
```
Container c = this.getContentPane();  
JPanel painel = new JPanel();  
c.add(painel);
```



## ❑ JPanel – Código Exemplo:

```
package testeJFrame;  
import java.awt.Container;  
import javax.swing.*.*;  
public class Janela extends JFrame  
{    public Janela()  
    {    this.setTitle("Tela de Acesso ao Sistema");  
        this.setSize(500,300);  
        this.setVisible(true);  
        this.setResizable(false);  
        Container c = this.getContentPane();  
        JPanel painel = new JPanel();  
        c.add(painel);  
    }  
}
```





## □ Definição

1. Representa um rótulo, isto é, um componente que contém uma linha de texto simples, não editável.
2. Usada para fornecer instruções de texto ou informações sobre uma GUI.
3. Exibe uma única linha de texto de leitura (os programas raramente alteram o conteúdo do rótulo).
4. Não reage à ações do usuário.
5. Instância de um JLabel

*JLabel rotulo1 = new JLabel();*



## □ Definição (continuação):

6. Adicionar Texto a um rótulo:

```
JLabel rotulo1 = new JLabel ("Informe o nome do usuário: ");
```

*ou*

```
rotulo1.setText("Informe o nome do usuário:");
```

7. Método `setToolTipText`: usado para apresentar textos de dica, quando o mouse é posicionado sobre o rótulo.

```
rotulo1.setToolTipText("username");
```

8. Adicionando o JLabel ao JFrame:

```
painel.add(rotulo1);
```



## ❑ JLabel – Código Exemplo:

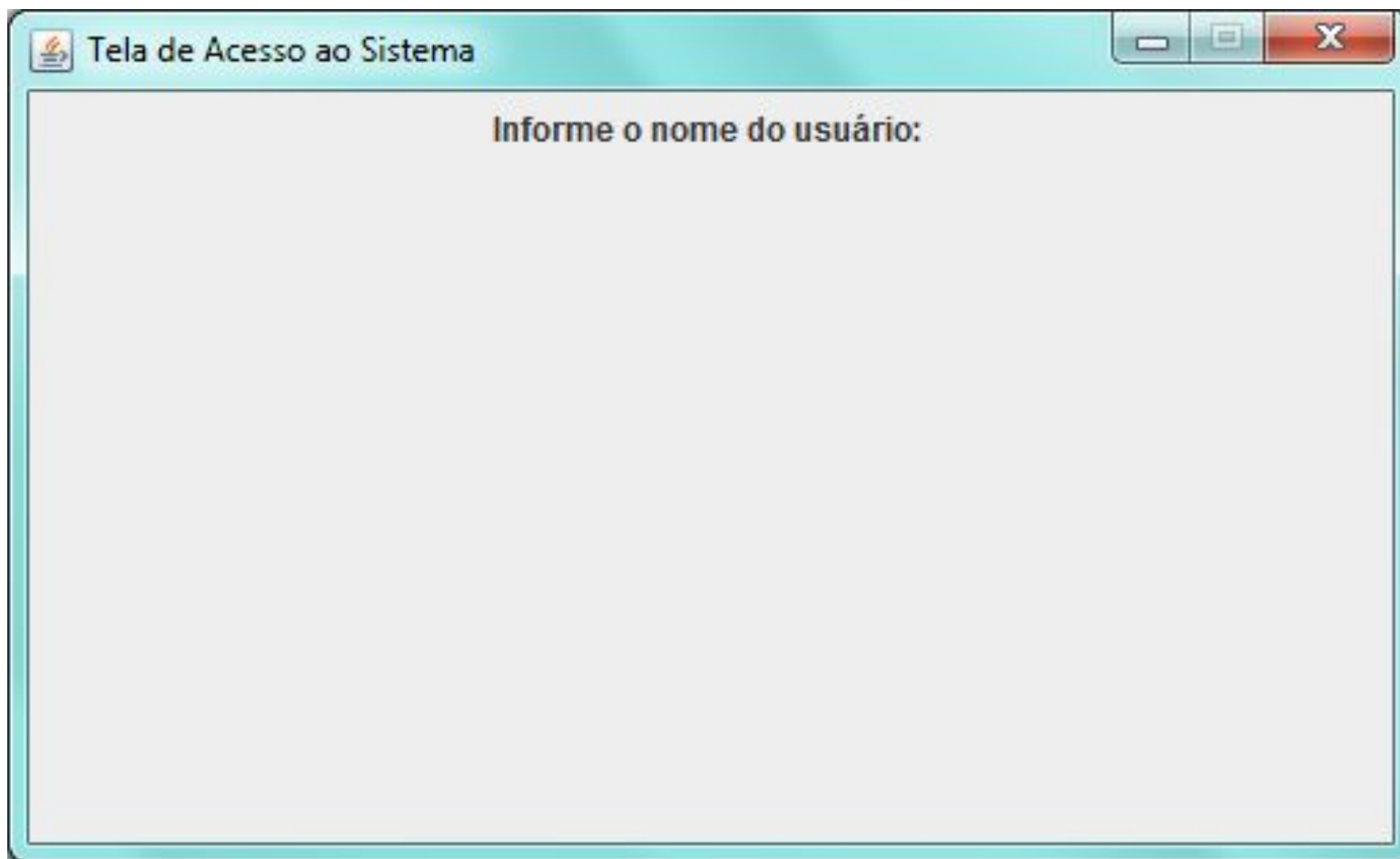
```
package testeJFrame;
import java.awt.Container;
import javax.swing.*.*;
public class Janela extends JFrame
{   public Janela()
    {   this.setTitle("Tela de Acesso ao Sistema");
        this.setSize(500,300);
        this.setVisible(true);
        this.setResizable(false);
        Container c = this.getContentPane();
        JPanel painel = new JPanel();
        c.add(painel);
        JLabel rotulo1 = new JLabel("Informe o nome do usuário:
    ");
        rotulo1.setToolTipText("username");
        painel.add(rotulo1);
    }
}
```



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

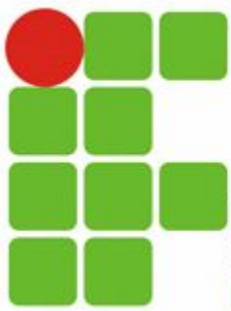
# JLabel

## □ JLabel - Representação Gráfica:



## □ Definição:

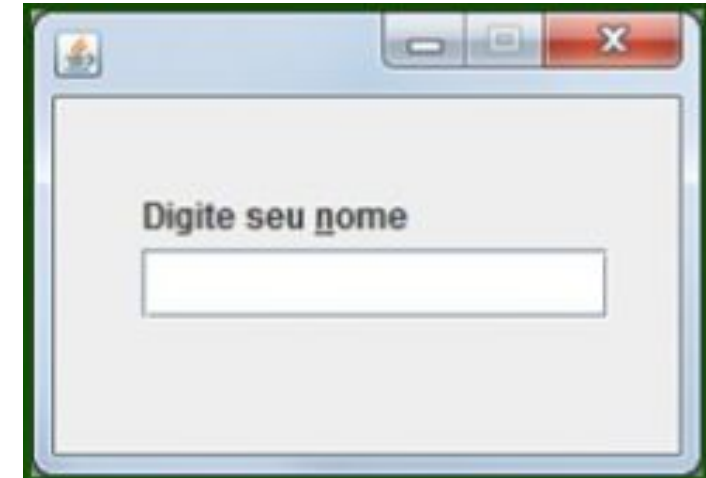
1. Representa uma caixa de texto de uma única linha que pode ser editável ou não. O texto pode ser inserido pelo usuário via teclado ou pode ser simplesmente exibido.
2. Instância de um JTextField  
*JTextField caixa1 = new JTextField(25);*
3. Adicionando o JTextField ao JFrame  
*painel.add(caixa1);*
4. Para tornar um **JTextField** não editável, impossibilitando que o usuário insira texto via teclado, basta fazer o seguinte:  
*caixa1.setEditable(false);*<sup>20</sup>

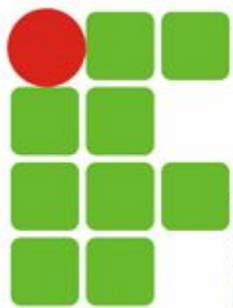


# JTextField

## ❑ JTextField – Código Exemplo:

```
package testeJFrame;  
import java.awt.Container;  
import javax.swing.*.*;  
public class Janela extends JFrame  
{  
    public Janela()  
    {  
        this.setTitle("Minha Janela");  
        this.setSize(500,300);  
        this.setVisible(true);  
        this.setResizable(false);  
        Container c = this.getContentPane();  
        JPanel painel = new JPanel();  
        c.add(painel);  
        JLabel rotulo1 = new JLabel("Informe o nome do usuário: ");  
        rotulo1.setToolTipText("username");  
        painel.add(rotulo1);  
        JTextField caixa1 = new JTextField(25);  
        painel.add(caixa1);  
    }  
}
```

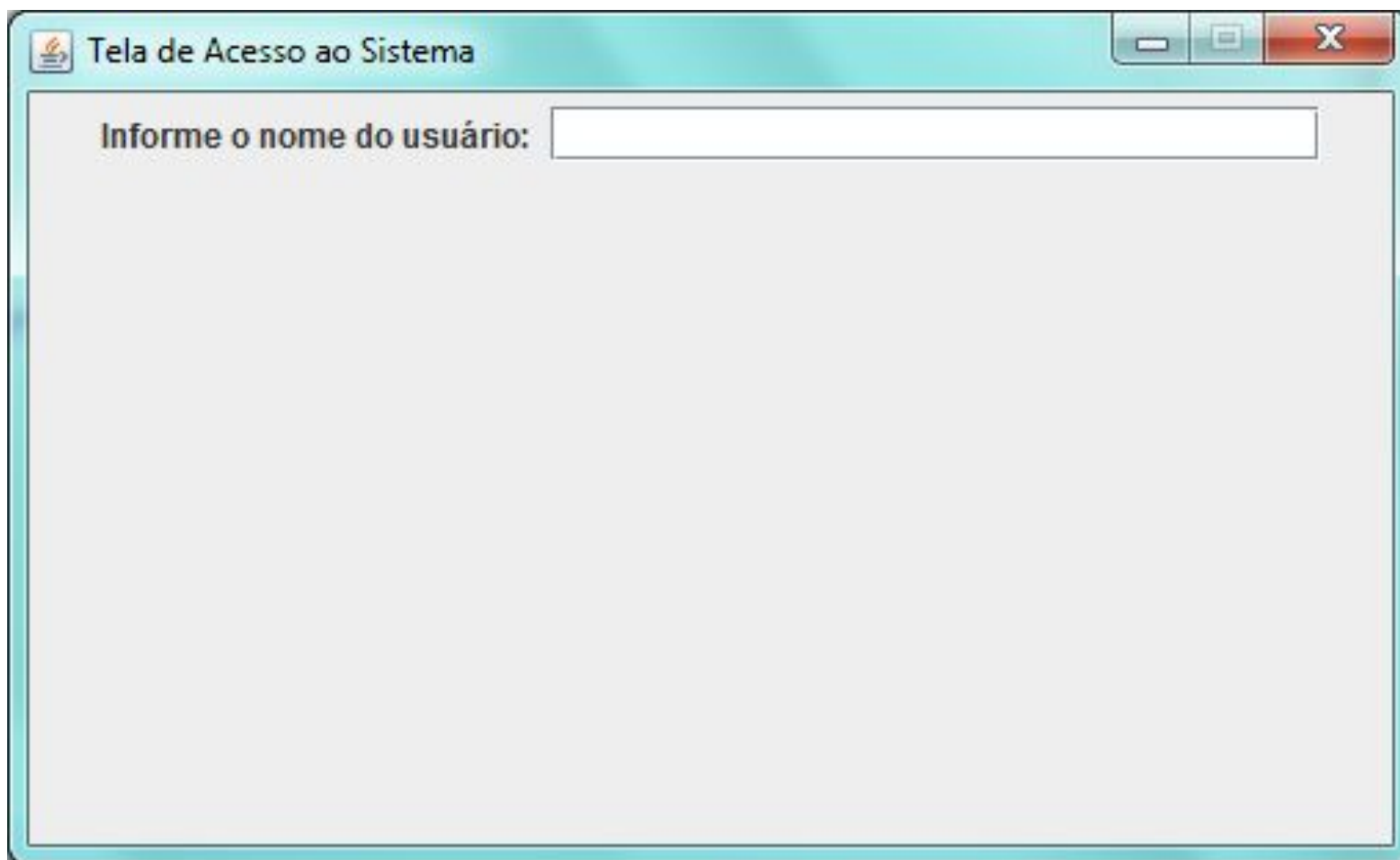


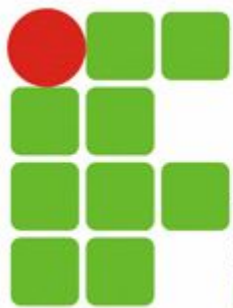


INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# JTextField

## ❑ JTextField - Representação Gráfica:

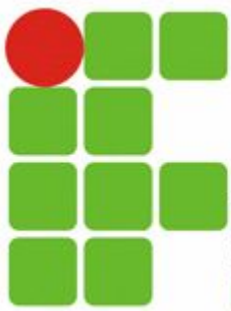




# JPasswordField

## □ Definição:

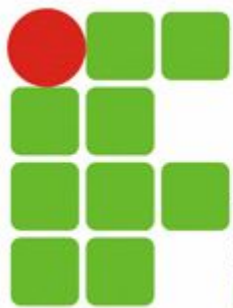
1. Representa uma caixa de texto específica para inserção de uma senha, ocultando a exibição dos caracteres quando inseridos pelo usuário.
2. Instância de um **JPasswordField**  
*JPasswordField senha = new JPasswordField(25);*
3. Adicionando o **JPasswordField** ao **JFrame**  
*painel.add(senha);*



## ❑ JPasswordField – Código Exemplo:

```
package testeJFrame;
import java.awt.Container;
import javax.swing.*.*;
public class Janela extends JFrame
{
    public Janela()
    {
        .....
        JLabel rotulo1 = new JLabel("Informe o nome do usuário: ");
        rotulo1.setToolTipText("username");
        painel.add(rotulo1);
        JTextField caixa1 = new JTextField(25);
        painel.add(caixa1);
        JLabel rotulo2 = new JLabel("Informe a senha: ");
        rotulo2.setToolTipText("password");
        painel.add(rotulo2);
        JPasswordField senha = new JPasswordField(25);
        painel.add(senha);
    }
}
```





INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# JPasswordField

## ❑ JPasswordField - Representação Gráfica:

The screenshot shows a Java Swing window titled "Tela de Acesso ao Sistema". Inside the window, there are two text input fields. The first field is labeled "Informe o nome do usuário:" and contains the text "admin". The second field is labeled "Informe a senha de acesso:" and contains ten black dots, indicating a password field. The window has a standard Mac OS X-style title bar with minimize, maximize, and close buttons.

## □ Definição:

1. Representa um botão que, ao ser clicado pelo usuário, dispara / aciona uma ação específica.

2. Instância de um JButton

```
JButton botao1 = new JButton("Entrar");  
JButton botao2 = new JButton("Limpar");  
JButton botao3 = new JButton("Fechar");
```

3. Adicionando o **JButton** ao **JFrame**

```
painel.add(botao1);  
painel.add(botao2);  
painel.add(botao3);
```

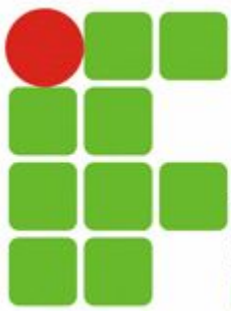
## ❑ Definição (continuação):

4. Disparando eventos através de botões.

```
botao1.addActionListener(new ActionListener()  
{ public void actionPerformed(ActionEvent e)  
    { //codificar evento    } });
```

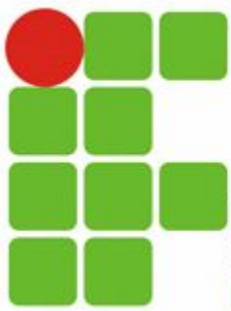
```
botao2.addActionListener(new ActionListener()  
{ public void actionPerformed(ActionEvent e)  
    { //codificar evento    } });
```

```
botao3.addActionListener(new ActionListener()  
{ public void actionPerformed(ActionEvent e)  
    { //codificar evento    } });
```



## □ JButton – Código Exemplo:

```
package testeJFrame;  
import java.awt.Container;  
import javax.swing.*;  
public class Janela extends JFrame  
{   public Janela()  
    {   .....  
        JButton botao1 = new JButton("Entrar");  
        JButton botao2 = new JButton("Limpar");  
        JButton botao3 = new JButton("Fechar");  
  
        painel.add(botao1);  
        painel.add(botao2);  
        painel.add(botao3);
```



## ■ JButton – Código Exemplo (continuação):

.....

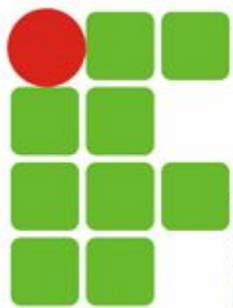
```
botao1.addActionListener(new ActionListener()  
{    public void actionPerformed(ActionEvent e)  
    {        JOptionPane.showMessageDialog(null,"USUÁRIO: " +  
            caixa1.getText() + "\nSENHA: " +senha.getPassword());  
    }});
```

```
botao2.addActionListener(new ActionListener()           // botão limpar  
{    public void actionPerformed(ActionEvent e)  
    {        caixa1.setText("");  
            senha.setText("");    }});
```

```
botao3.addActionListener(new ActionListener()           // botão fechar  
{    public void actionPerformed(ActionEvent e)  
    {        System.exit(0);  
    }});
```

```
}
```

```
}
```

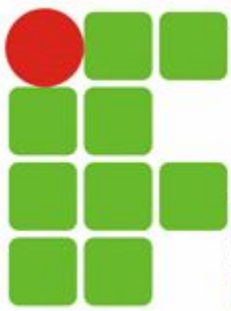


INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# JButton

## ■ JButton - Representação Gráfica:

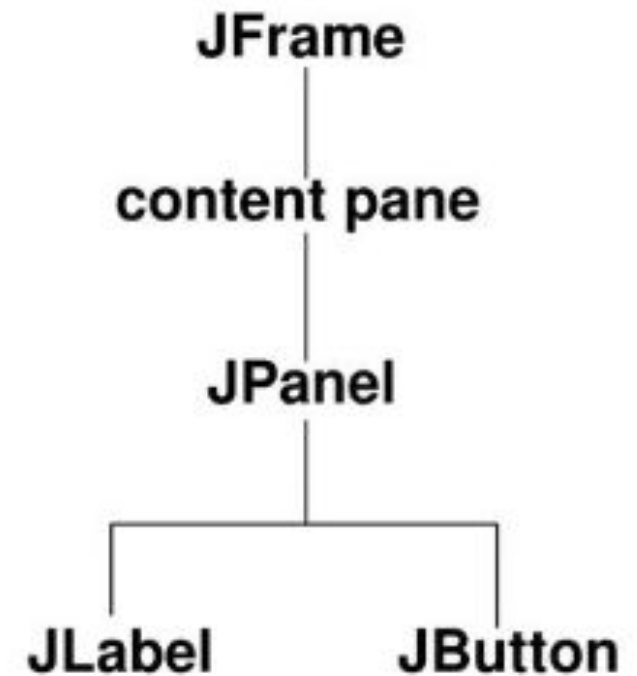
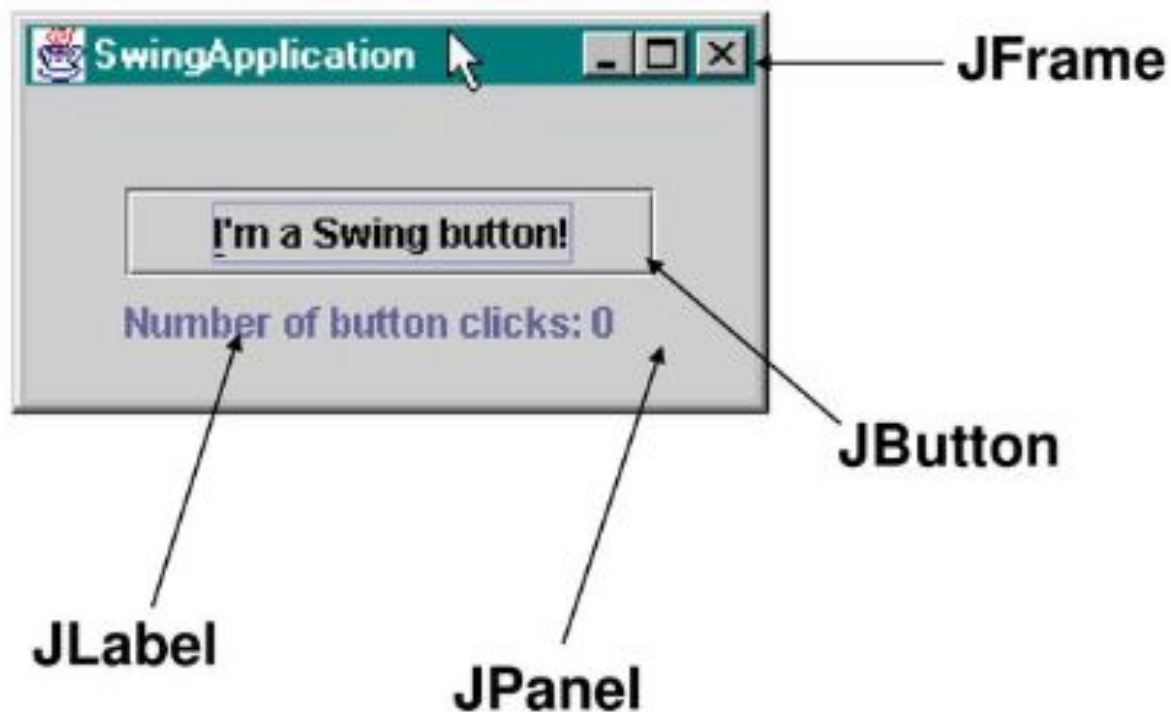
The image shows a Java Swing window titled "Tela de Acesso ao Sistema". Inside the window, there are two text input fields. The first field is labeled "Informe o nome do usuário:" and contains the text "admin". The second field is labeled "Informe a senha de acesso:" and contains ten black dots, representing a password. Below the password field, there are three buttons: "Entrar", "Limpar", and "Fechar". The window has a standard title bar with minimize, maximize, and close buttons.

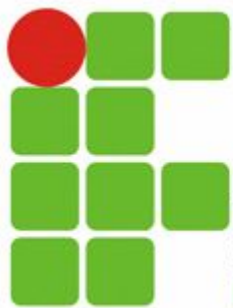


INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# GUI PRINCIPAIS COMPONENTES

## □ Hierarquia dos Componentes

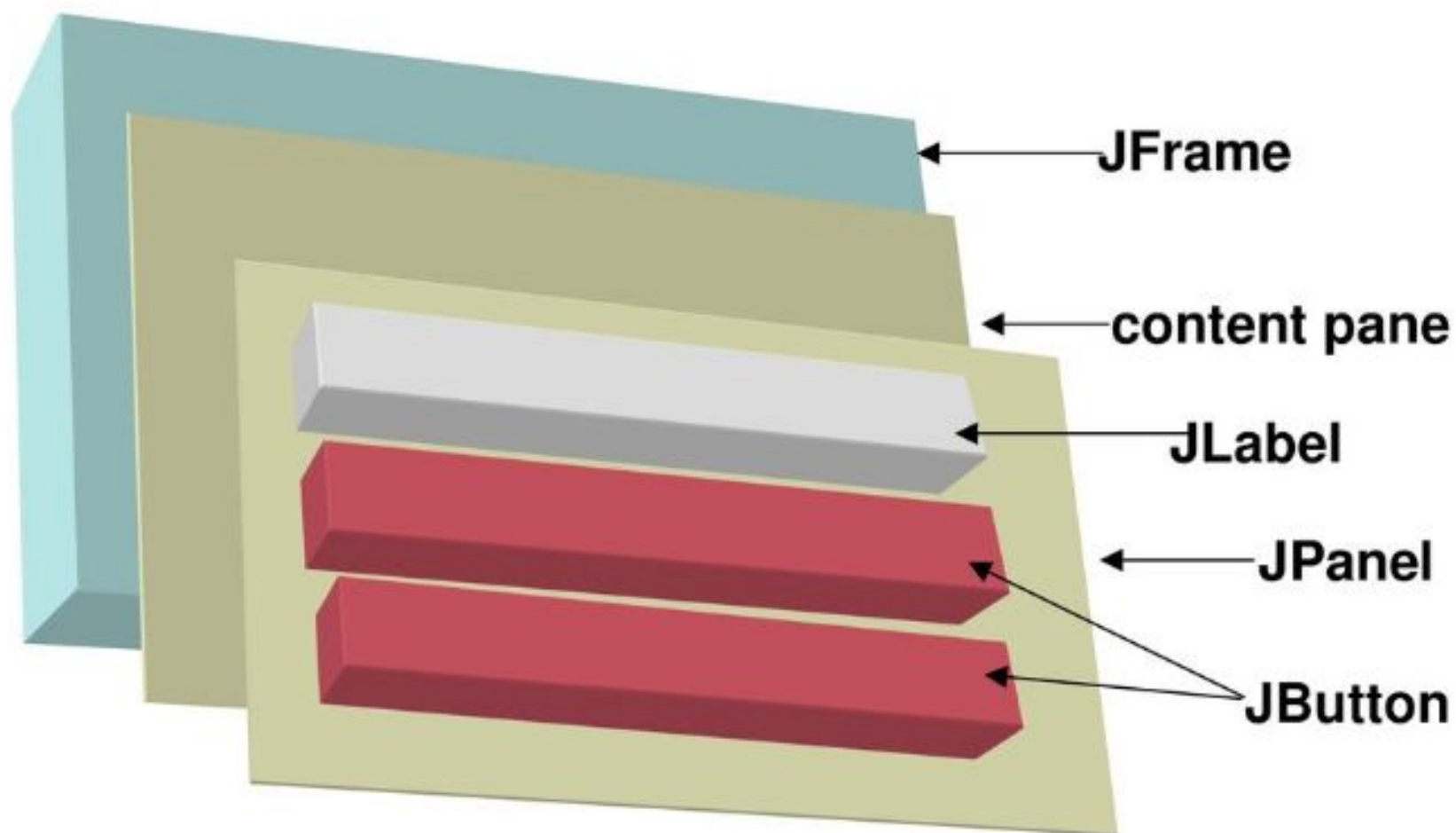




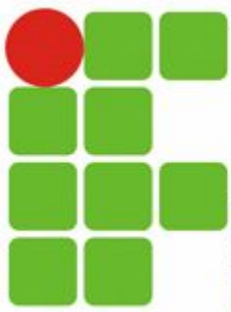
INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# GUI PRINCIPAIS COMPONENTES

## □ Hierarquia dos Componentes







INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# GUI

## PRINCIPAIS COMPONENTES

- Principais Componentes de uma GUI
  - JCheckBox
  - JRadioButton
  - JComboBox
  - JList

---

Execute o código-fonte ***JanelaParte02.java*** para visualizar estes componentes de uma GUI.

## ❑ Definição:

1. Representa uma caixa de seleção, é uma espécie de botão de estado que pode ter os valores ativado/desativado ou verdadeiro/falso, permitindo que mais de um item seja selecionado.

2. Instância de um JCheckBox

```
JCheckBox escolha1 = new JCheckBox("Administrador",false);  
JCheckBox escolha2 = new JCheckBox("Programador",false);  
JCheckBox escolha3 = new JCheckBox("DBA",false);  
JCheckBox escolha4 = new JCheckBox("Usuário Final",true);  
JCheckBox escolha5 = new JCheckBox("Testador",false);
```

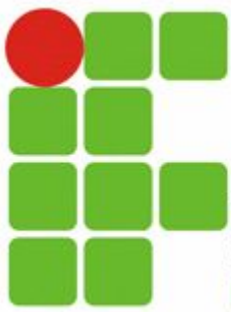
3. Adicionando o **JCheckBox** ao **JFrame**

```
painel.add(escolha1);  
painel.add(escolha2);  
painel.add(escolha3);  
painel.add(escolha4);  
painel.add(escolha5);
```

## ❑ JCheckbox – Código Exemplo:

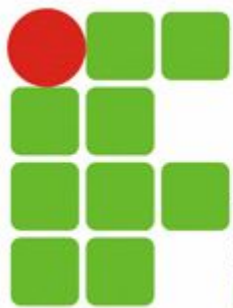
```
package testeJFrame;
import java.awt.Container;
import javax.swing.*.*;
public class Janela extends JFrame
{
    public Janela()
    {
        .....
        JLabel rotulo3 = new JLabel("É obrigatório selecionar o(s) perfil(is)
desejado(s) para acesso aos sistemas: ");
        rotulo3.setToolTipText("perfil de acesso");
        painel.add(rotulo3);

        JCheckBox escolha1 = new JCheckBox("Administrador",false);
        JCheckBox escolha2 = new JCheckBox("Programador",false);
        JCheckBox escolha3 = new JCheckBox("DBA",false);
        JCheckBox escolha4 = new JCheckBox("Usuário",true);
        JCheckBox escolha5 = new JCheckBox("Testador",false);
```



## ❑ JCheckbox – Código Exemplo (continuação):

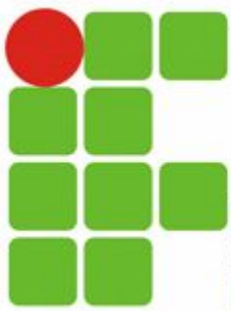
```
.....  
    painel.add(escolha1);  
    painel.add(escolha2);  
    painel.add(escolha3);  
    painel.add(escolha4);  
    painel.add(escolha5);  
    .....  
    }  
}
```



## ❑ JCheckBox - Representação Gráfica:

The screenshot shows a Java Swing window titled "Tela de Acesso ao Sistema". Inside the window, there is a login form with the following elements:

- A label "Informe o nome do usuário:" followed by a text input field containing the text "admin".
- A label "Informe a senha de acesso:" followed by a password input field with 10 dots.
- A text label: "É obrigatório selecionar o(s) perfil(is) desejado(s) para acesso aos sistemas:"
- A row of five JCheckBox components:
  - ☒ Administrador
  - ☐ Programador
  - ☐ DBA
  - ☒ Usuário Final
  - ☒ Testador
- A row of three buttons: "Entrar", "Limpar", and "Fechar".



# JRadioButton

## Definição:

1. Representa um botão de rádio permitindo a seleção de apenas um único item dentre os vários disponíveis. O relacionamento lógico entre um conjunto de componentes **JRadioButton** é mantido por um objeto **ButtonGroup**.

2. Instância de um **JRadioButton**

```
JRadioButton radio1 = new JRadioButton("Administrador",false);  
JRadioButton radio2 = new JRadioButton("Programador",false);  
JRadioButton radio3 = new JRadioButton("DBA",false);  
JRadioButton radio4 = new JRadioButton("Usuário Final",true);  
JRadioButton radio5 = new JRadioButton("Testador",false);
```

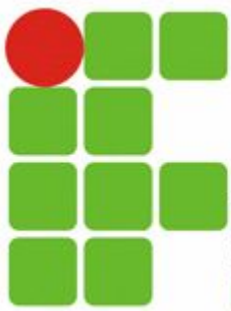
3. Criando um **ButtonGroup** e adicionando os componentes **JRadioButton**

```
ButtonGroup grupo = new ButtonGroup();  
grupo.add(radio1);  
grupo.add(radio2);  
grupo.add(radio3);  
grupo.add(radio4);  
grupo.add(radio5);
```

## ▣ JRadioButton – Definição (continuação):

4. Adicionando o JRadioButton ao JFrame

```
painel.add(audio1);  
painel.add(audio2);  
painel.add(audio3);  
painel.add(audio4);  
painel.add(audio5);
```



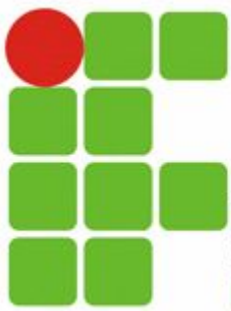
## ❑ JRadioButton – Código Exemplo:

```
package testeJFrame;  
import java.awt.Container;  
import javax.swing.*;  
public class Janela extends JFrame  
{  
    public Janela()  
    {  
        .....
```

```
        JLabel rotulo3 = new JLabel("É obrigatório selecionar o(s) perfil(is)  
desejado(s) para acesso aos sistemas: ");  
        rotulo3.setToolTipText("perfil de acesso");  
        painel.add(rotulo3);
```

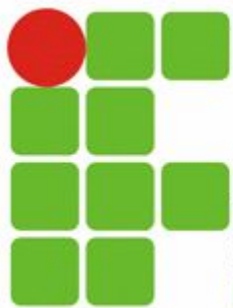
```
        JRadioButton radio1 = new JRadioButton("Administrador",false);  
        JRadioButton radio2 = new JRadioButton("Programador",false);  
        JRadioButton radio3 = new JRadioButton("DBA",false);  
        JRadioButton radio4 = new JRadioButton("Usuário Final",true);  
        JRadioButton radio5 = new JRadioButton("Testador",false);
```





## ❑ JRadioButton – Código Exemplo (continuação):

```
.....  
ButtonGroup grupo = new ButtonGroup();  
grupo.add(audio1);  
grupo.add(audio2);  
grupo.add(audio3);  
grupo.add(audio4);  
grupo.add(audio5);  
  
    painel.add(audio1);  
    painel.add(audio2);  
    painel.add(audio3);  
    painel.add(audio4);  
    painel.add(audio5);  
    .....  
}  
}
```



INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# JRadioButton

## ❑ JRadioButton - Representação Gráfica:

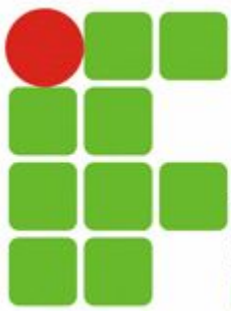
The screenshot shows a Java Swing window titled "Tela de Acesso ao Sistema". Inside the window, there is a login form with the following elements:

- A text field labeled "Informe o nome do usuário:" containing the text "admin".
- A text field labeled "Informe a senha de acesso:" containing eight dots, representing a password.
- A label: "É obrigatório selecionar o(s) perfil(is) desejado(s) para acesso aos sistemas:".
- A row of five radio buttons with the following labels: "Administrador", "Programador", "DBA", "Usuário Final", and "Testador". The "Usuário Final" radio button is selected.
- Three buttons at the bottom: "Entrar", "Limpar", and "Fechar".

## □ Definição:

1. É uma caixa de combinação, também conhecida como lista **drop-down** (lista suspensa). Fornece uma lista de itens entre os quais o usuário pode escolher um.
2. Quando o usuário clica no componente, aparece uma lista de opções permitindo que uma delas seja escolhida.
3. O método **setMaximumRowCount** define o número máximo de elementos que serão exibidos quando o usuário clicar no componente. Uma barra de rolagem aparecerá automaticamente para que os demais elementos possam ser visualizados.
4. Instância de JComboBox :  

```
String[] perfil = {"Administrador", "Programador", "DBA", "Usuário Final", "Testador"};  
JComboBox combo = new JComboBox (perfil);  
combo.setMaximumRowCount(4);  
painel.add(combo); // adiciona o JComboBox ao JFrame
```

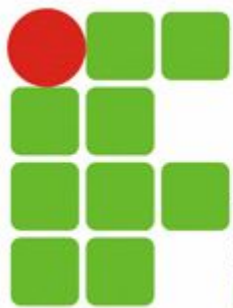


## ❑ JComboBox – Código Exemplo:

```
package testeJFrame;
import java.awt.Container;
import javax.swing.*.*;
public class Janela extends JFrame
{
    public Janela()
    {
        .....
        JLabel rotulo3 = new JLabel("É obrigatório selecionar o(s) perfil(is)
desejado(s) para acesso aos sistemas: ");
        rotulo3.setToolTipText("perfil de acesso");
        painel.add(rotulo3);

        String[] perfil = {"Administrador", "Programador", "DBA", "Usuário
Final", "Testador"};
        JComboBox combo = new JComboBox(perfil);
        combo.setMaximumRowCount(4);
        painel.add(combo);

        .....
    }
}
```



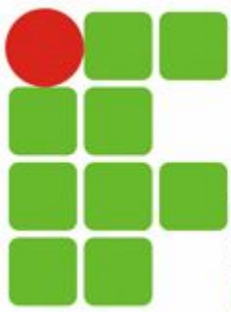
## ❑ JComboBox - Representação Gráfica:

The screenshot shows a Java Swing window titled "Tela de Acesso ao Sistema". It contains two text input fields: "Informe o nome do usuário:" with the text "admin" and "Informe a senha de acesso:" with masked characters. Below these is a label "É obrigatório selecionar o(s) perfil(is) desejado(s) para acesso aos sistemas:". Underneath is a JComboBox with a list of user profiles: "Administrador", "Administrador", "Programador", "DBA", and "Usuário Final". The first "Administrador" entry is selected and highlighted. The window has standard Windows-style title bar controls (minimize, maximize, close).

## Definição:

1. É uma lista que exibe vários itens, permitindo ao usuário selecionar um ou mais itens.
2. Pode ser de única seleção (um item selecionado por vez) ou de múltipla seleção (vários itens selecionados por vez).
3. Não fornece automaticamente uma barra de rolagem, sendo necessário incluí-la através do componente JScrollPane.
4. Instância de JList :

```
String[] list = {Administrador", "Programador", "DBA", "Usuário Final", "Testador" };  
JList lista = new JList (list);  
lista.setVisibleRowCount(5);
```



## Definição:

5. Os métodos mais usados são:

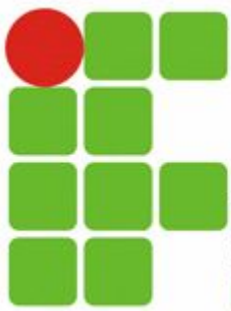
- setVisibleRowCount - define o número de itens visíveis.
- setSelectionMode - especifica o modo de seleção da lista.

*Exemplos:*

*lista.setSelectionMode(ListSelectionModel.SINGLE\_SELECTION);*

*ou*

*lista.setSelectionMode(ListSelectionModel.MULTIPLE\_INTERVAL\_SELECTION);*



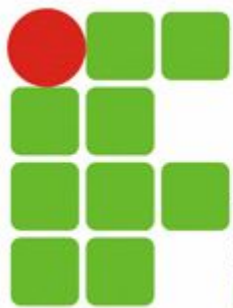
## ❑ JList – Código Exemplo:

```
package testeJFrame;
import java.awt.Container;
import javax.swing.*;
public class Janela extends JFrame
{
    public Janela()
    {
        .....
        JLabel rotulo3 = new JLabel("É obrigatório selecionar o(s) perfil(is)
desejado(s) para acesso aos sistemas: ");
        rotulo3.setToolTipText("perfil de acesso");
        painel.add(rotulo3);

        String[] relacao = {"Administrador", "Programador", "DBA", "Usuário Final",
        "Testador"};
        JList lista = new JList (relacao);
        lista.setVisibleRowCount(5);
        lista.setSelectionMode(ListSelectionModel.SINGLE_SELECTION);
        painel.add(lista);

        .....
    }
}
```





## □ JList - Representação Gráfica:

Tela de Acesso ao Sistema

Informe o nome do usuário: admin

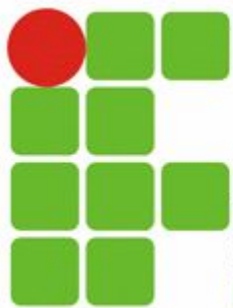
Informe a senha de acesso: ●●●●●●●●

É obrigatório selecionar o(s) perfil(is) desejado(s) para acesso aos sistemas:

- Administrador
- Programador**
- DBA
- Usuário Final
- Testador

Entrar Limpar Fechar

*lista.setSelectionMode(ListSelectionModel.SINGLE\_SELECTION);*



## ❑ JList - Representação Gráfica:

Tela de Acesso ao Sistema

Informe o nome do usuário: admin

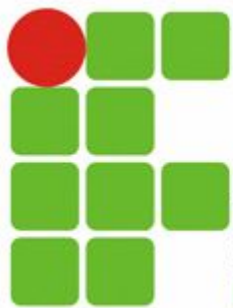
Informe a senha de acesso: ●●●●●●●●

É obrigatório selecionar o(s) perfil(is) desejado(s) para acesso aos sistemas:

- Administrador
- Programador
- DBA
- Usuário Final
- Testador

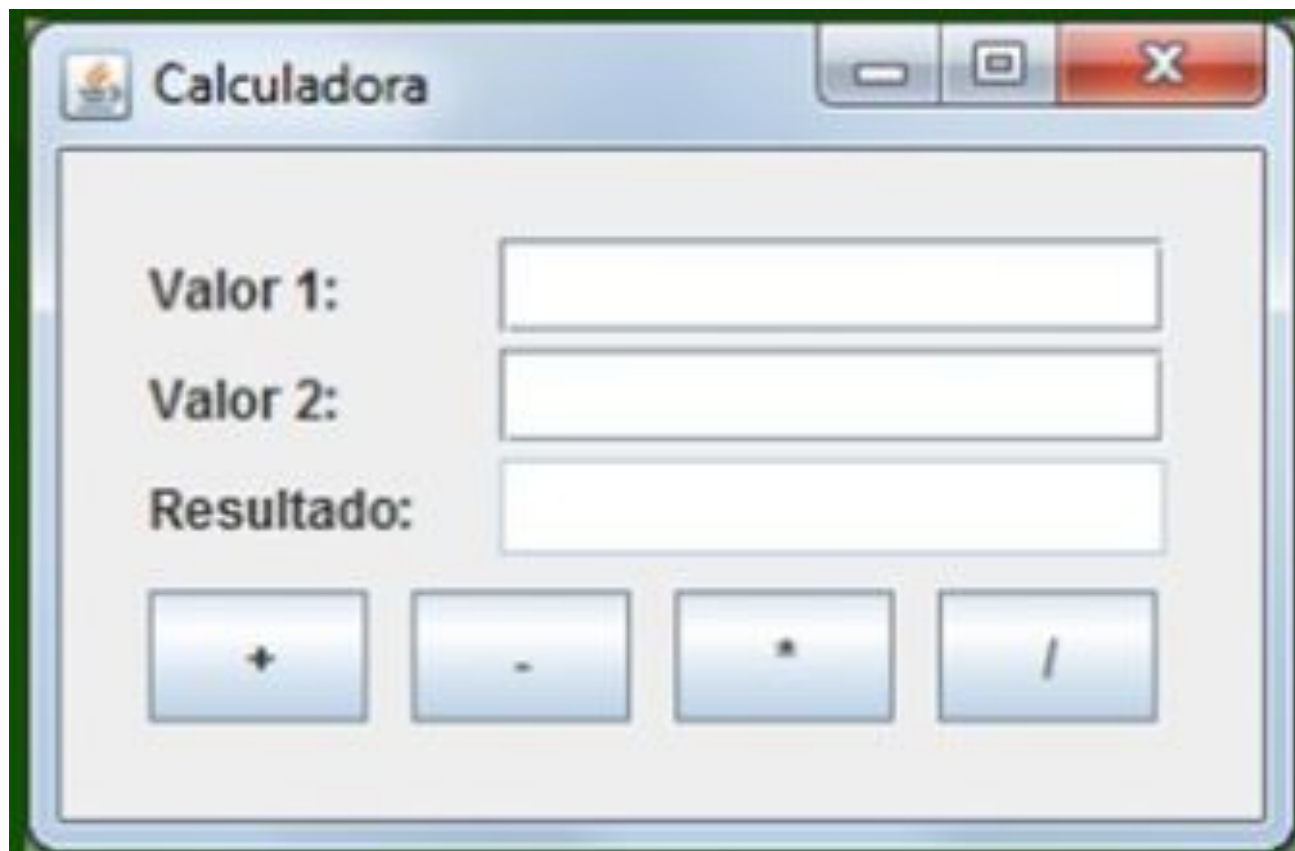
Entrar Limpar Fechar

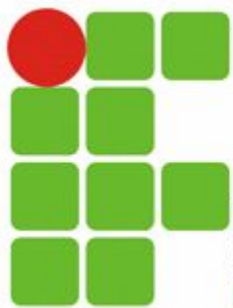
```
lista.setSelectionMode(ListSelectionModel.MULTIPLE_INTERVAL_SELECTION);
```



# COMPONENTES GUI

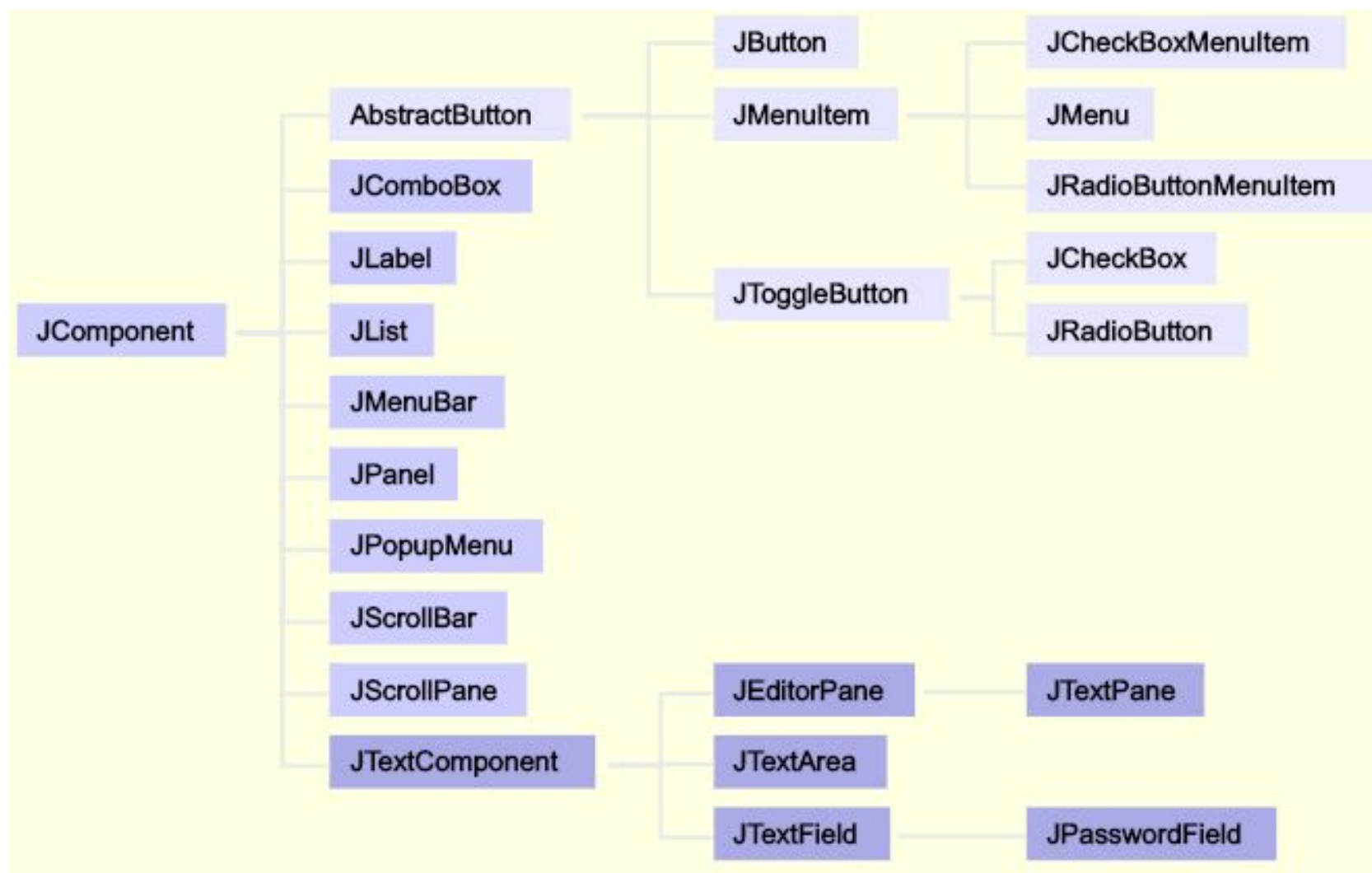
- ❑ **Exercício:** implemente esta calculadora com as operações básicas. Ao clicar nos botões, o resultado da operação deve ser apresentado na caixa Resultado.

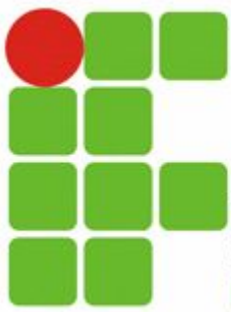




# GUI PRINCIPAIS COMPONENTES

## □ Hierarquia de alguns dos Componentes da GUI





INSTITUTO FEDERAL DE  
EDUCAÇÃO, CIÊNCIA E TECNOLOGIA  
FLUMINENSE

# Material Complementar

- <https://slideplayer.com.br/slide/358129/>
- <https://slideplayer.com.br/slide/14154557/>
- <https://slideplayer.com.br/slide/367311/>
- [https://www.youtube.com/watch?v=S\\_0iPf5-iDs](https://www.youtube.com/watch?v=S_0iPf5-iDs) - Usando JPanel e posicionando componentes
- [https://cae.ucb.br/conteudo/programar/labor2/new\\_swing.html](https://cae.ucb.br/conteudo/programar/labor2/new_swing.html)
- <https://slideplayer.com.br/slide/367311/> - Componente ButtonGroup (slide 37)