

POO e Componentes

Graphical User Interface (GUI)

André Santanchè
Laboratory of Information Systems – LIS
Instituto de Computação – UNICAMP
Junho 2020



AWT

■ Abstract Window Toolkit (AWT)

- biblioteca de classes do Java
- responsável pela elaboração da Interface Gráfica

■ Lowest-common Denominator (LCD)

- componentes visuais compartilhados por todas as plataformas

■ GUI peer controls

- cada componente tem um par na interface nativa



JFC e Swing

■ Java Foundation Classes (JFC)

- biblioteca de classes lançada em 1997
- superconjunto da AWT

■ Componentes Swing

- parte da JFC
- estende a AWT com mais recursos para GUI
- não há peer controls

Componente

Descrição

Exemplo

Containers de alto nível

Componente principal que irá conter os demais

JFrame



Containers intermediários

Pode conter outros componentes porém tem que estar inserido em um componente superior

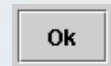
JPanel



Componentes atômicos

Componentes auto-suficientes que possuem uma apresentação e funcionalidade

JButton



Criando uma Janela

Passo 1 – Criar uma Janela (alto nível)

JFrame

```
JFrame janela = new JFrame();
```



Passo 2 – Customizar a Janela



```
JFrame janela = new JFrame();  
  
janela.setSize(300, 200);  
janela.setDefaultCloseOperation(  
    JFrame.EXIT_ON_CLOSE);
```

Passo 3 – Apresentar a Janela



```
JFrame janela = new JFrame();
```

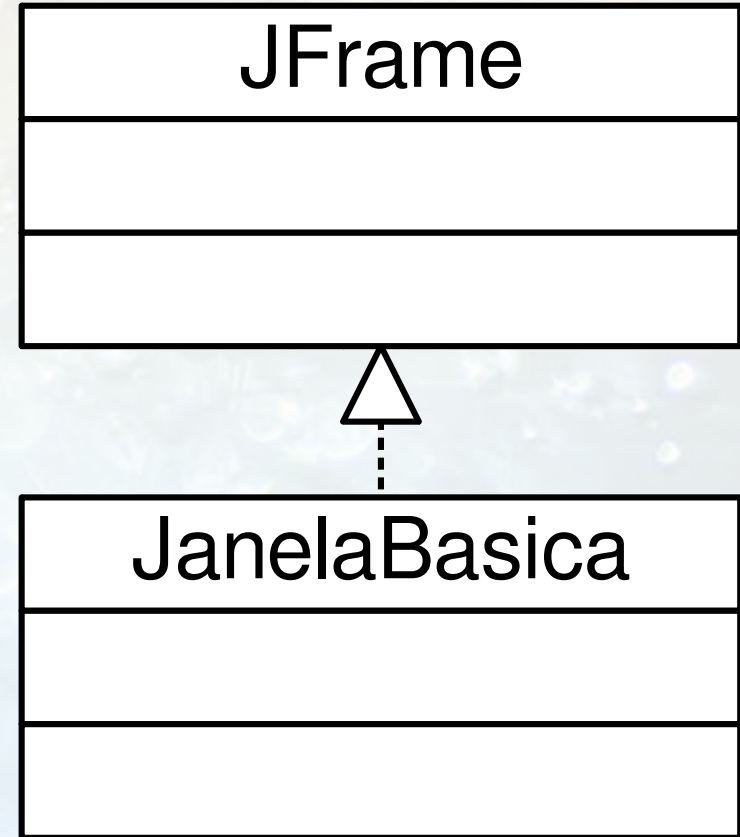
```
janela.setSize(300, 200);
```

```
janela.setDefaultCloseOperation(  
    JFrame.EXIT_ON_CLOSE);
```

```
janela.setVisible(true);
```


Estendendo uma Janela

Herdeira de JFrame + Customização





POO e Componentes

Graphical User Interface (GUI)

Janela com Componentes

André Santanchè

Laboratory of Information Systems – LIS

Instituto de Computação – UNICAMP

Junho 2020

Janela com Componentes

Estratégia 1: Criando Componentes dentro da Janela

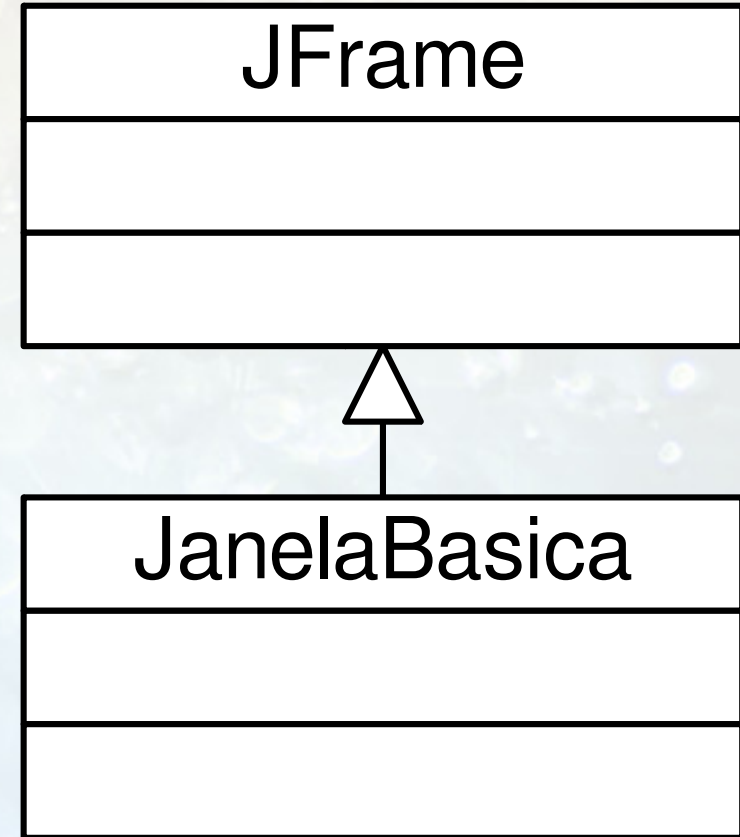
Passo 1 – Criar uma Janela (alto nível)

JFrame

```
JFrame janela = new JFrame();
```



Passo 1 – Janela herdeira de JFrame

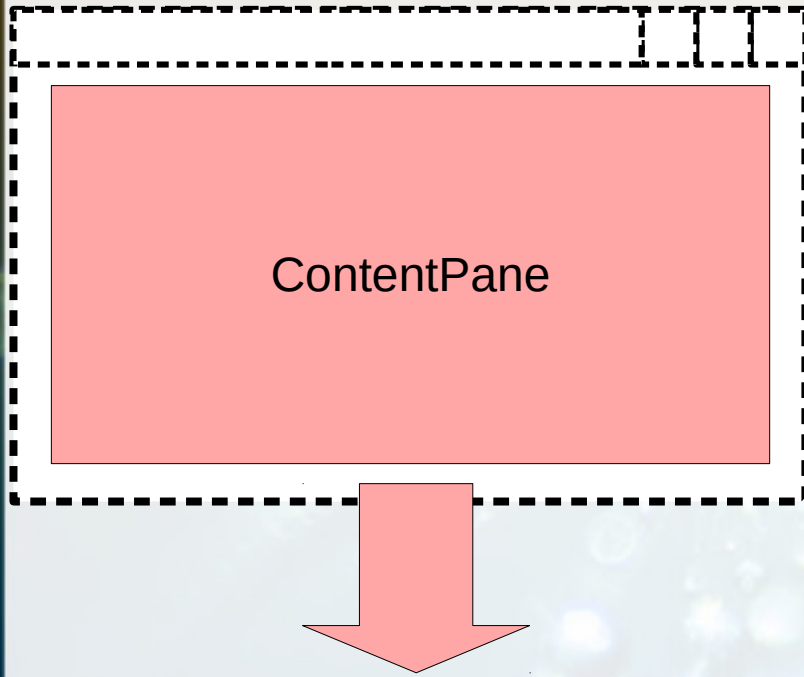


Passo 2 – Customizar a Janela



```
setSize(500, 200);  
setDefaultCloseOperation(  
    EXIT_ON_CLOSE);
```

Passo 3 – Recuperar o ContentPane



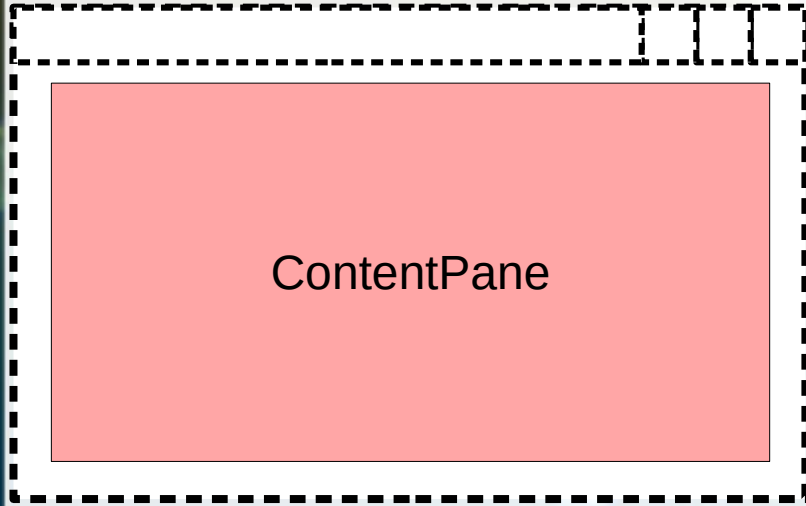
Painel do JPanel que representa a área de conteúdo.

```
setSize(500, 200);
```

```
setDefaultCloseOperation(  
    EXIT_ON_CLOSE);
```

```
Container contentPane =  
    getContentPane();
```

Passo 4 – Escolher um Gerente de Layout



```
setSize(500, 200);
```

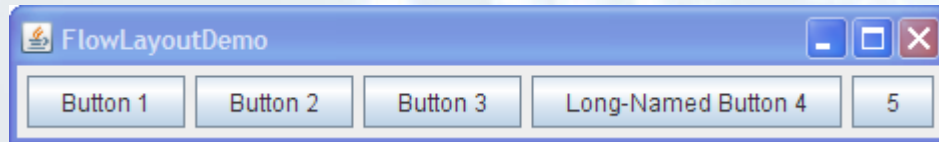
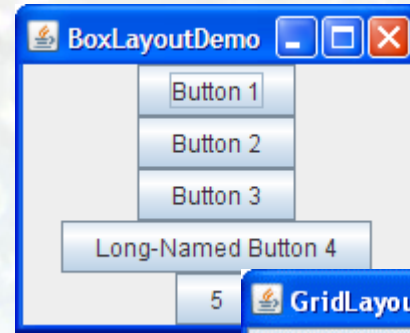
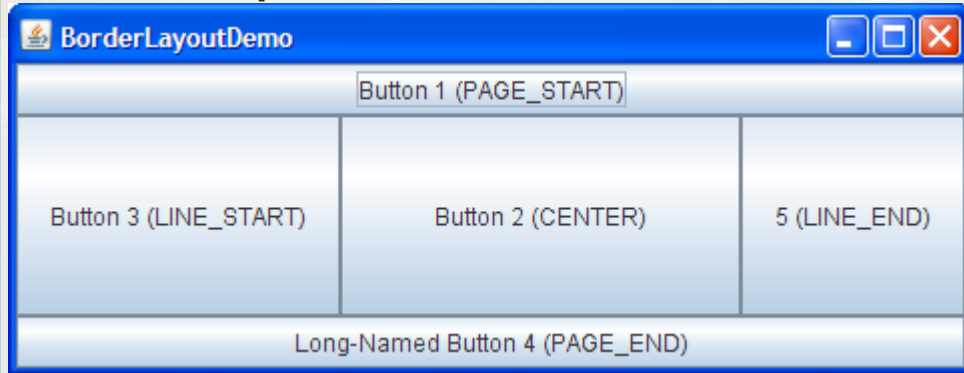
```
setDefaultCloseOperation(  
    EXIT_ON_CLOSE);
```

```
Container contentPane =  
    getContentPane();
```

```
contentPane.setLayout(  
    new BorderLayout());
```

Gerentes de Layout

- Responsáveis pela organização dos componentes em um painel



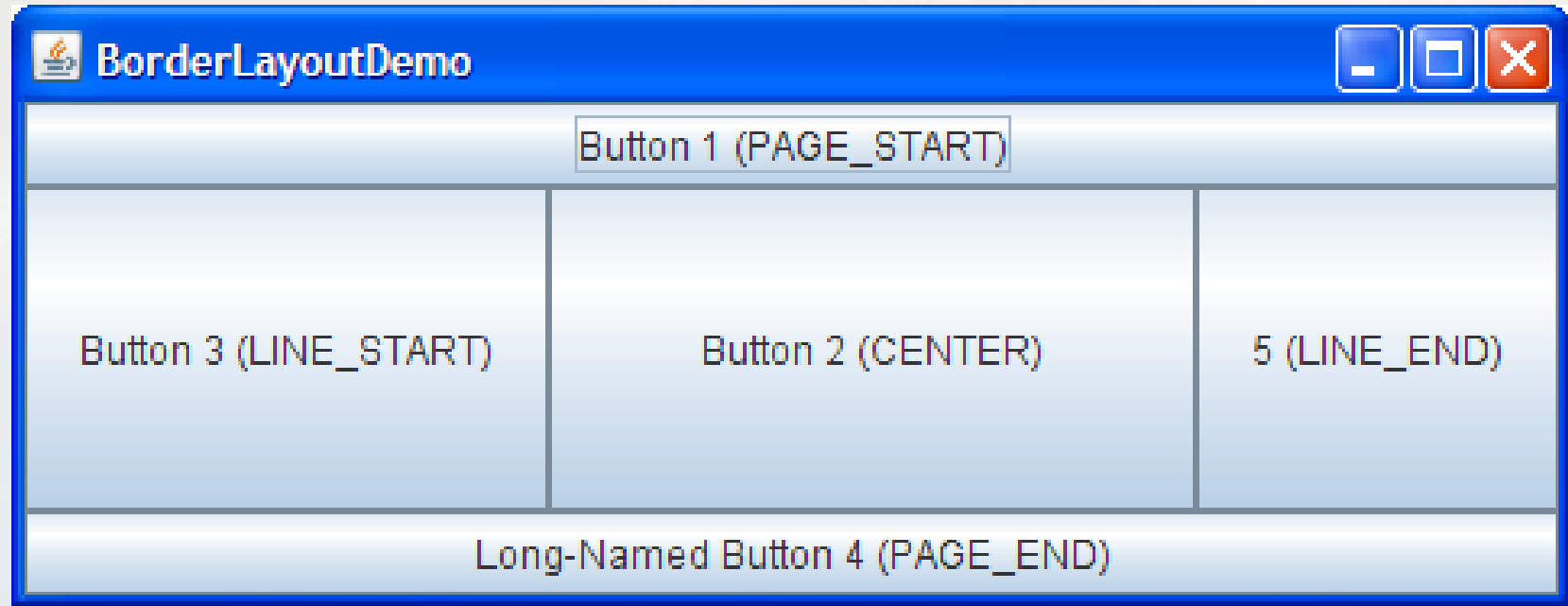
A Visual Guide to Layout Managers

<https://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

Guia Visual de Gerentes de Layout

<https://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

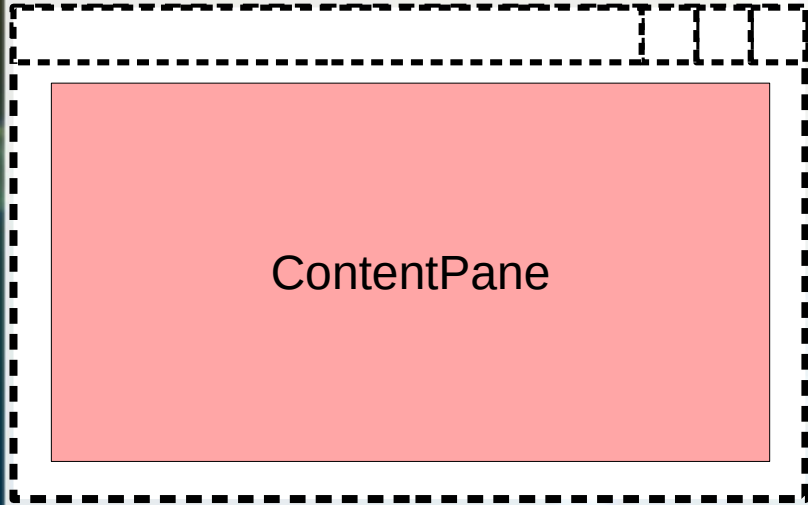
BorderLayout()



A Visual Guide to Layout Managers

<https://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

Escolhendo o BorderLayout



```
setSize(500, 200);
```

```
setDefaultCloseOperation(  
    EXIT_ON_CLOSE);
```

```
Container contentPane =  
    getContentPane();
```

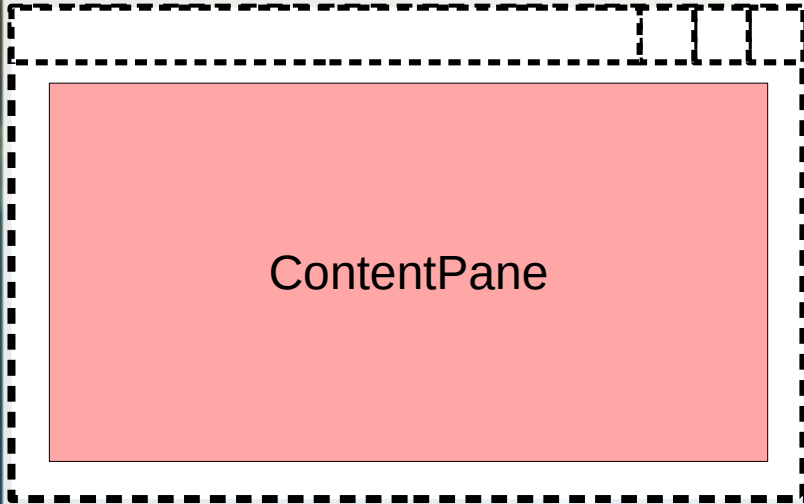
```
contentPane.setLayout(  
    new BorderLayout());
```

Passo 5 – Criar um Botão

...

```
contentPane.setLayout(  
    new BorderLayout());
```

```
JButton botao =  
    new JButton("Ok");
```



Guia Visual de Componentes

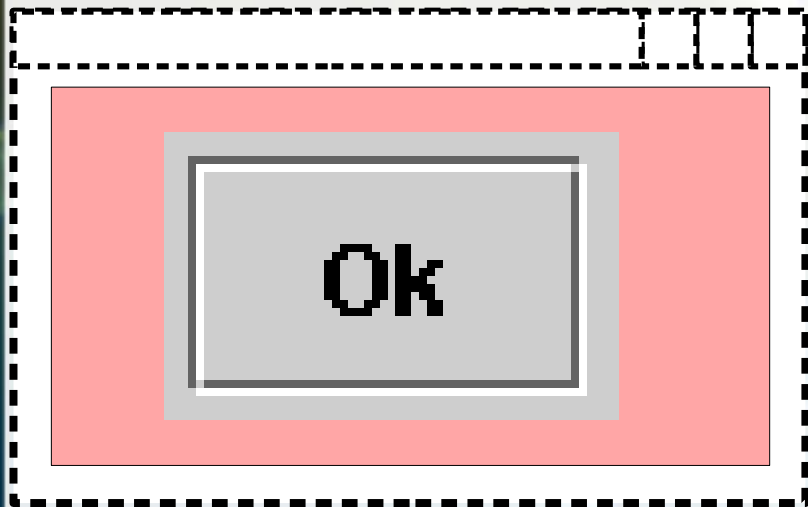
<https://web.mit.edu/6.005/www/sp14/psets/ps4/java-6-tutorial/components.html>

Passo 6 – Acrescentar o Botão no Painei

...

```
contentPane.setLayout(  
    new BorderLayout());
```

```
 JButton botao =  
    new JButton("Ok");  
contentPane.add(botao,  
    BorderLayout.CENTER);
```



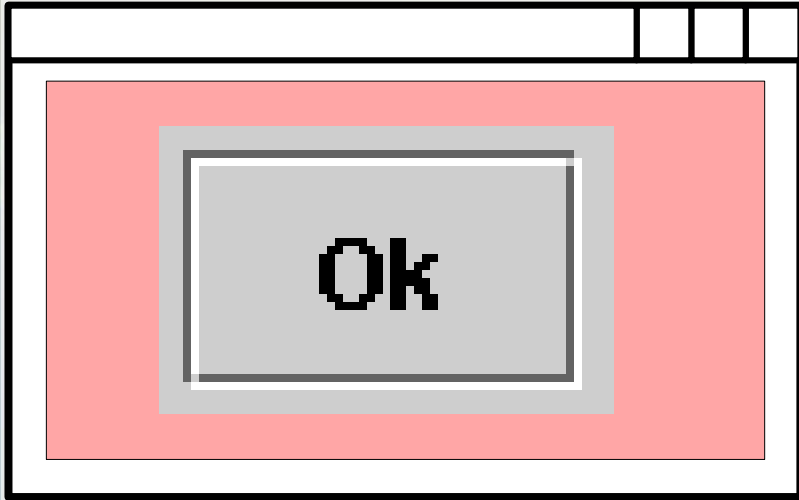
posição no layout



Passo 7 – Apresentar a Janela

...

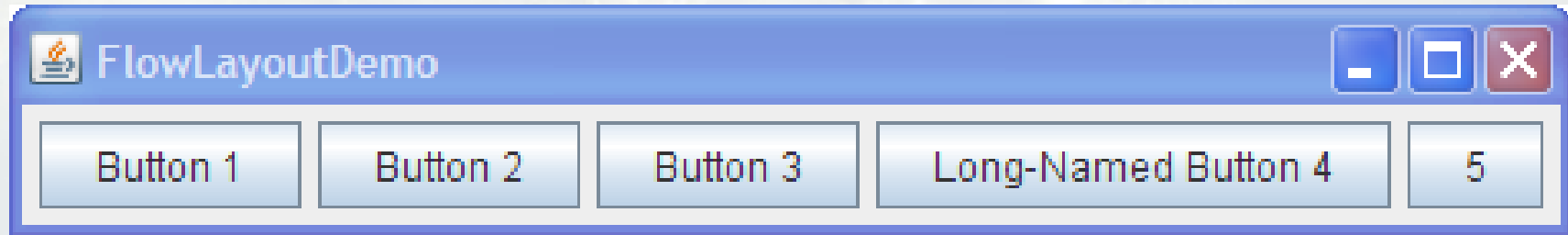
```
contentPane.setLayout(  
    new BorderLayout());
```



```
JButton botao =  
    new JButton("Ok");  
contentPane.add(botao,  
    BorderLayout.CENTER);
```

```
setVisible(true);
```

Mudando para FlowLayout()



A Visual Guide to Layout Managers

<https://docs.oracle.com/javase/tutorial/uiswing/layout/visual.html>

Guardando Referências para Componentes para Controlá-los

Usando Imagens

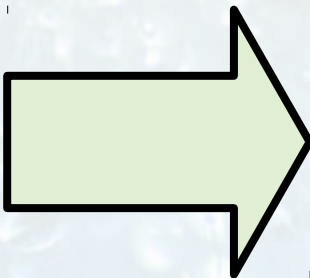
Passo 1 – Criar um objeto da Imagem

```
ImageIcon imagem = new ImageIcon(arquivoImagem);
```



Passo 2 – Criar um JLabel com a Imagem

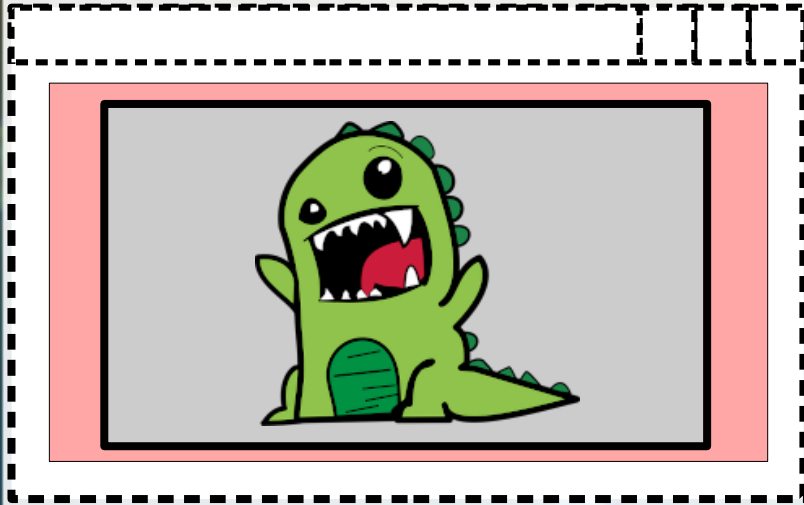
```
ImageIcon imagem = new ImageIcon(arquivoImagem);  
JLabel campoImagem = new JLabel(imagem);
```



Passo 3 – Acrescentando o JLabel no Painel

...

```
ImageIcon imagem =  
    new ImageIcon(arquivoImagem);  
JLabel campoImagem =  
    new JLabel(imagem);  
contentPane.add(campoImagem,  
    BorderLayout.CENTER);
```



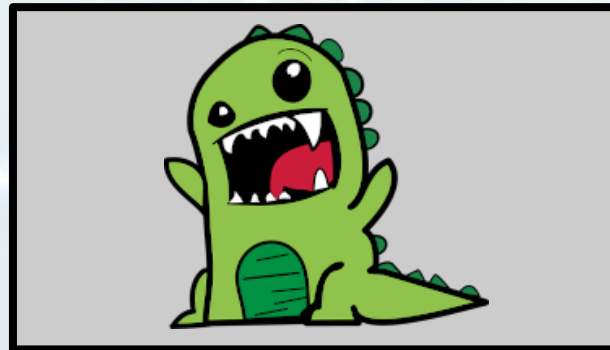
posição no layout



Janela com Componentes

Estratégia 2: Estendendo Componentes

JLabel para Componente de Imagem



André Santanchè

<http://www.ic.unicamp.br/~santanche>

Licença

- Estes slides são concedidos sob uma Licença Creative Commons. Sob as seguintes condições: Atribuição, Uso Não-Comercial e Compartilhamento pela mesma Licença.
- Mais detalhes sobre a referida licença Creative Commons veja no link:
<http://creativecommons.org/licenses/by-nc-sa/3.0/>
- Agradecimento a Picture by Neal Fowler [<https://www.flickr.com/photos/31878512@N06/>] por sua fotografia “Explosion” usada na capa e nos fundos, disponível em [<https://flic.kr/p/oCNoe6>]. Vide licença específica da fotografia.