

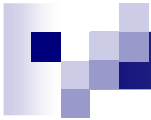


Curso de Java

by Antonio Rodrigues Carvalho Neto



Layouts



Layouts

- Os componentes gráficos podem ser dispostos no formulário exatamente na posição desejada definindo suas coordenadas cartesianas em *pixels*, através do método *setLocation(int X, int Y)*. Todo elemento gráfico que herda da classe *java.awt.Component* possui este método.
- Porém como a aplicação Java é portátil e pode rodar em qualquer plataforma, o ideal é utilizar outros meios que não posicionem os elementos diretamente.
- Para dispor os elementos gráficos na posição que desejamos de forma indireta teremos que usar um recurso chamado gerenciador de layout (*LayoutManager*).

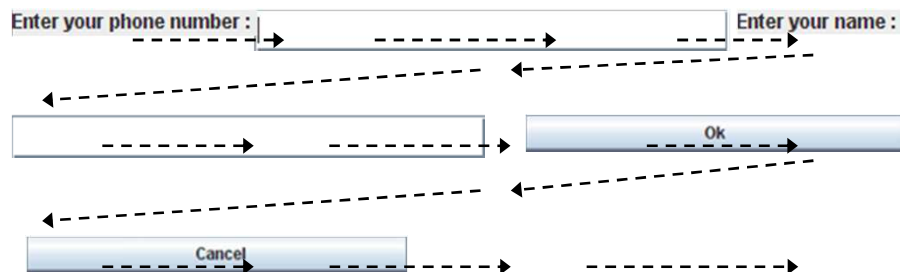


Layout Manager

- Os Layout managers definem como os componentes gráficos serão colados nos Containers.
- Existem diversos tipos de Layout Manager. Nos próximos slides serão mostrados os 3 tipos de *layouts* mais comuns (***FlowLayout***, ***BorderdLayout***, ***GridLayout***)

FlowLayout

- Este layout manager é o padrão definido nos elementos gráficos do tipo *Container*, sua definição rege que os componentes gráficos inseridos no *Container* serão dispostos um ao lado do outro (da esquerda para a direita) e de cima para baixo.



- Caso não exista mais espaço na mesma linha o *LayoutManager* cria mais uma linha.



FlowLayout

- Para usar este *LayoutManager* é preciso criar um objeto do tipo **FlowLayout**, e atribuí-lo ao *container* ou *JPanel* desejado.

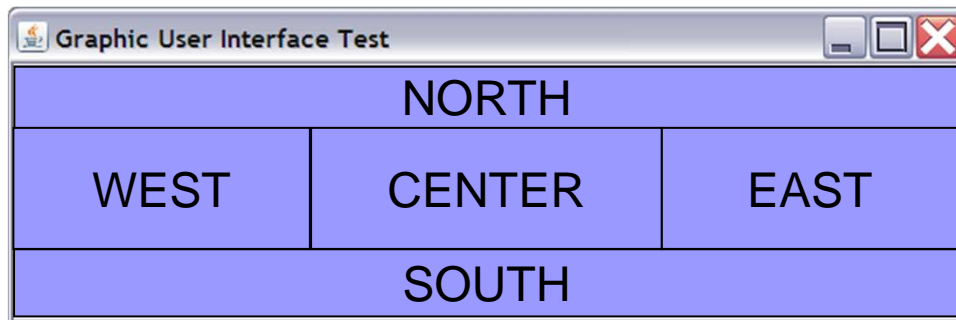
- Sintaxe do uso do FlowLayout:
 - Criar o LayoutManager
 - Sintaxe: `FlowLayout <objeto FlowLayout> = new FlowLayout();`
 - Exemplo: `FlowLayout flow1 = new FlowLayout();`

 - Atribuir o LayoutManager ao objeto JPanel ou *container*
 - Sintaxe: `<obj grafico container>.setLayout(<objeto FlowLayout>);`
 - Exemplo: `painel1.setLayout(flow1);`

 - Adicionar um componente gráfico
 - Sintaxe: `<obj grafico container>.add(<componente>);`
 - Exemplo: `painel1.add(lblHello);`

BorderLayout

- BorderLayout divide o painel em zonas (NORTH, SOUTH, WEST, EAST, CENTER). Dessa forma é possível posicionar os elementos gráficos, porém cada zona pode receber no máximo um componente gráfico.



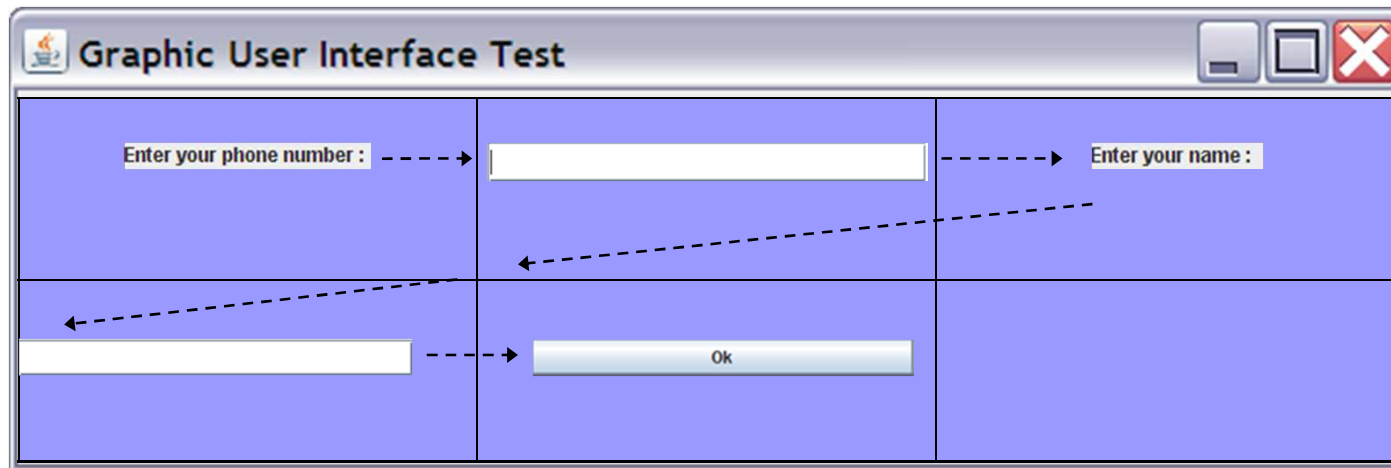


BorderLayout

- Para usar este *LayoutManager* é preciso criar um objeto do tipo **BorderLayout**, e atribuí-lo ao **container** ou **JPanel** desejado.
- Quando for adicionar um componente gráfico no **container** você precisará especificar em qual **zona** o componente será colocado
- Sintaxe do uso do BorderLayout:
 - Criar o LayoutManager
 - Sintaxe: **BorderLayout** <objeto BorderLayout> = new **BorderLayout**();
 - Exemplo: **BorderLayout** border1 = new **BorderLayout**();
 - Atribuir o LayoutManager ao objeto JPanel ou **container**
 - Sintaxe: <obj grafico container>.setLayout(<objeto BorderLayout>);
 - Exemplo: painel1.setLayout(border1);
 - Especificar a zona quando for adicionar o componente gráfico
 - Sintaxe: <obj grafico container>.add(<componente>, BorderLayout.<zona>);
 - Exemplo: painel1.add(lblHello, BorderLayout.NORTH);

GridLayout

- Grid Layout divide o painel em uma grade com linhas e colunas. Cada celula pode receber no máximo um componente gráfico. Os componentes são posicionados no Container formatado com Grid Layout da esquerda para a direita e de cima para baixo.





GridLayout

- Para usar este *LayoutManager* é preciso criar um objeto do tipo **GridLayout**, e atribuí-lo ao *container* ou *JPanel* desejado.

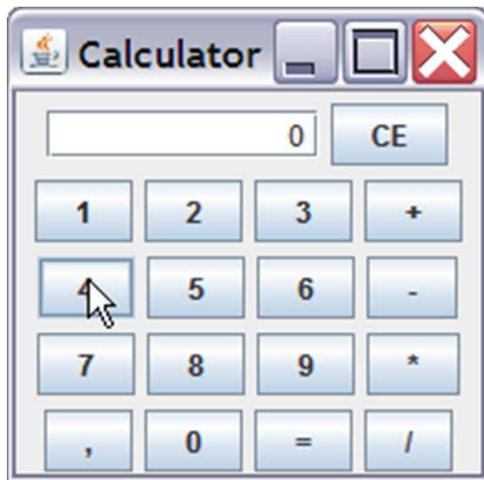
- Sintaxe do uso do GridLayout:
 - Criar o LayoutManager
 - Sintaxe: **GridLayout** <objeto BorderLayout> = new **GridLayout**(<lin> , <col>);
 - Exemplo: **GridLayout** grid1 = new **GridLayout**(3, 2);

 - Atribuir o LayoutManager ao objeto JPanel ou *container*
 - Sintaxe: <obj grafico container>.setLayout(<objeto GridLayout>);
 - Exemplo: painel1.setLayout(grid1);

 - Adicionar um componente gráfico
 - Sintaxe: <obj grafico container>.add(<componente>);
 - Exemplo: painel1.add(lblHello);

Exercício 3

- Faça uma tela gráfica que desenhe uma calculadora conforme uma figura abaixo



```
BorderLayout <objeto BorderLayout> = new BorderLayout();  
<obj grafico container>.setLayout( <objeto BorderLayout> );  
<obj grafico container>.add( <componente>, BorderLayout.<zona> );  
GridLayout <objeto BorderLayout> = new GridLayout();  
<obj grafico container>.setLayout( <objeto GridLayout> );  
<obj grafico container>.add( <componente>);
```



Referências

- Java como programar 6ª edição

Capítulo 12

- Use a cabeça Java 2ª edição

Capítulo 13:

pags. 282 a 293

pags. 297 a 300 (Exercícios)