

PROGRAMAÇÃO ORIENTADA A OBJETOS

Introdução a GUIs

Prof. Emanuel Barreiros



Interfaces gráficas

- Interfaces gráficas (GUIs) apresentam um mecanismo amigável para interação com aplicações
- GUIs são construídas a partir de componentes, comumente chamados de controles ou widgets
- Embora as IDEs forneçam ferramentas para criar as telas visualmente, muitas vezes o código gerado pelas ferramentas é bem diferente, logo, inicialmente vamos dar preferência a desenvolver a interface na mão

Entrada de dados simples

- Aplicações usam muito as caixas de diálogo para interagir com o usuário
- Diálogos são pequenas caixas usadas para informar o usuário de algo ou obter informações do usuário
- Para esse fim, podemos usar a classe JOptionPane
- Para testar, vê-la funcionando, faça o seguinte programa rodar:

Exemplo com JOptionPane

```
1 package br.upe;
2
3 import javax.swing.JOptionPane;
4
5 public class TesteJOP {
6
7     public static void main(String[] args) {
8         String numero1 = JOptionPane
9             .showInputDialog("Informe o primeiro inteiro");
10
11         String numero2 = JOptionPane
12             .showInputDialog("Informe o segundo inteiro");
13
14         int num1 = Integer.parseInt(numero1);
15         int num2 = Integer.parseInt(numero2);
16
17         int soma = num1 + num2;
18
19         JOptionPane.showMessageDialog(null, "A soma é: " + soma,
20             "Soma de dois inteiros", JOptionPane.PLAIN_MESSAGE);
21
22     }
23
24 }
```

Tipos de mensagem no JOptionPane

Tipo de mensagem	Descrição
ERROR_MESSAGE	Indica um erro (possui ícone próprio)
INFORMATION_MESSAGE	Indica uma mensagem informacional (possui ícone próprio)
WARNING_MESSAGE	Indica uma mensagem de aviso, algo que deve se dada atenção (possui ícone próprio)
QUESTION_MESSAGE	Indica uma mensagem de pergunta. (possui ícone próprio)
PLAIN_MESSAGE	Indica um diálogo com uma mensagem simples

Mostrando textos e imagens em uma janela

- Janelas em aplicações Java são instâncias da classe JFrame
- Quando construímos uma GUI, sempre precisamos associar os componentes a um container (JFrame, por exemplo)
- Você precisa também dizer onde cada componente deverá ser posicionado
- O Java fornece uma série de gerenciadores de layouts que posicionam os componentes de acordo com regras

Layouts

- No nosso exemplo, usamos o `FlowLayout`
- O `FlowLayout` posiciona os elementos da direita para a esquerda, na ordem que em que são adicionados ao container
- Quando componentes não cabem mais em uma linha, o container começa a posicionar componentes na próxima linha
- O método `setLayout` de cada container é usado para determinar o layout a ser usado

Exercícios

- Crie um programa que faça a conversão de temperatura de fahrenheit para celsius. Use o JOptionPane para obter dados do usuário e exiba o resultado usando um JLabel. A fórmula para conversão é a seguinte: $celsius = 5 / 9 * (fahrenheit - 32)$
- Escreva um joguinho de “advinhe o número”. Faça a aplicação escolher aleatoriamente um inteiro entre 1 e 1000. A cada tentativa do usuário, o programa informa se a jogada foi abaixo ou acima do valor a ser adivinhado. Eventualmente, o usuário acertará, nesse momento informe o usuário e encerre a aplicação.
 - OBS: use as seguintes linhas para obter o número aleatório
`Random rand = new Random();`
`int valor = 1 + rand.nextInt(1000);`