

Desenvolvimento de interfaces gráficas em JAVA com o pacote **SWING**

Ítalo Magno Pereira

POO 2018

Conteúdo

- Introdução ao pacote javax.swing
- Implementação de estudo de caso
 - Implementação da classe de modelo e de dados
 - Implementação da classe DAO com persistência em memória
 - Implementação de interface gráfica

Introdução ao pacote javax.swing

- O pacote javax.swing possui componentes que permitem a criação de interfaces gráficas ou GUI's (*Graphical User Interface*).
- GUI's fornecem mecanismos para que usuários interajam com aplicações de maneira mais fácil, através do uso de mouse, teclado ou outra forma de entrada.

Introdução ao pacote javax.swing

- As GUI's tem como componentes:
 - Barra de título (title bar);
 - Barra de menu
 - Menus
 - Botões
 - Caixas de combinação
 - Campos de texto.
 - Entre outros

Introdução ao pacote javax.swing

Componente	Descrição
<i>JLabel</i>	Exibe texto ou ícones que não podem ser editados.
<i>TextField</i>	Permite que o usuário entre com dados a partir do teclado. Também pode ser utilizado para exibir texto editável ou não.
<i>Button</i>	Dispara um evento quando acionado.
<i>CheckBox</i>	Especifica uma opção que pode ser selecionada ou não.
<i>ComboBox</i>	Fornecer uma lista drop-down de elementos em que o usuário pode selecionar um elemento ou possivelmente digitar no campo adequado.
<i>List</i>	Fornecer uma lista de itens em que o usuário pode selecionar um item clicando nele. Múltiplos itens podem ser selecionados.
<i>Panel</i>	Fornecer uma área em que os componentes podem ser colocados e organizados. Pode também ser utilizado como uma área para desenhar gráficos.

Introdução ao pacote javax.swing

- A interface gráfica mais simples é caixa de diálogo
- O pacote javax.swing fornece através da classe JOptionPane caixas de diálogo para entrada ou saída de dados.
- Apesar de permitir a entrada e saída de dados geralmente as interfaces gráficas são mais elaboradas.

Introdução ao pacote javax.swing

```
import javax.swing.JOptionPane;
```

```
public class Addition {  
    public static void main( String args[] )  
    {  
        String firstNumber; //primeira string digitada pelo usuário  
        String secondNumber; //segunda string digitada pelo usuário  
        int number1; //primeiro número  
        int number2; //segundo número  
        int sum; //soma  
  
        //lê o primeiro número como uma string  
        firstNumber = JOptionPane.showInputDialog("Enter first integer");  
  
        //lê o segundo número como uma string  
        secondNumber = JOptionPane.showInputDialog("Enter second integer");
```

Introdução ao pacote javax.swing

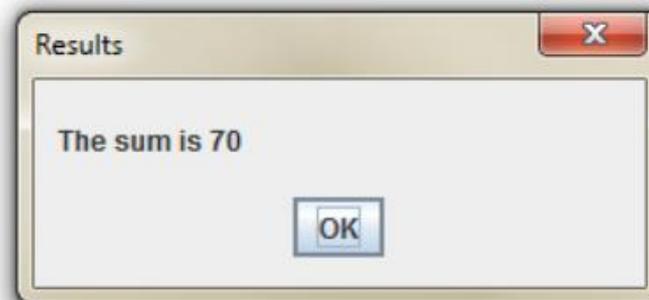
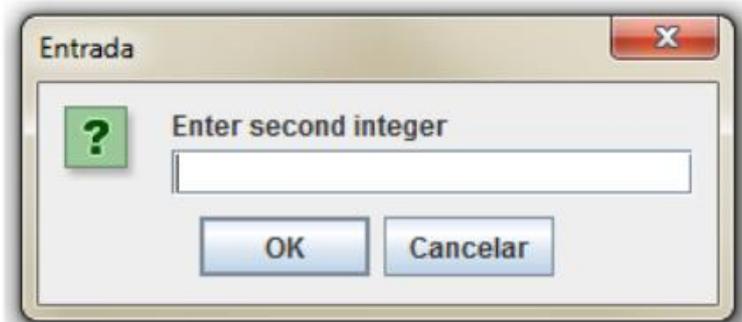
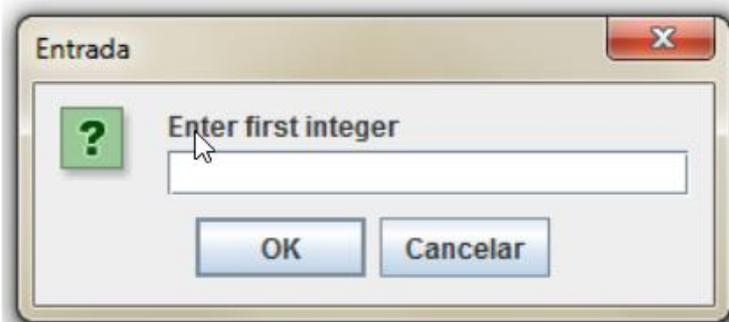
```
// converte os números de String para int
number1 = Integer.parseInt(firstNumber);
number2 = Integer.parseInt(secondNumber);

// adiciona os numeros
sum = number1 + number2;

// mostra o resultado
JOptionPane.showMessageDialog(null, "The sum is " + sum, "Results",
                              JOptionPane.PLAIN_MESSAGE);

System.exit( 0 );
}
```

Introdução ao pacote javax.swing



Introdução ao pacote javax.swing

JOptionPane Icons in Java Look and feel :



Error Message



Information Message



Question Message



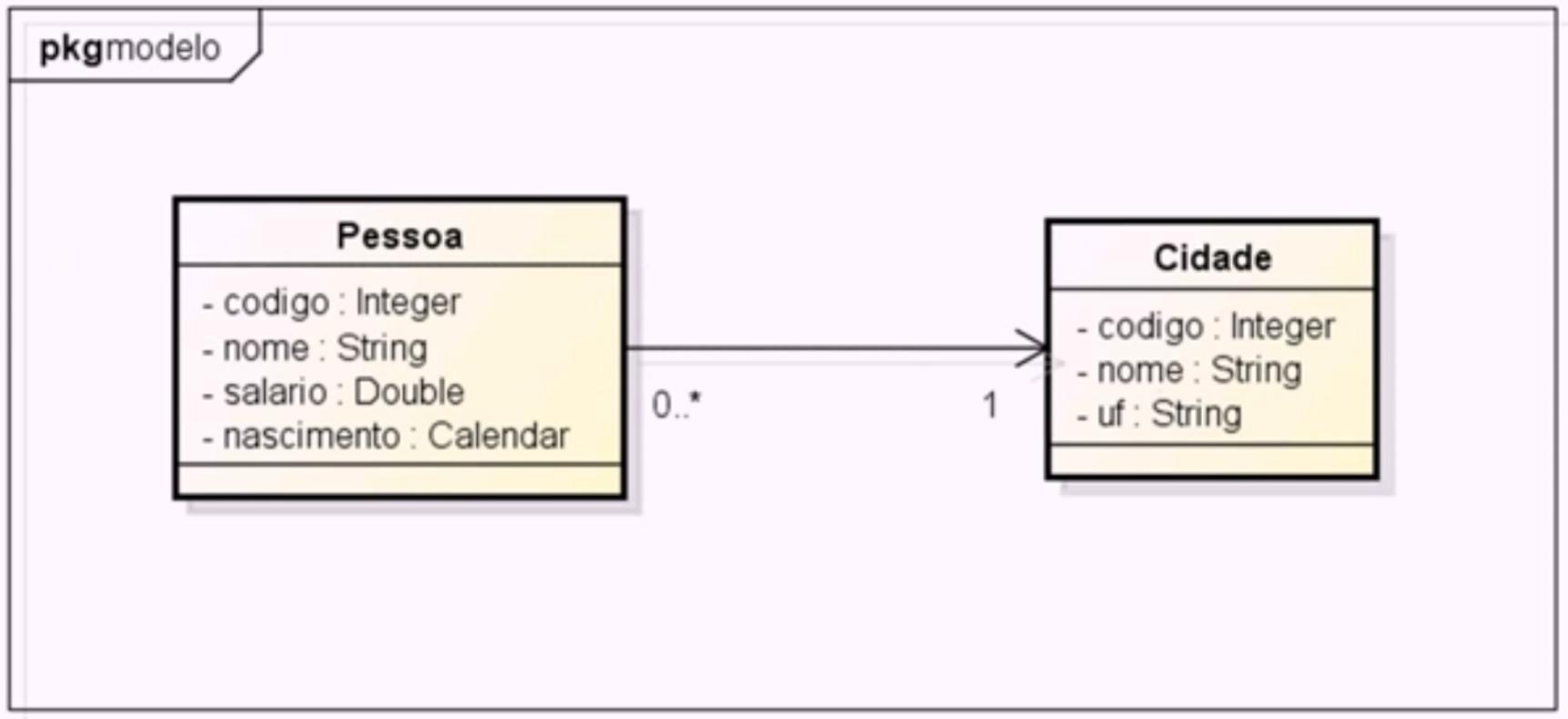
Warning Message

ERROR_MESSAGE	INFORMATION_MESSAGE
QUESTION_MESSAGE	WARNING_MESSAGE
PLAIN_MESSAGE	

Implementação de estudo de caso

- Definição de estudo de caso
- Implementação da classe de modelo e de dados
- Implementação da classe DAO com persistência em memória
- Implementação de interface gráfica

Definição de estudo de caso



Implementação da classe de modelo e de dados

- Para esta aula criares um CRUD para Cidades.
- Criação da classe Cidade com os atributos
 - Código
 - Nome
 - UF

Implementação da classe de modelo e de dados

- Implementação da classe dados com uma lista estática de cidades
 - `public static List<Cidade> listaCidades = new ArrayList<>();`

Implementação de classe DAO com persistência em memória

- Implementação da classe DAOCidade e seus métodos:
 - getLista
 - salvar
 - remover

Implementação de interface gráfica

- Criação do formulário principal
 - FormPrincipal (JFrame)
- Inserção dos componentes
 - Barra de menu
 - Itens de menu
 - Cadastros
 - Cidades
 - Ajuda
 - Sobre

Implementação de interface gráfica

- Criação do formulário de cadastro de cidades
 - FormCidade (JDialog)
 - Adicionar evento ao menu cadastro/cidade para abertura deste formulário
 - Adicionar um painel com Guias (abas)
 - Adicionar uma guia listagem
 - Adicionar uma tabela
 - » Adicionar a biblioteca “Vinculação de *beans*” para realizar a persistência dos dados
 - Adicionar uma guia dados
 - Adicionar botões de ação
 - » **Novo, Editar, Salvar, Cancelar, Excluir**
 - Adicionar campos e rótulos (*labels*)

Implementação de interface gráfica

- Criação do formulário de cadastro de cidades
 - Adicionar o objeto de vinculação “Resultado da consulta” e atribuir o código de criação personalizado
 - `org.jdesktop.observablecollections.ObservableCollections.observableList(new ArrayList<Cidade>())`
 - Realizar a vinculação da tabela com o objeto de vinculação.
 - Realizar a vinculação dos campos com a tabela.

Implementação de interface gráfica

- Criação do formulário de cadastro de cidades
 - Adição do método de atualização da tabela de exibição para chamada na abertura e alteração dos dados.
 - Definir a ação do botão novo.
 - Definir a ação do botão editar.
 - Definir a ação do botão salvar.
 - Definir a ação do botão cancelar.
 - Definir a ação do botão excluir.

Referências Bibliográficas

- http://www.decom.ufop.br/guillermo/BCC221/slides/13_Aula_13new.pdf
- <https://www.youtube.com/watch?v=hFcSUIVj8LY&index=4&list=PLcxA6SshISoYvQjNeznIYjK8Y9LxNDqFG>