APLICAÇÃO DA LINGUAGEM DE PROGRAMAÇÃO ORIENTADA À OBJETOS – ALPOO

Introdução a Interfaces Gráfica (Swing)

Prof. Danilo Pereira - danilo.pereira1@docente.unip.br

TÓPICOS DA AULA

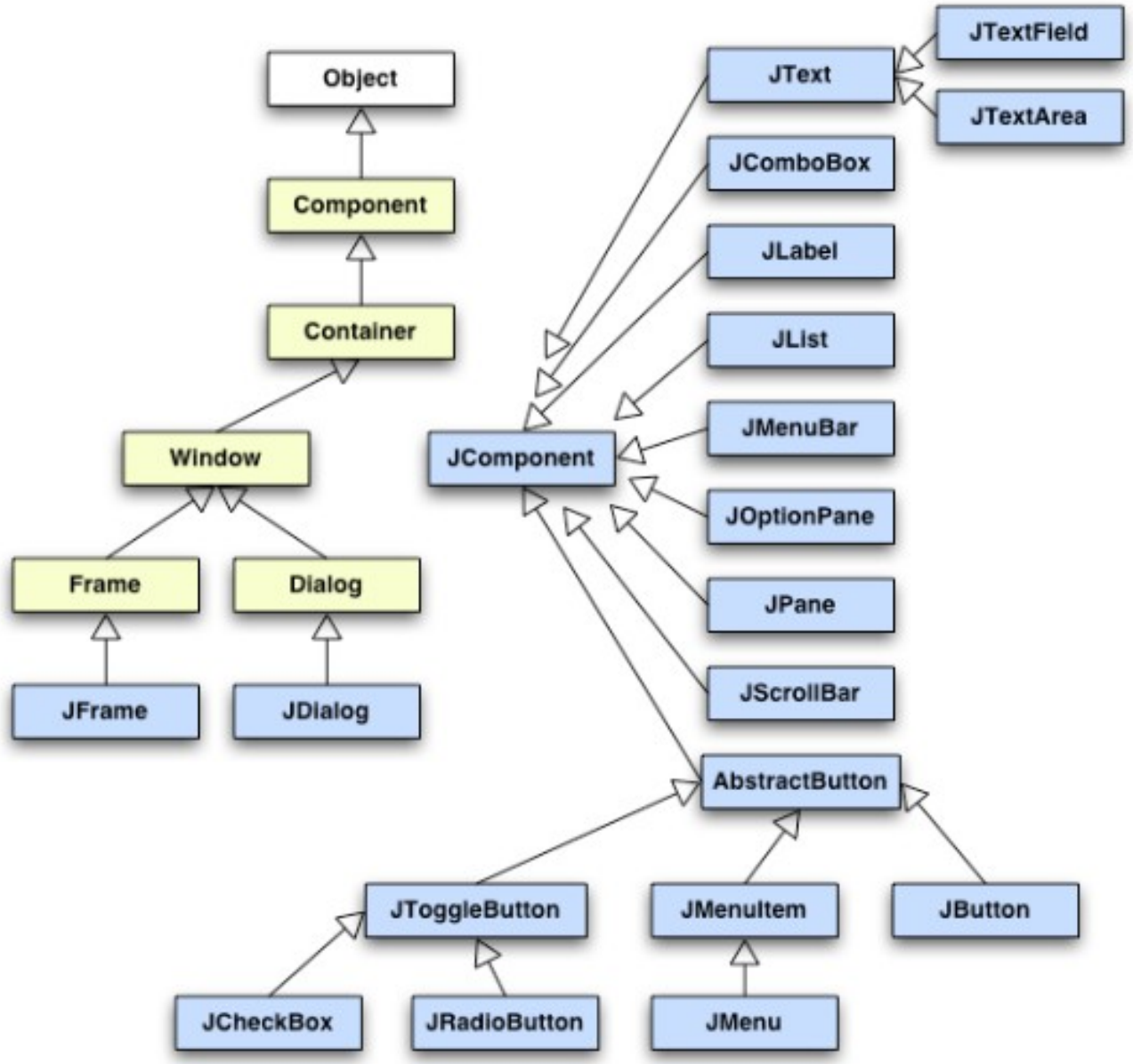
- Introdução a interface gráfica – Java Swing
- Entender as etapas de um desenvolvimento usando Java Swing
- Criar um exemplo usando JFrame e JOptionPane
- Java Swing usando NetBeans IDE e Eclipse IDE
- Exercícios de fixação

INTRODUÇÃO A INTERFACES GRÁFICAS EM JAVA

O desenvolvimento de uma interface GUI em Swing se baseia nos conceitos de janela e eventos

- Uma janela é um contêiner de objetos gráficos
- Os objetos devem ser anexados ao contêiner para que sejam exibidos
- Existem diferentes classes que podem representar uma janela, no entanto, a classe JFrame fornece o padrão de janela comum da maioria dos aplicativos GUI

ESTRUTURA HIERÁRQUIA – JAVA SWING



UM APLICATIVO GUI - SWING

1ª etapa:

- Criação da janela que conterá os demais objetos gráficos da aplicação

2ª etapa:

- Inserção dos componentes da interface

3ª etapa

- Tratamento de eventos

4ª etapa

- Lógica do programa

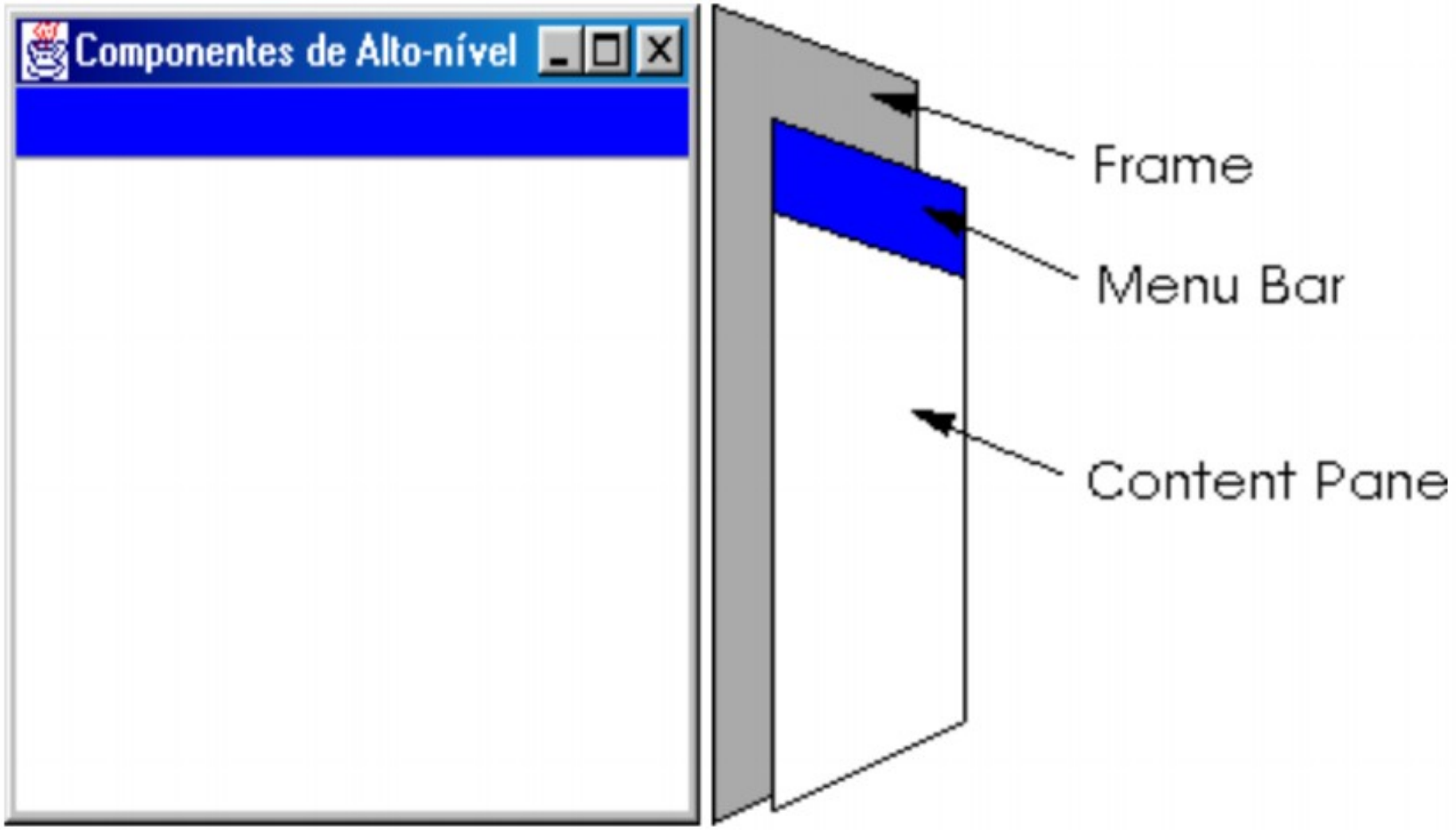
Containers

- Todo programa que oferece uma interface vai possuir pelo menos um container de mais alto nível, que pode ser:
  - JFrame: janela principal de um aplicativo
  - JDialog: janela para diálogos

UM APLICATIVO GUI - SWING

- JFrame: Um objeto desta classe possui uma barra de título e características para receber menus e outros componentes.
- JDialog: Usada para definir janelas de diálogo para entrada de dados. Normalmente usada em resposta a uma opção de menu selecionada. Definida em função de um objeto JFrame.

UM APLICATIVO GUI - SWING



UM APLICATIVO GUI - SWING

- 1ª etapa: Definição da janela que conterá os demais objetos gráficos da aplicação

```
public class ListaPresenca extends JFrame{
    public ListaPresenca( ){
        super("Lista de Presença");
    }
}
```

javax.swing.JFrame
Fornece uma janela com atributos padrão, como barra de títulos, botões de minimizar, maximizar e fechar.

UM APLICATIVO GUI - SWING

- 1ª etapa: Definição da janela que conterá os demais objetos gráficos da aplicação

```
public class ListaPresenca extends JFrame{
    public ListaPresenca( ){
        super("Lista de Presença");
    }
}
```

Chamada ao construtor de JFrame, para inicializar a janela com o título 'Lista de presença'

UM APLICATIVO GUI - SWING

- 1ª etapa: Definição da janela que conterá os demais objetos gráficos da aplicação

```
public static void main(String[] args){
   ListaPresenca janela = new ListaPresenca();
   janela.setSize(270,340);
   janela.setVisible( true );
}
```

UM APLICATIVO GUI - SWING

- 1ª etapa: Definição da janela que conterá os demais objetos gráficos da aplicação

```
public static void main(String[] args){
   ListaPresenca janela = new ListaPresenca();
   janela.setSize(270,340);
   janela.setVisible( true );
}
```

UM APLICATIVO GUI - SWING

- 2ª etapa: Inserção dos componentes da interface
- A bilioteca Swing fornece diversas classes que representam os elementos de interface padrão
  - Além disto, estas classes podem ser estendidas e novos componentes criados, personalizados.

JTextField

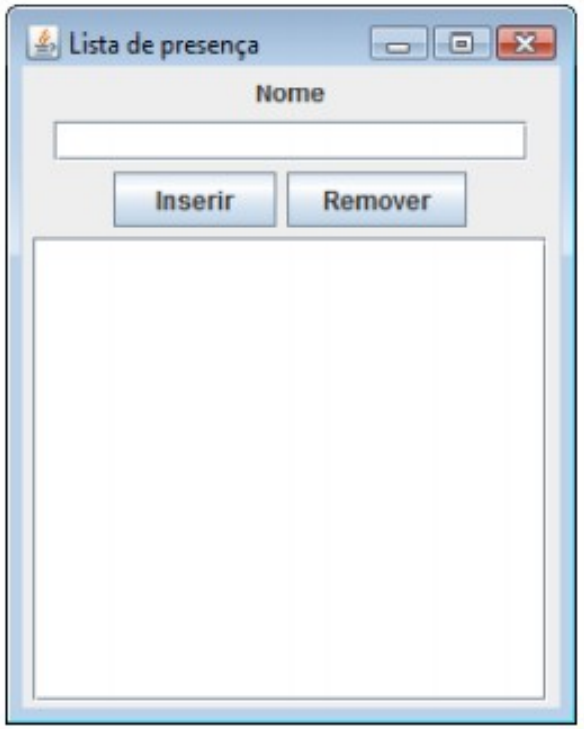
JButton

JList

JComboBox

UM APLICATIVO GUI - SWING

- 2ª etapa: Inserção dos componentes da interface
- Inserção dos componentes na Janela
  - LayoutManager: Gerencia o posicionamento dos componentes na janela



UM APLICATIVO GUI – SWING - LAYOUTS

- O arranjo dos componentes no container é gerenciado por um *LayoutManager*.
  - A vantagem da existência de um LayoutManager é que a apresentação dos componentes se adapta quando do redimensionamento da janela.
  - A desvantagem é o pouco domínio que o programador tem da posição dos componentes com alguns dos gerenciadores de layout.
- É possível definir seus próprios layouts, mas a linguagem oferece um conjunto de layouts básicos que simplificam o trabalho.

UM APLICATIVO GUI – SWING - LAYOUTS

- FlowLayout:
  - Coloca os componentes em uma fila da esquerda superior do container para a direita.
  - Respeita o tamanho preferido dos componentes mesmo quando não houver espaço suficiente no container.
  - É o padrão do JPanel.

```
JPanel c =new JPanel();
c.add(new JButton("1"));
c.add(new JTextField(9));
c.add(new JButton("dois"));
c.add(new JButton("três"));
```

UM APLICATIVO GUI – SWING - LAYOUTS

- Exemplo:

```
import javax.swing.*;
import java.awt.*;
public class TestaFlowLayout{
  public static void main (String args[]){
    int i;
    JFrame janela = new JFrame(”Testa FlowLayout");
    janela.setBounds(50, 100, 400, 150);
    janela.setDefaultCloseOperation( JFrame.EXIT_ON_CLOSE);
    FlowLayout flow = new FlowLayout(); // Define o layout do
     container
    Container caixa = janela.getContentPane(); // Define o container
    caixa.setLayout(flow); // Seta layout do container
    for (i=1; i<=6; i++) caixa.add(new JButton("Aperte " + i));
    janela.setVisible(true);
  }
}
```

UM APLICATIVO GUI – SWING - LAYOUTS

- GridLayout:
  - Divide o container em linhas e colunas.
  - Coloca os componentes da esquerda para a direita, de cima para baixo.
  - Todos os componentes terão o mesmo tamanho.
  - Permite definir um vetor ou matriz de células onde com componentes serão colocados.
  - Não respeita o tamanho original dos componentes.

```
JPanel c =new JPanel();
c .setLayout(new GridLayout(2,2));
c.add(new JButton(”um"));
c.add(new JTextField(5));
c.add(new JButton("dois"));
c.add(new JButton("três"));
```

Um aplicativo GUI - Swing

- 3ª etapa: Tratamento de eventos
- Os eventos permitem a interação entre usuários e interface, permitindo que o programa execute em função das ações do usuário
  - Movimentação de mouse
  - Pressionar teclas
  - Clicar em botões
  - Selecionar itens

UM APLICATIVO GUI – SWING

- 3ª etapa: Tratamento de eventos
  - Classes do pacote java.awt.event
  - Os componentes de interface disparam rotinas ao 'ouvir' um evento
  - Define-se uma classe que serve como 'tratador de eventos' de determinados objetos
  - Quando acionada, esta classe executa o método associado ao evento para tratá-lo
  - Diferentes objetos podem se vincular ao mesmo tratador de eventos

UM APLICATIVO GUI – SWING

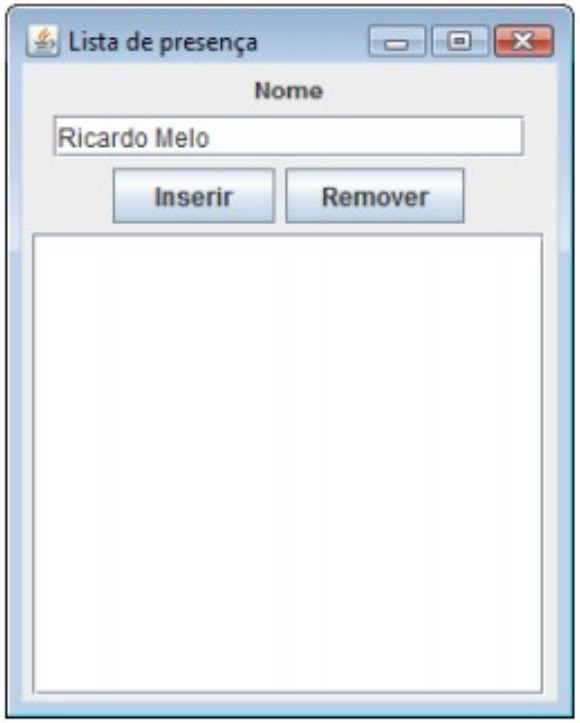
- 3ª etapa: Tratamento de eventos
- Interfaces comuns para o tratamento de eventos
  - **ActionListener** – eventos de ações do usuário como cliques em botões ou alteração de campos texto
  - **ItemListener** – eventos relacionados à manipulação de itens em lista de itens
  - **MouseListener** – eventos associados à ação do usuário no programa através do mouse

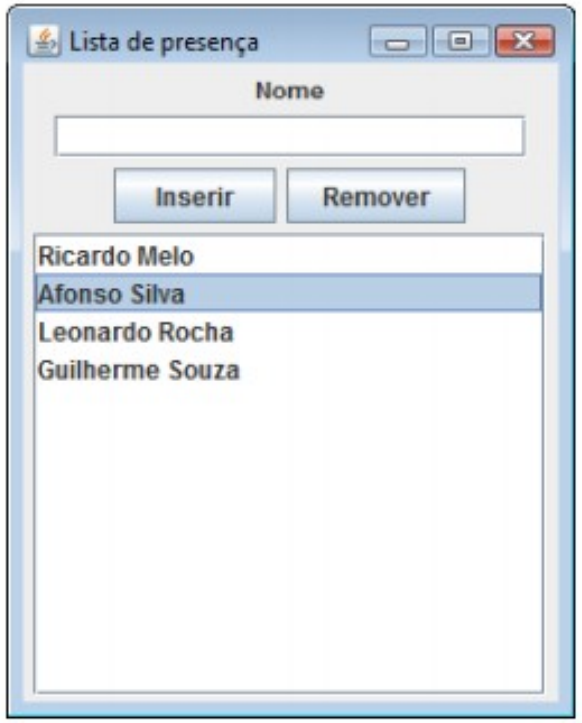
UM APLICATIVO GUI – SWING

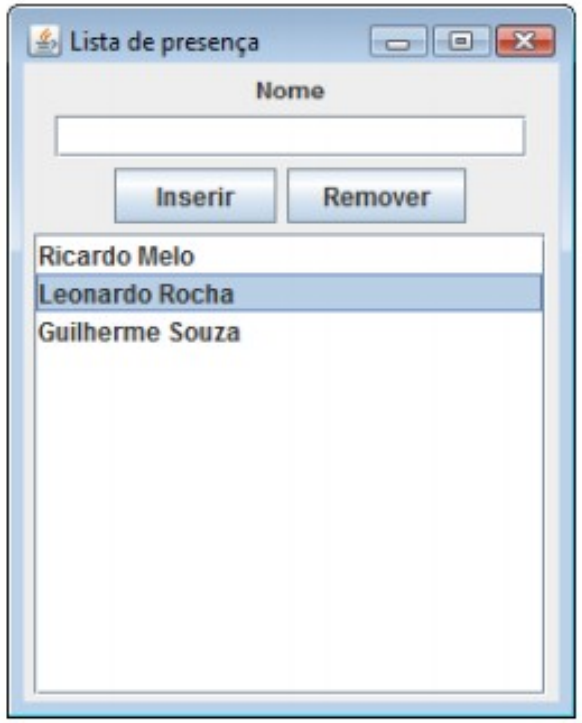
- 4ª etapa: Lógica do programa
- A aplicação é concluida após inserir a lógica do programa nos os eventos de clique nos botões

UM APLICATIVO GUI – SWING

- 4ª etapa: Lógica do programa
- A aplicação é concluida após inserir a lógica do programa nos os eventos de clique nos botões

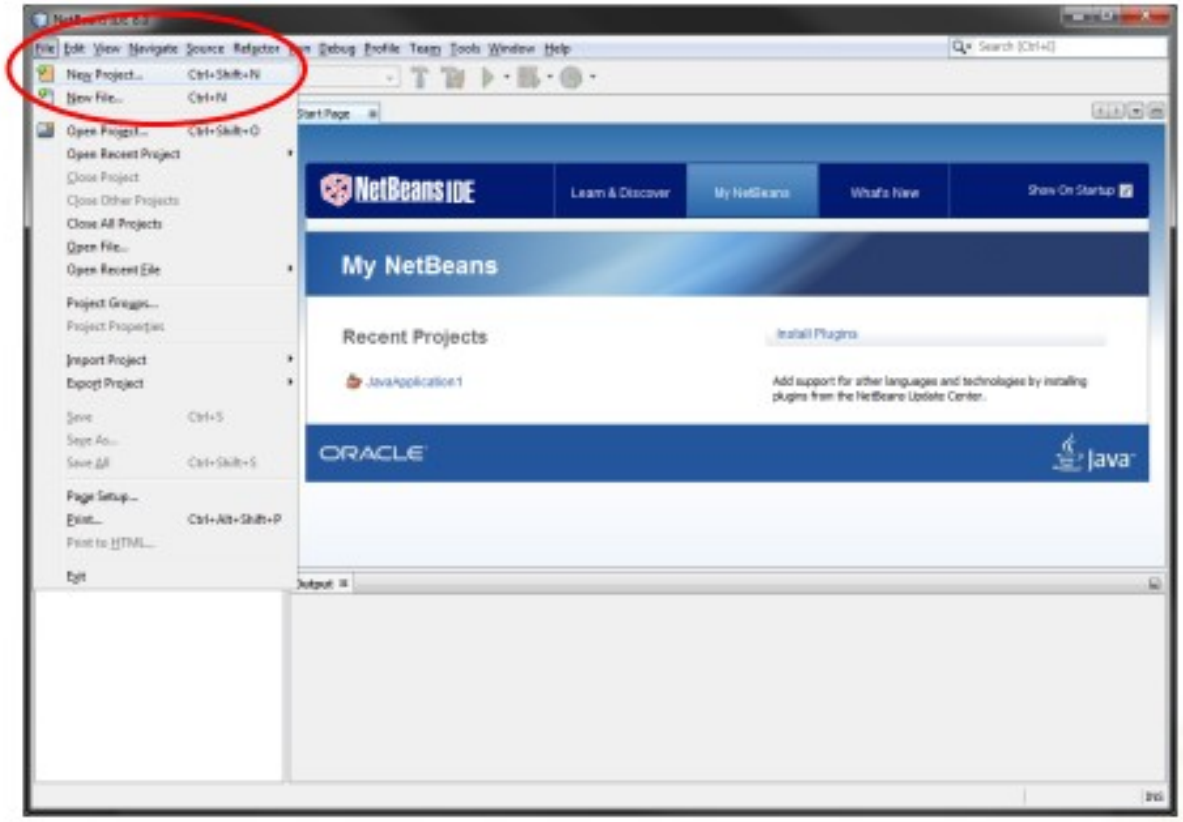






Netbeans GUI Builder – Criando Projeto

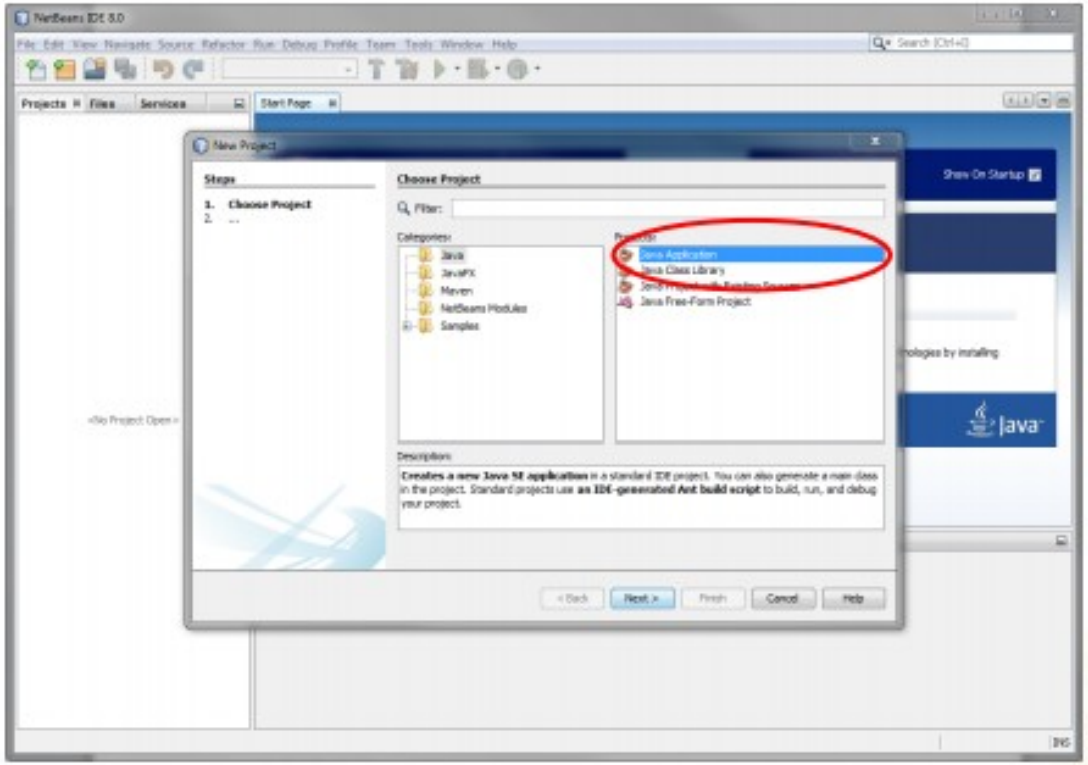
1) Acesse o menu File->New Project...



JAVA SWING USANDO NETBEANS IDE

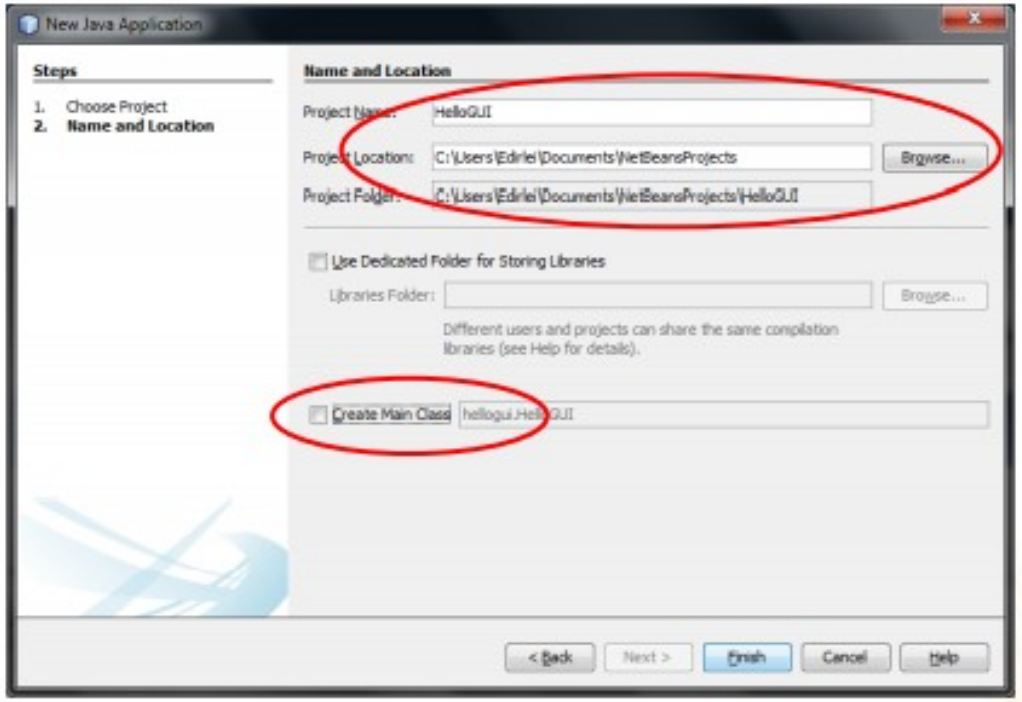
Netbeans GUI Builder – Criando Projeto

2) Selecione o tipo de projeto “Java Application” e em seguida clique em “Next”:



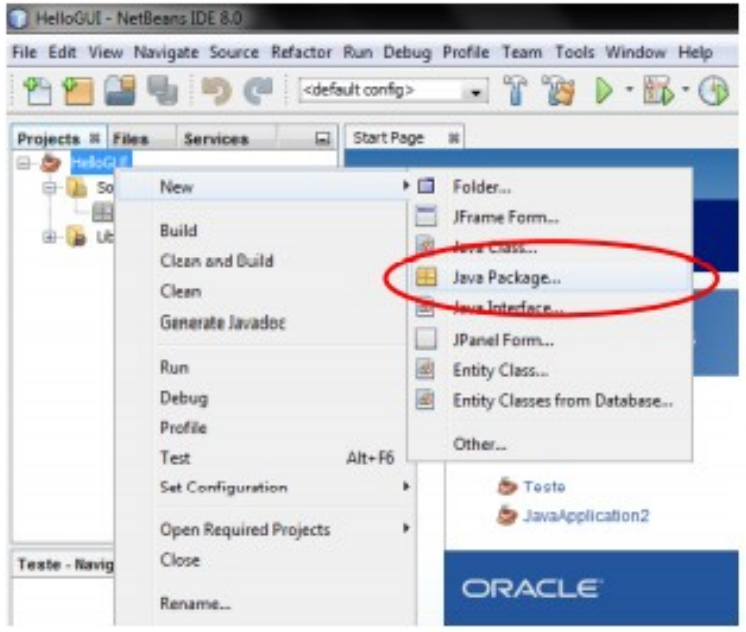
Netbeans GUI Builder – Criando Projeto

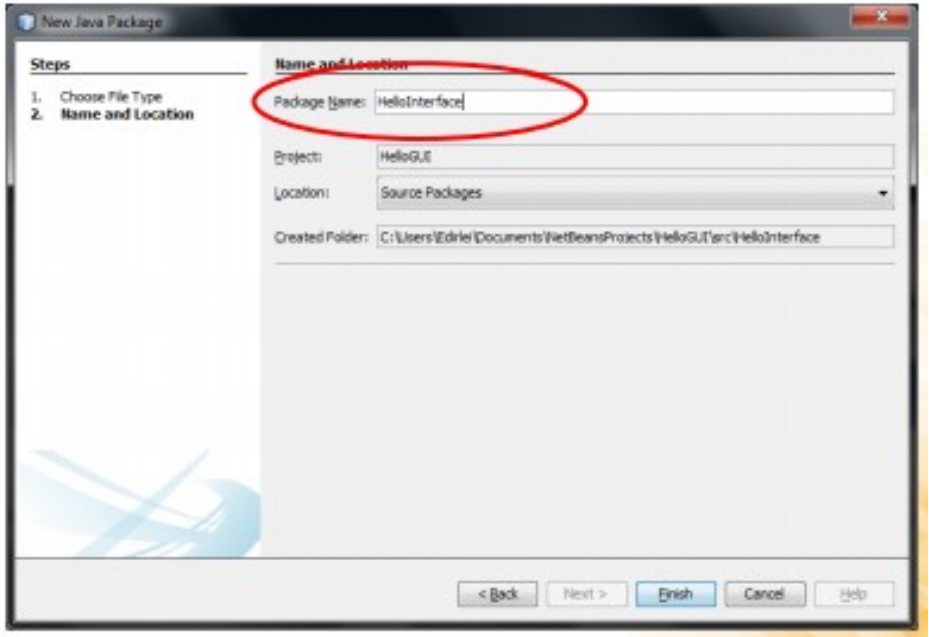
3) De um nome para o projeto, selecione o local onde ele será salvo e desmarque a opção “Create Main Class”. Em seguida clique em “Finish”:



Netbeans GUI Builder – Criando Projeto

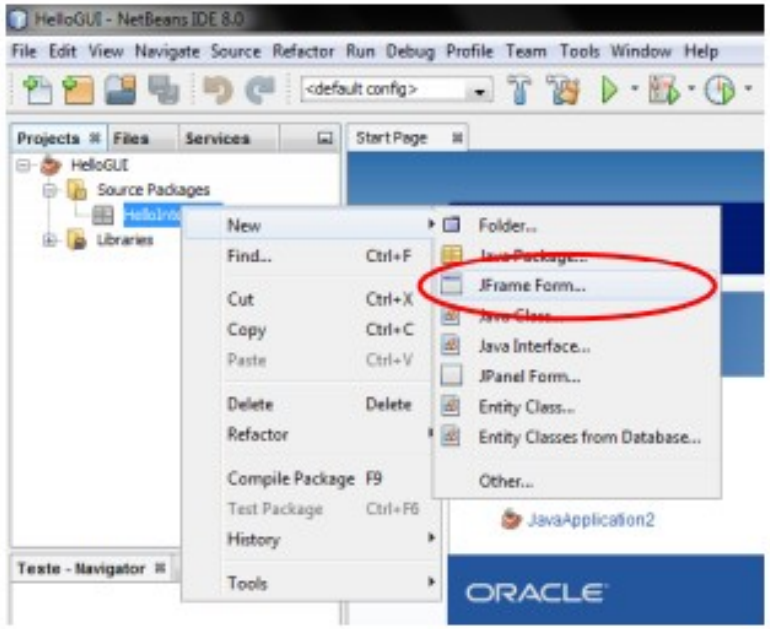
4) Crie um novo "Java Package" no projeto:

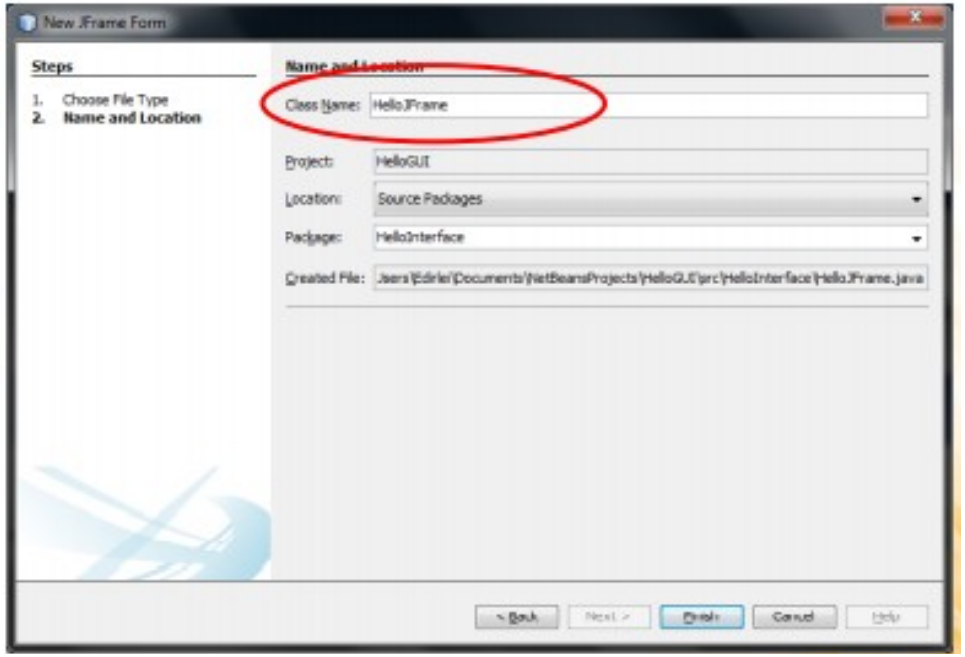




Netbeans GUI Builder – Criando Projeto

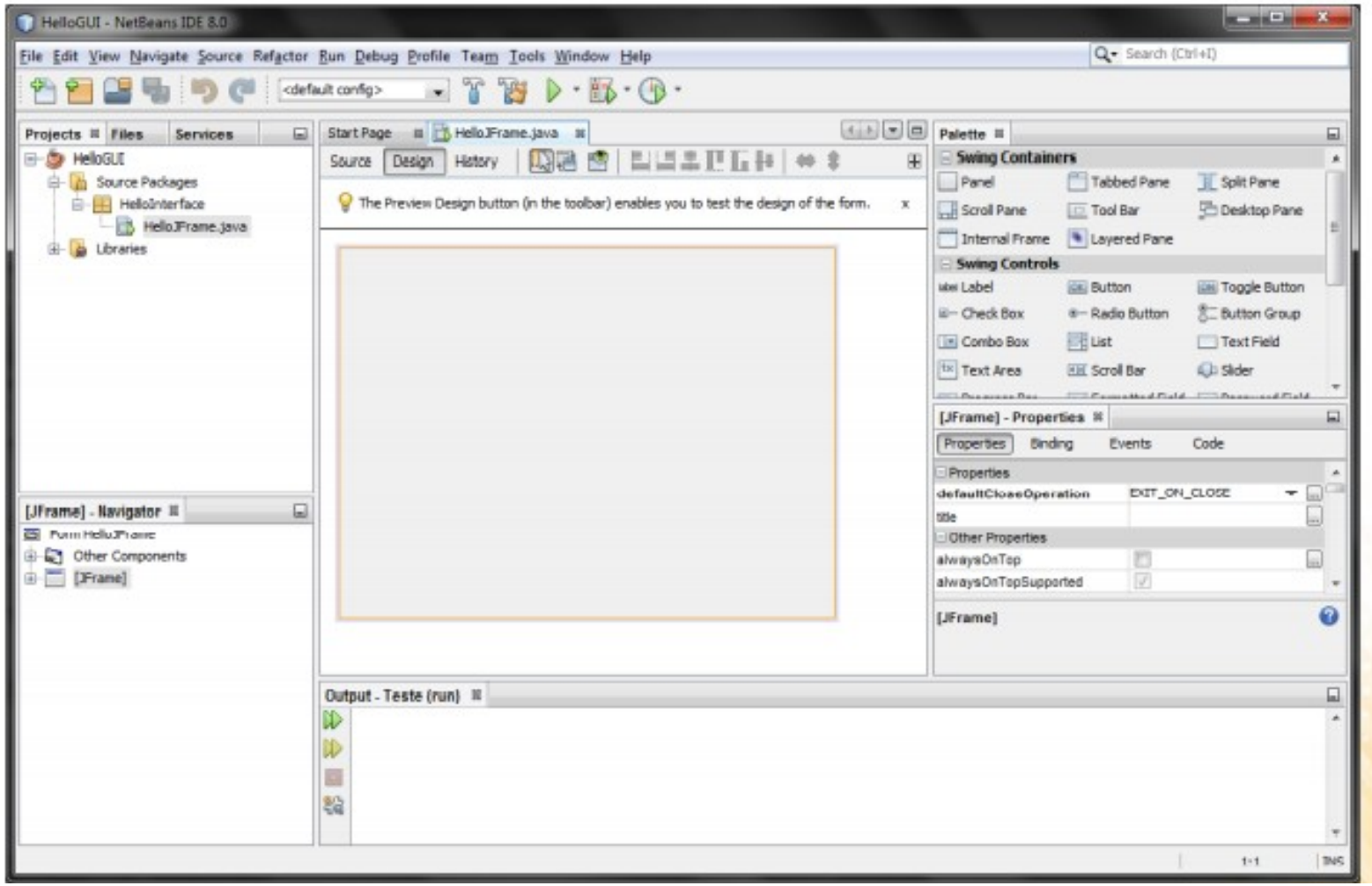
5) Crie um novo "JFrame Form":





JAVA SWING USANDO NETBEANS IDE

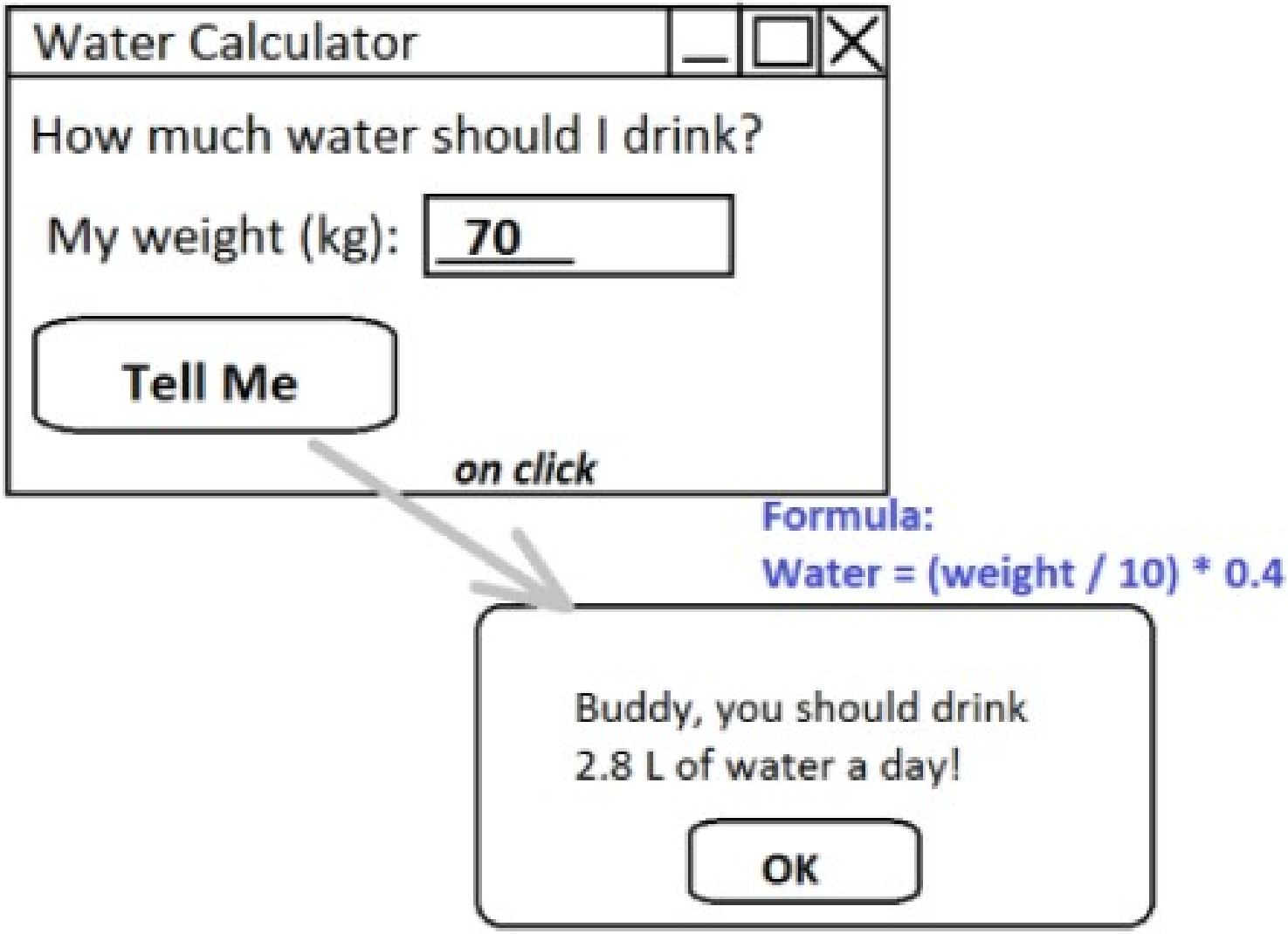
Netbeans GUI Builder



Tutorials – Java Swing

- **Java Code**
  - https://www.codejava.net/swing-tutorials
- **Java T Point**
  - https://www.codejava.net/swing-tutorials
- **Java GUI Tutorial - Make a GUI in 13 Minutes**
  - https://www.codejava.net/swing-tutorials

EXERCÍCIOS - Water Calculator – Java Swing



EXERCÍCIOS - Water Calculator – Java Swing

Este é um pequeno programa de Swing que permite ao usuário inserir seu peso (kg) e calcular a quantidade de água que deve beber todos os dias, de acordo com a fórmula fornecida. Quando o usuário clica no botão Tell Me, uma caixa de diálogo de mensagem aparece informando a quantidade exata de água.

Nesse exemplo veremos:

- Como criar uma janela usando a classe JFrame.
- Como usar o rótulo usando a classe JLabel.
- Como usar o campo de texto usando a classe JTextField.
- Como usar o botão usando a classe JButton.
- Como organizar componentes no quadro usando **FlowLayout** - um gerenciador de layout simples.
- Como lidar com o evento de clique de um botão.
- Como mostrar uma caixa de mensagem usando a classe JOptionPane.