

LPVS2 – Linguagem de Programação Visual

Prof. Edilson José Davoglio Candido

E-mail: edilson.ifsp@gmail.com

Prof. Fernando Vieira Duarte

E-mail: fernandoduarte@ifsp.edu.br

Aula 4 – Utilização de interface gráfica

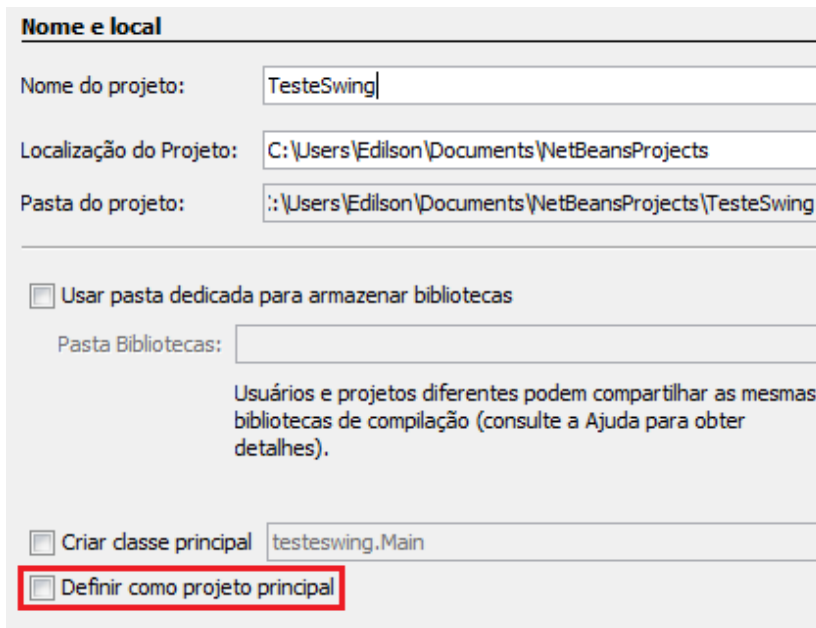
1. Swing

Swing é a tecnologia Java utilizada para o desenvolvimento de aplicações baseadas em desktop que necessitam de uma interface gráfica com o usuário (GUI – *Graphical User Interface*). O pacote utilizado é o `javax.swing`.

1.1. Criando a Interface Gráfica

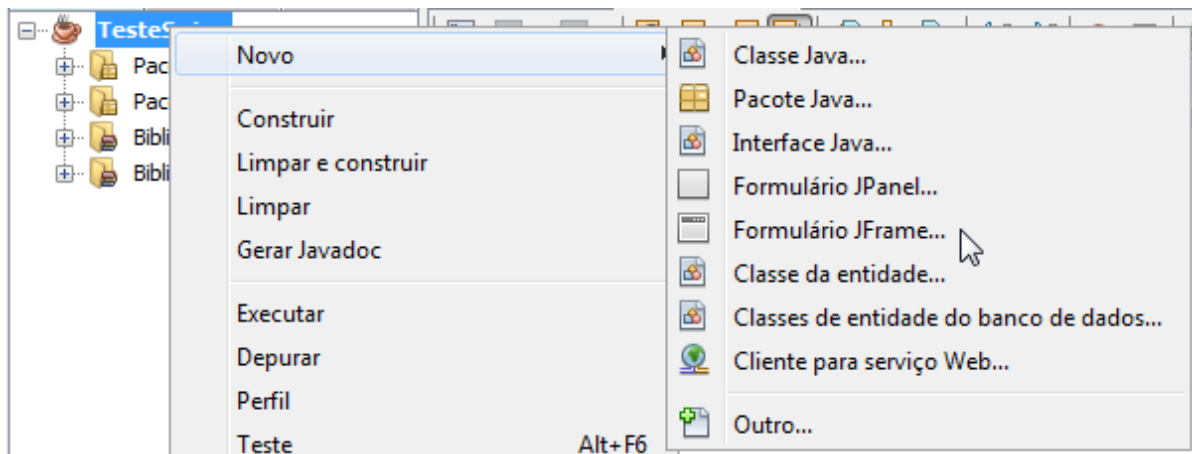
Geralmente, o componente `JFrame` é o *container* principal de uma aplicação Swing. Após criá-lo, é possível inserir os demais componentes para criar a interface gráfica.

O primeiro passo para elaborar um aplicativo Swing no Netbeans é criar um novo projeto. Para isso, vá até o menu “Arquivo” e selecione a opção “Novo Projeto...”. Após isso, clique na categoria “Java” e escolha o tipo de projeto “Aplicativo Java”. Especifique um nome para o projeto e **remova a opção “Definir como projeto principal”¹**. A opção deve ser removida porque queremos que o formulário que será criado posteriormente seja o ponto de partida quando a aplicação for executada.



Após criar um novo projeto já é possível adicionar um `JFrame`. Para isso clique com o botão direito do mouse sobre o nome do projeto, escolha “Novo” e logo depois selecione “Formulário `JFrame`”.

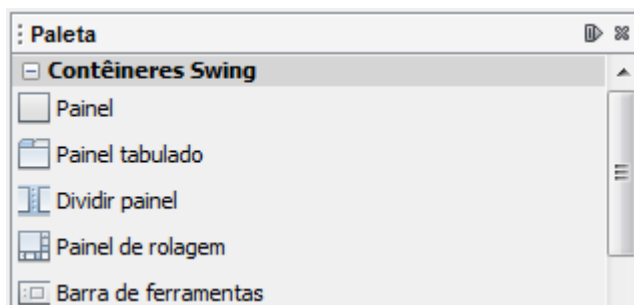
¹ Em versões mais recentes do NetBeans, é exibida a opção “Criar Classe Principal”. Desmarque essa opção.



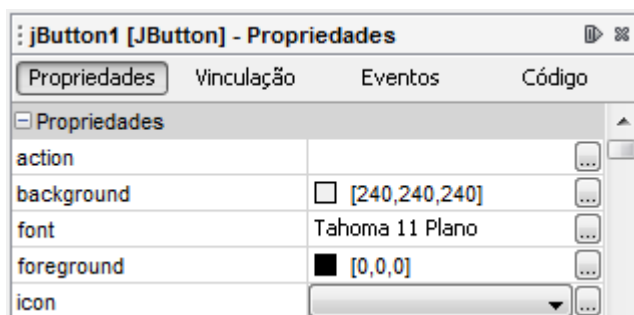
A seguir, informe o nome da classe que será criada e defina também o pacote no qual essa classe ficará armazenada.

Nome e local	
Nome da classe:	Exemplo1
Projeto:	TesteSwing
Localização:	Pacotes de código-fonte
Pacote:	testeswing

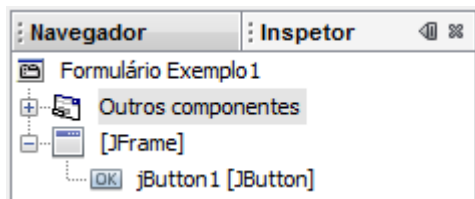
Note que três janelas foram adicionadas ao IDE. São elas: Paleta de Componentes, Propriedades e Inspetor. A Paleta de Componentes exibe uma lista com todos os componentes Swing, AWT, Beans e Java Persistence.



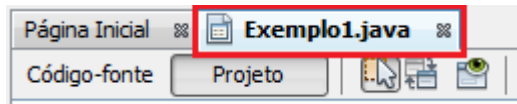
A Janela de propriedades exibe as propriedades do componente que estiver selecionado naquele momento. No exemplo abaixo são exibidas algumas das propriedades de um componente JButton.



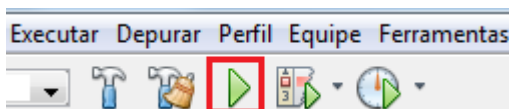
O componente Inspetor exibe todos os componentes da aplicação em uma hierarquia de árvore.



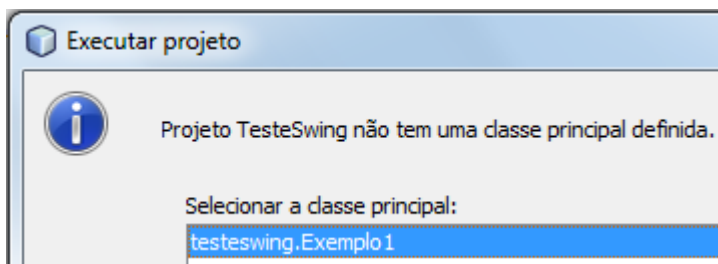
O formulário e a classe “Exemplo1” foram criados. Você pode navegar entre eles escolhendo as opções “Código-fonte” ou “Projeto”. A ideia é bastante intuitiva. O formulário é utilizado para elaborar a interface gráfica e o código fonte permite que as funcionalidades associadas com os componentes sejam implementadas.



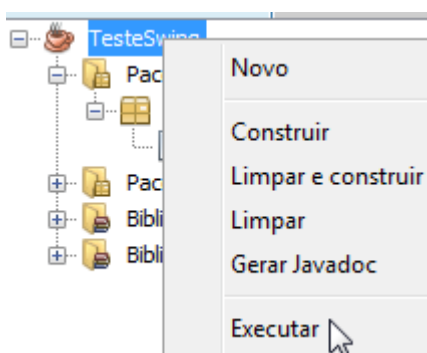
O próximo passo é testar se o projeto está funcionando. No menu, clique no botão ▶. Outra maneira é clicar na opção “Executar” e depois selecionar “Executar Projeto”.



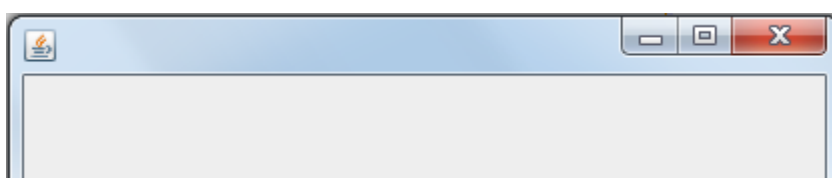
Uma mensagem será exibida solicitando que a classe principal do projeto seja definida. Selecione qual das classes será a principal e clique em “Ok”.



Atenção. Caso você encontre dificuldades para executar o projeto, clique com o botão direito no nome do projeto e a seguir selecione a opção “Executar”.



O formulário é então, exibido. Como nenhum componente foi adicionado, temos apenas uma janela com os botões de minimizar, maximizar e fechar.

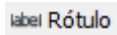


1.2. Componentes da Interface Gráfica

A seguir são listados alguns dos componentes que podem ser utilizados para a criação de uma interface gráfica através do Swing.

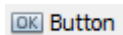
- JLabel (Rótulo)

O JLabel representa um rótulo, ou seja, uma área de texto não editável. Ele pode exibir texto ou imagens. Para alterar o texto do JLabel o método `setText()` pode ser utilizado. A string contendo o novo texto deve ser passada como parâmetro. É possível também recuperar o texto do JLabel utilizando-se o método `getText()`.



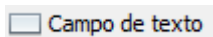
- JButton (Botão)

O JButton representa um botão. Ele é conectado a um gerenciador de eventos que executa o código associado a ele quando o JButton é clicado.



- JTextField (Campo de texto)

Representa um campo de texto, permitindo que você escreva e edite uma única linha de texto.



- JOptionPane

Representa uma caixa de diálogo que pode ser utilizada para exibir mensagens, solicitar confirmações do usuário ou obter informações de entrada.

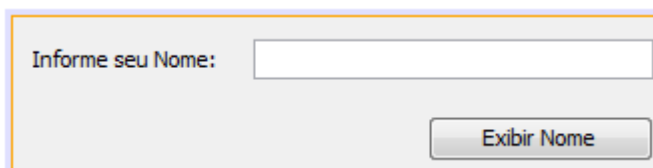
- Exibir Mensagens

O método utilizado é o `showMessageDialog()`.

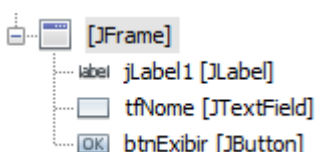
1.3. Exercício Resolvido

Elabore uma interface gráfica na qual o usuário possa informar seu nome. Ao clicar no botão “Exibir Nome”, a informação inserida no campo de texto deve ser exibida.

Adicione três componentes ao formulário: rótulo, campo de texto e button.



Altere o nome dos componentes `jTextField1` e `jButton1`. No inspetor, clique com o botão direito sobre o nome da variável `jTextField1` e a seguir, clique na opção “Alterar o nome da variável...”. Informe o novo nome e clique no botão “Ok”. O procedimento para alterar o nome da variável `jButton1` é idêntico.



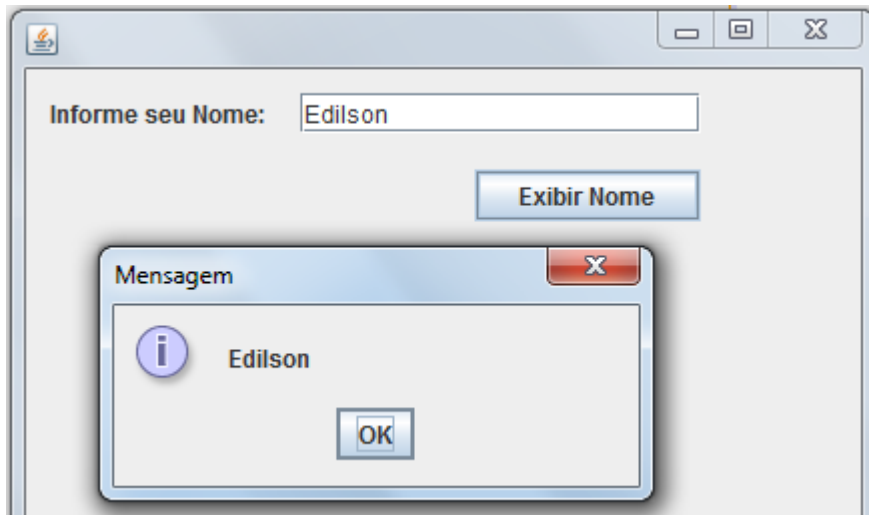
Após isso, dê um duplo clique no botão “Exibir Nome”. Note que o método associado com a ação de clicar no botão foi criado de forma automática.

```
private void btnExibirActionPerformed(java.awt.event.ActionEvent evt) {  
  
}
```

Adicione ao método a seguinte linha:

```
JOptionPane.showMessageDialog(null, tfNome.getText());
```

O código acima exibe uma janela com o conteúdo digitado no campo tfNome. Veja o exemplo a seguir:



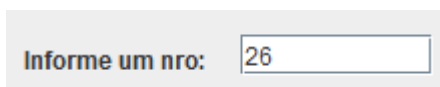
Note que a classe JOptionPane possui três métodos “showMessageDialog”. Posteriormente discutiremos isso.

1.4. Convertendo String para Número

No exercício acima, utilizamos apenas String (caracteres). Mas e no caso de números? É possível utilizar um JTextField? A resposta é sim.

Veja a seguir:

Exemplo 1:



Para efetuar a leitura do número “26” a partir do JTextField para uma variável do tipo inteiro, a seguinte linha deve ser utilizada:

```
int nro;  
nro = Integer.parseInt(tfNro1.getText());
```

No código acima, estamos utilizando a classe Integer. Ela permite que o tipo primitivo int seja tratado como um objeto. A classe Integer fornece vários métodos, dentre os quais está o parseInt. Esse método é capaz de converter um valor do tipo String para int.

Observe que o método parseInt recebe um String como parâmetro (obtido através do método getText()) e converte esse String para um inteiro.

De forma semelhante, temos:

```
Byte.parseByte(String s);  
Short.parseShort(String s);  
Float.parseFloat(String s);
```

```
Long.parseLong(String s);  
Double.parseDouble(String s);
```

2. Exercícios

- 2.1. Crie um programa que simule uma calculadora. O usuário deverá informar dois números e especificar umas das seguintes operações (Adição, Subtração, Divisão, Multiplicação).
- 2.2. Crie uma calculadora de reembolso para viagens de negócio. Ela deverá permitir ao usuário informar a quilometragem inicial e a final. A partir desses números, a distância será calculada e o valor do reembolso informado. Considere que a empresa paga R\$ 2,83 para cada quilômetro rodado. Insira um botão que informe a quantidade de quilômetros rodados, a partir da quilometragem inicial e final.
- 2.3. Escreva um programa que seja capaz de converter um valor informado em reais para dólares ou euros. Além disso, o programa deve permitir que os valores de conversão sejam ajustados.
- 2.4. João recebeu seu salário e precisa pagar duas contas que estão atrasadas. Como as contas estão atrasadas, João terá de pagar multa de 2% sobre cada conta. Faça um programa que receba o salário do João e o valor das contas, também calcule e mostre quanto restará do salário do João.
- 2.5. Sabe-se que para iluminar de maneira correta os cômodos de uma casa, para cada m^2 , deve-se usar 18 W de potência. Faça um programa que receba as dimensões de um cômodo (em metros), calcule e mostre a sua área (em m^2) e a potência de iluminação que deverá ser utilizada.
- 2.6. Crie um programa capaz de automatizar o processo de gerenciamento das diárias de um Hotel. É possível informar a quantidade de diárias, se o serviço de táxi do hotel será utilizado (encarece em R\$ 60,00 o preço final) e se o cliente deseja um flat simples ou de luxo.
 - Flat Simples: R\$ 120,00 a diária
 - Flat Luxo: R\$ 190,00 a diária
- 2.7. Faça um programa que receba:
 - O código do produto comprado;
 - A descrição do produto comprado;
 - A quantidade em estoque do produto comprado;
 - O preço unitário do produto comprado e;
 - A quantidade comprada do produto.Calcule e mostre:
 - O valor total da nota;
 - O valor do desconto, seguindo a tabela I e aplicado sobre o valor total da nota e;
 - O valor final da nota depois do desconto.

Valor total da nota	% de desconto
Até R\$ 250,00	5%
Entre R\$ 250,00 e R\$ 500,00	10%
Acima de R\$ 500,00	15%

