



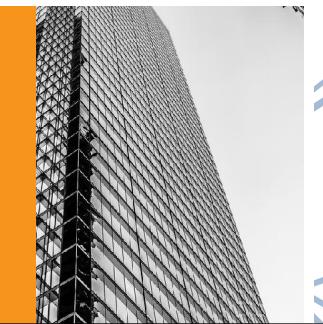
**Desenvolver e organizar interface
gráfica para aplicações desktop**

INTRODUÇÃO A JAVA SWING

Rede
Fecomércio RS
de Educação

Instalação IDE

Rede
Fecomércio RS
de Educação



Integrated Development Environment (IDE)

O que é?

É um software que reúne ferramentas essenciais para programadores (editor de código, compilador, depurador, automação) em uma única interface gráfica, facilitando e acelerando a criação de aplicações, desde a escrita do código até os testes finais.

Integrated Development Environment (IDE)

Componentes Principais

- Editor de Código-Fonte
- Compilador/Interpretador
- Depurador (Debugger)
- Automação de Build
- Integração com Controle de Versão (GIT)

Instalação

Acesse: <https://netbeans.apache.org/front/main/download/nb21/>

Procure por :

Installers and Packages:

- Apache-NetBeans-21-bin-windows-x64.exe (SHA-512, PGP ASC)

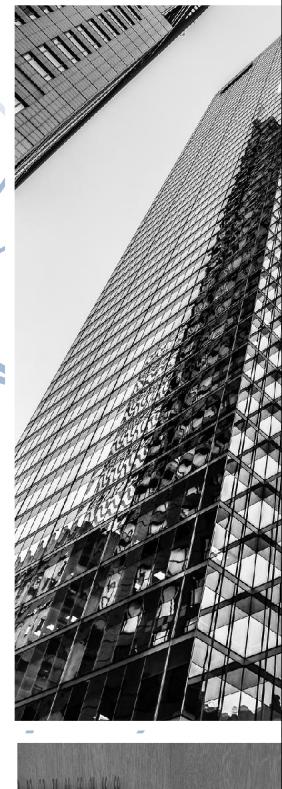
Baixe e Instale.

Next em tudo na janela de instalação e por fim, “install”.

Qualquer problema, consultar o professor

JAVA SWING

Rede
Fecomércio RS
de Educação



O que é Java?

- Linguagem de Programação de alto nível
- Sintaxe clara
- Orientada a objetos
- Plataforma robusta
- WORA "Write Once, Run Anywhere"
- Estaticamente tipada (Forte)

Principais Usos

- Desenvolvimento Android: A base para criar aplicativos nativos para smartphones e tablets Android.
- Aplicações Web (Backend): Muito usado em servidores para sistemas empresariais, APIs e microsserviços, com frameworks como Spring.
- Sistemas Empresariais: Software de grande escala, como sistemas bancários e de gestão, devido à sua segurança e escalabilidade.
- Big Data e Cloud: Soluções para análise de grandes volumes de dados e serviços baseados na nuvem.
- Internet das Coisas (IoT): Aplicações para dispositivos inteligentes e conectados.
- Aplicações Desktop: Softwares para computadores, embora menos comum para jogos de alta performance que C++.

Contexto

Algumas atividades só são eficientes quando temos a visualização de informações;

Usuários leigos em computação precisam ter uma forma “mais amigável” para utilizar programas.

Graphical user interface (**GUI**)



Product	Qtr 1	Qtr 2	Grand Total
Chocolade	\$ 744.60	\$ 162.56	\$ 907.16
Gummibärchen	\$ 5,079.60	\$ 1,249.20	\$ 6,328.80
Scottish Longbreads	\$ 1,267.50	\$ 1,062.50	\$ 2,330.00
Sir Rodney's Scones	\$ 1,418.00	\$ 756.00	\$ 2,174.00
Tarte au sucre	\$ 4,728.00	\$ 4,547.92	\$ 9,275.92
Chocolate Biscuits	\$ 943.89	\$ 349.60	\$ 1,293.49
Total	\$14,181.59	\$8,127.78	\$ 22,309.37

X

Command line interface (**CLI**)

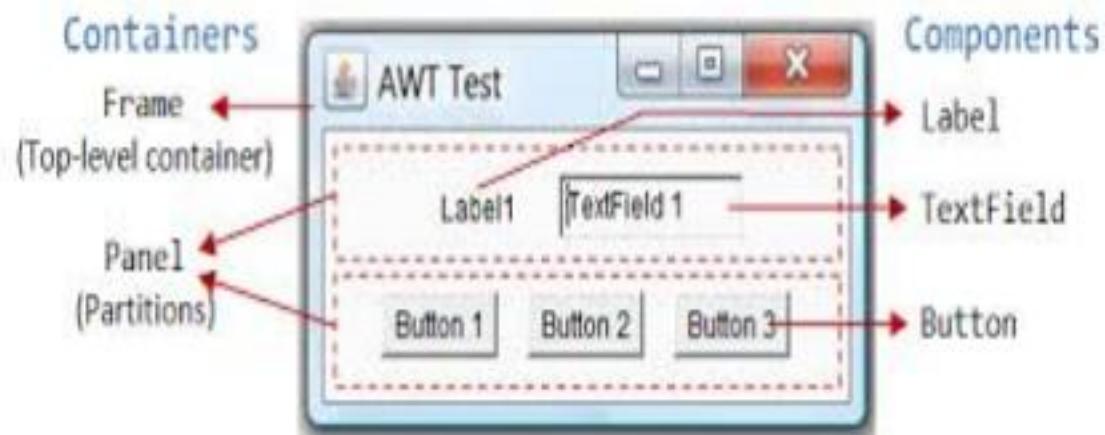
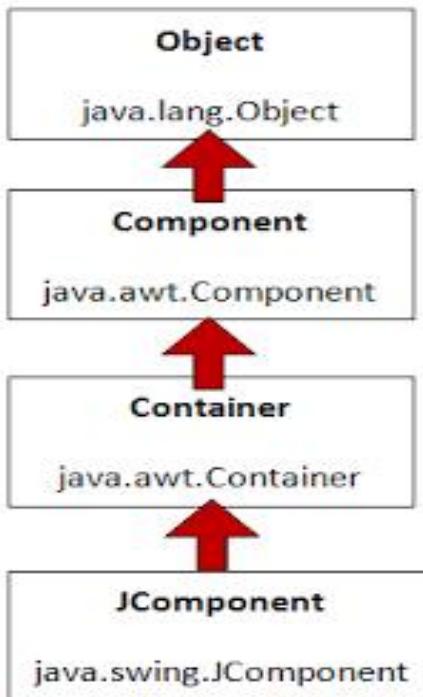


```
mysql> select * from student_db;
+----+-----+-----+-----+-----+-----+
| id | name | age | sex | class | grade |
+----+-----+-----+-----+-----+-----+
| 1  | anit | 17  | M   | 8B   | A+  |
| 2  | sneha| 28  | F   | 11A  | A++ |
+----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)

mysql>
```

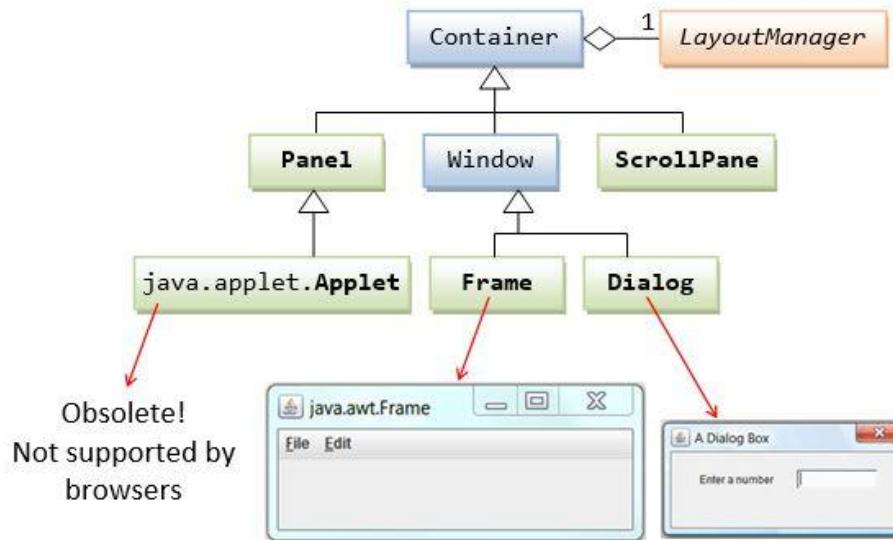
Interface Gráfica com Usuário (Graphical User Interface)

Essa interface é formada através de componentes GUI, conhecidos por controles ou widgets



Interface Gráfica com Usuário (Graphical User Interface)

Esses componentes são objetos que fazem a interação com usuário por teclado, mouse ou outros dispositivos que venham a servir para entrada de dados.



O que é Swing?

- Atualmente, o Java suporta, oficialmente, dois tipos de bibliotecas gráficas: AWT e Swing. A AWT (Abstract Window Toolkit) foi a primeira API para interfaces gráficas a surgir no Java e foi, mais tarde, superada pelo Swing, que possui diversos benefícios em relação a seu antecessor.
- AWT e Swing são bibliotecas gráficas oficiais inclusas em qualquer JRE (Java Runtime Environment ou Ambiente de Tempo de Execução Java) ou JDK (Java Development Kit).
- Além destas, existem algumas outras bibliotecas de terceiros. Exemplo: SWT - desenvolvida pela IBM e utilizada no Eclipse e em vários outros produtos.

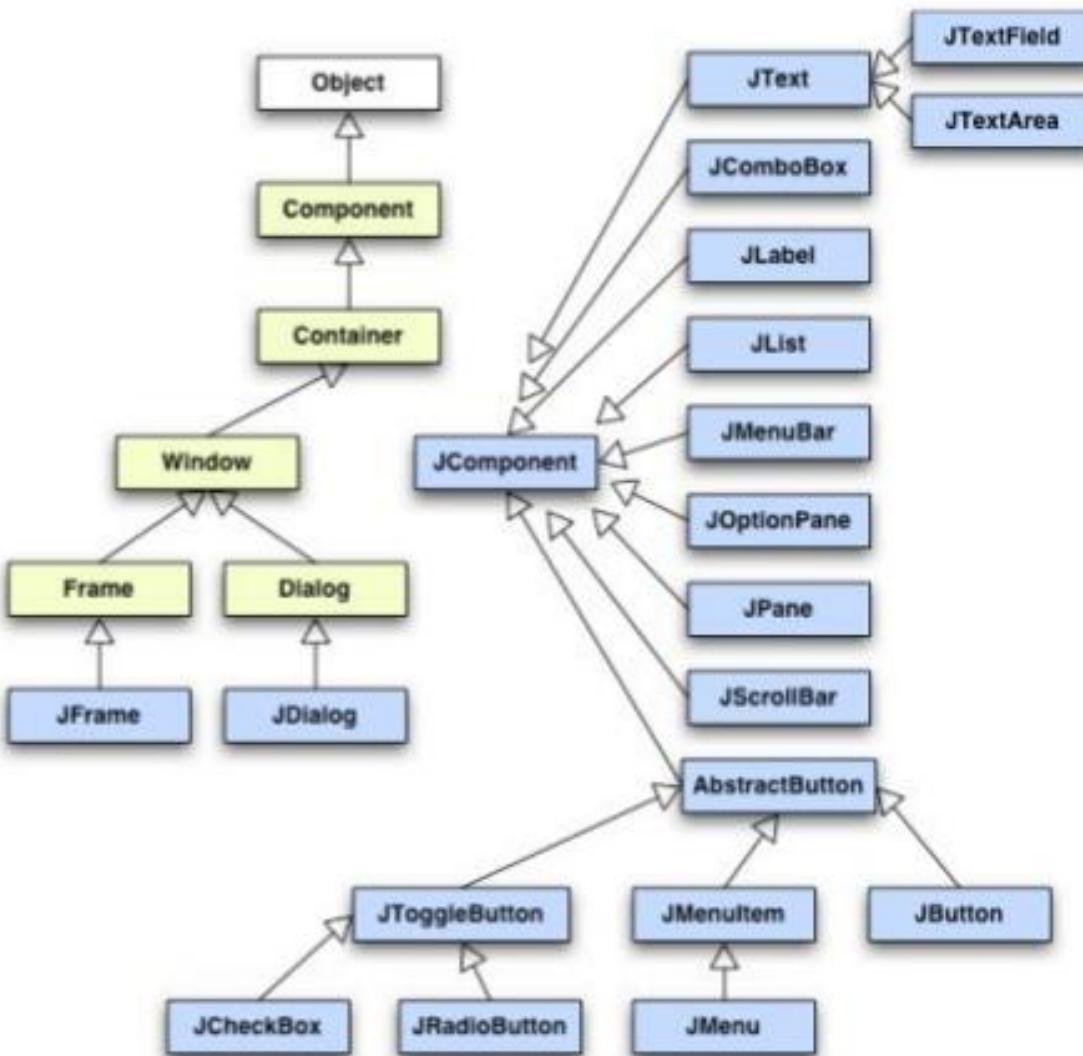


Figura 1: Diagrama resumido das classes AWT (amarelo) e Swing (azul).

O que é Swing?

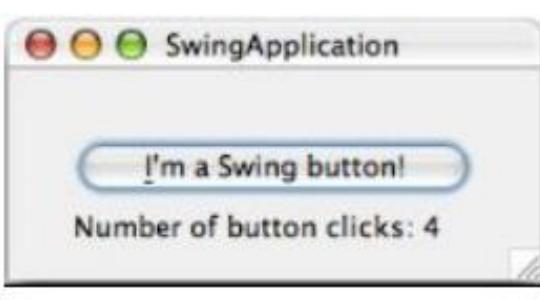
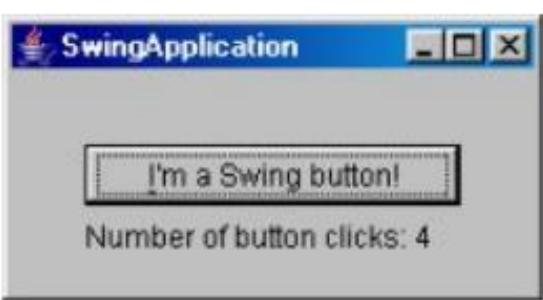
- Por padrão, o Java vem com um look-and-feel (LaF) próprio, que se comporta exatamente da mesma forma em todas as plataformas suportadas.

O que é LaF do java Swing?

- É um recurso da biblioteca Swing que permite modificar a aparência visual e o comportamento dos componentes de uma interface gráfica (GUI).
- Faz com que a aplicação se pareça e se comporte como o sistema operacional (Windows, Linux, macOS)
- Mas é possível alterar qual o look-and-feel!
- E, ainda pode baixar diversos LaF na Internet ou até desenvolver o seu próprio.

O que é LaF do java Swing?

- Screenshots da documentação do Swing mostrando a mesma aplicação rodando com 4 LaF diferentes:

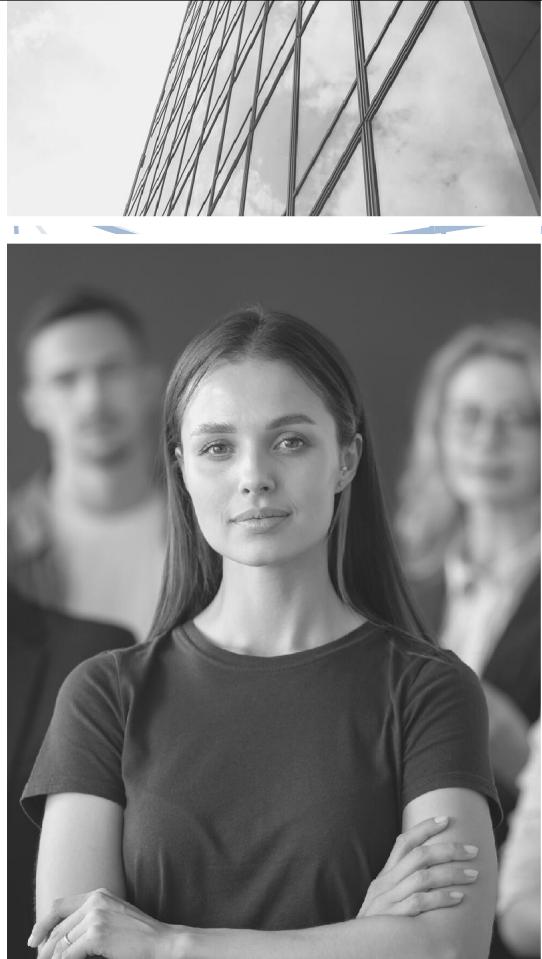


Exercício

- Escolha um LAF do java Swing para utilizar no seu projeto pessoal

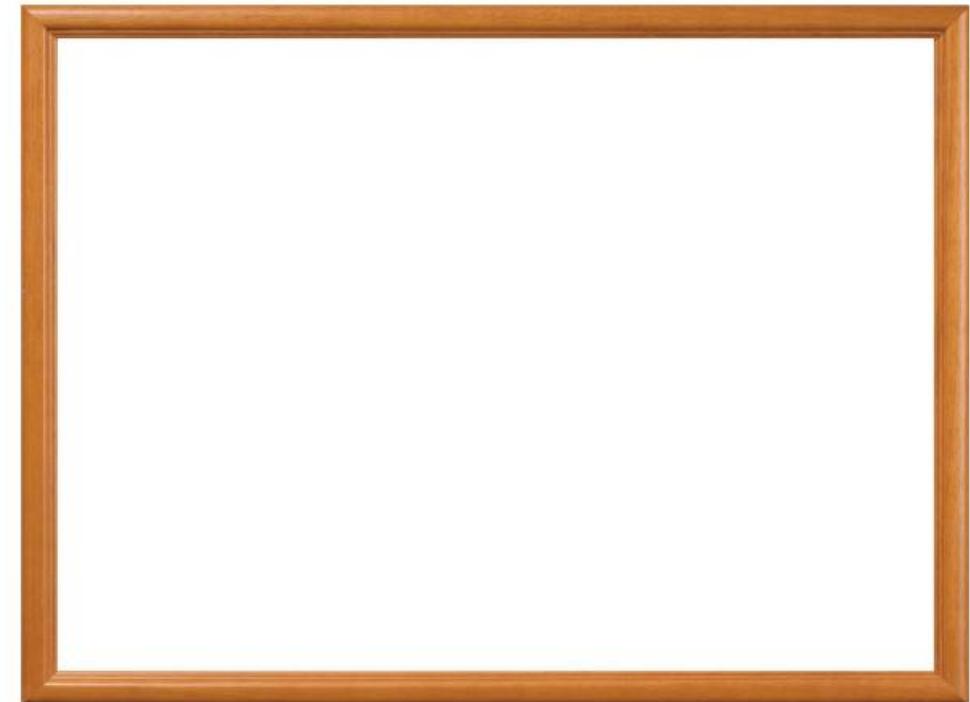
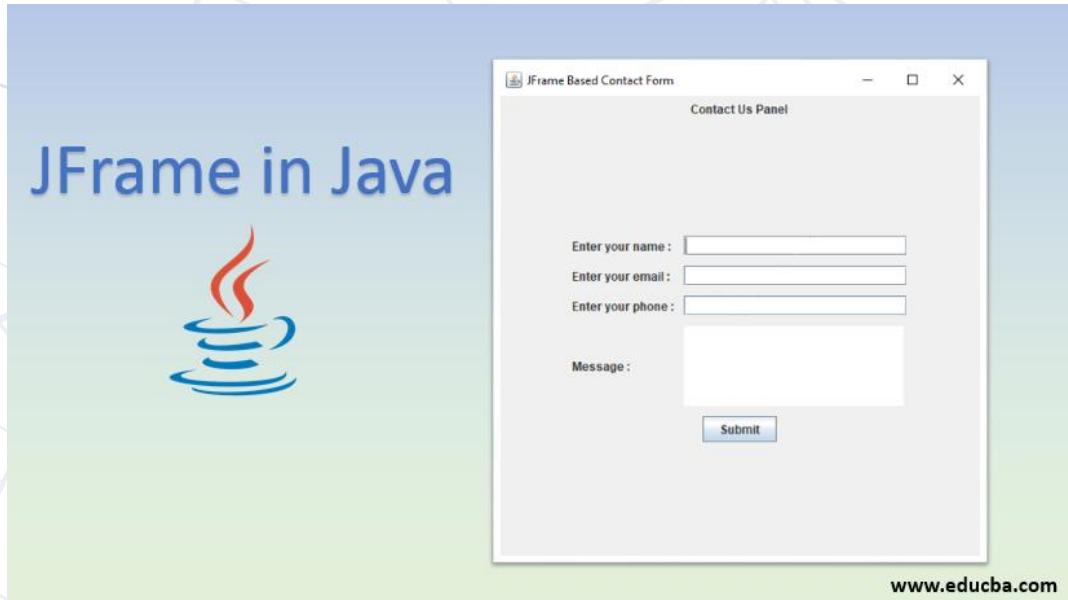
Criando uma interface gráfica

Rede
Fecomércio RS
de Educação



JFrame

O container de janela principal de uma aplicação Swing.



JFrame

- Tem barra de título, botão fechar, minimizar e maximizar
- É a janela principal do programa
- Possui um content pane onde você adiciona componentes

Exemplo de uso: abrir a tela principal de um sistema (login, dashboard etc.)

JDialog

Uma janela secundária, normalmente usada para:

- avisos
- configurações
- formulários menores

É semelhante ao JFrame, mas:

- nasce dependente de outra janela
- não aparece na barra de tarefas
- usado como janela auxiliar



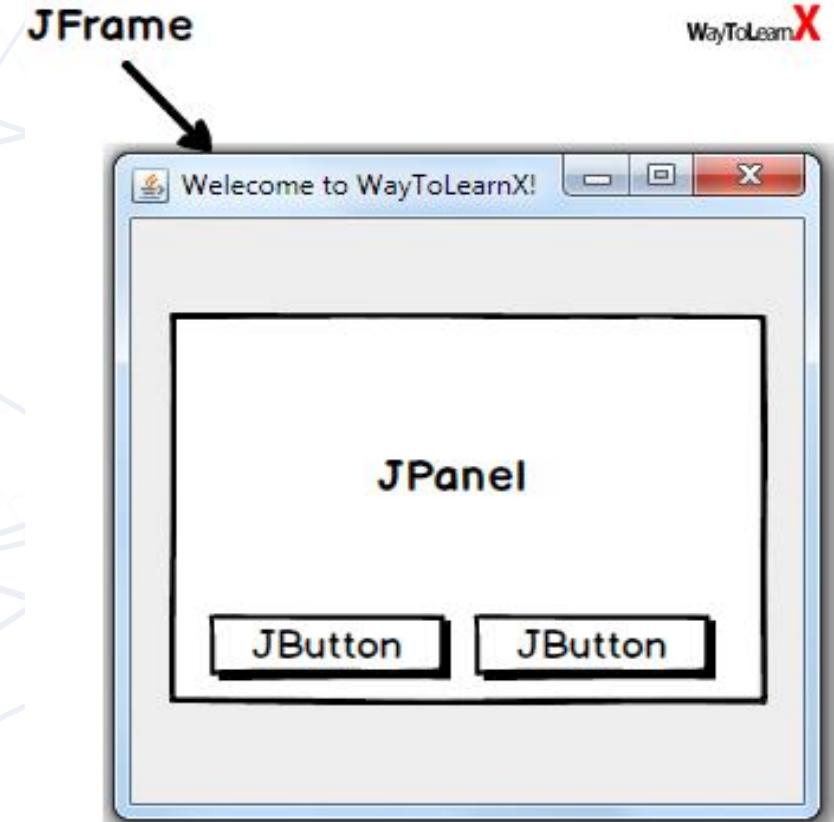
JPanel

O container mais básico e mais usado no Swing.

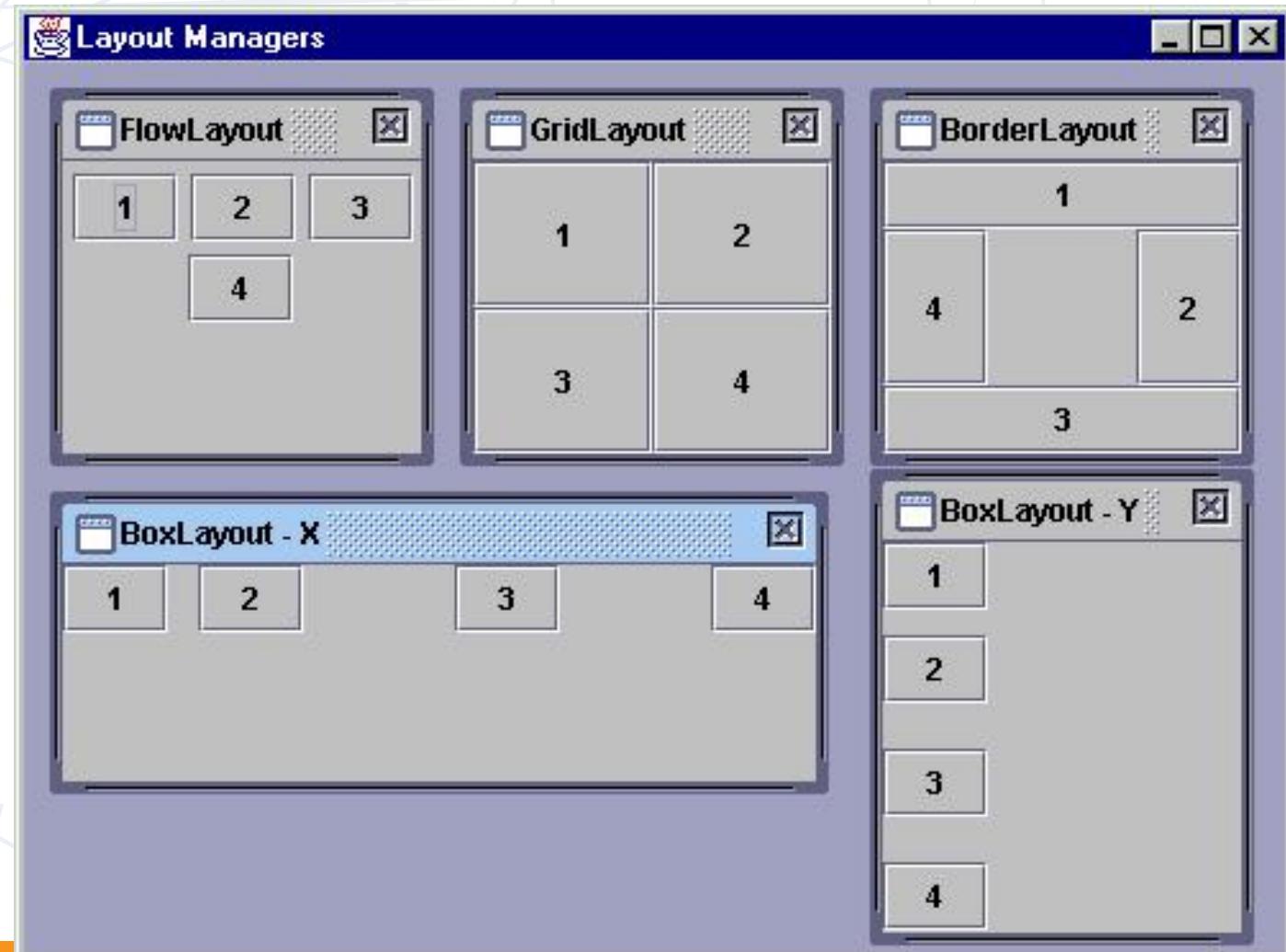
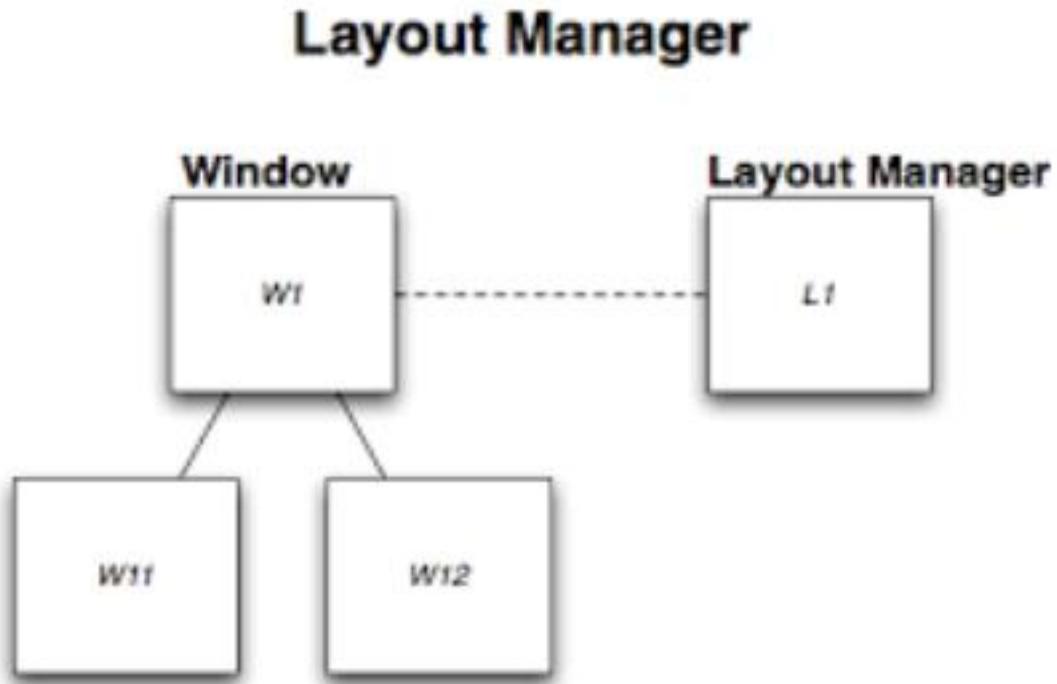
- Serve para organizar outros componentes
- Pode ser colocado dentro de outros containers
- Aceita qualquer LayoutManager (FlowLayout, GridLayout, BorderLayout etc.)

Exemplo:

Criar uma seção da tela (cabeçalho, parte lateral, rodapé, painel de botões etc.)



LayoutManager



JScrollPane

Um container que adiciona barras de rolagem a outro componente quando necessário.

Usos comuns:

- JList muito grande
- JTextArea com texto longo
- JTable com muitas linhas

Você coloca um componente dentro dele:

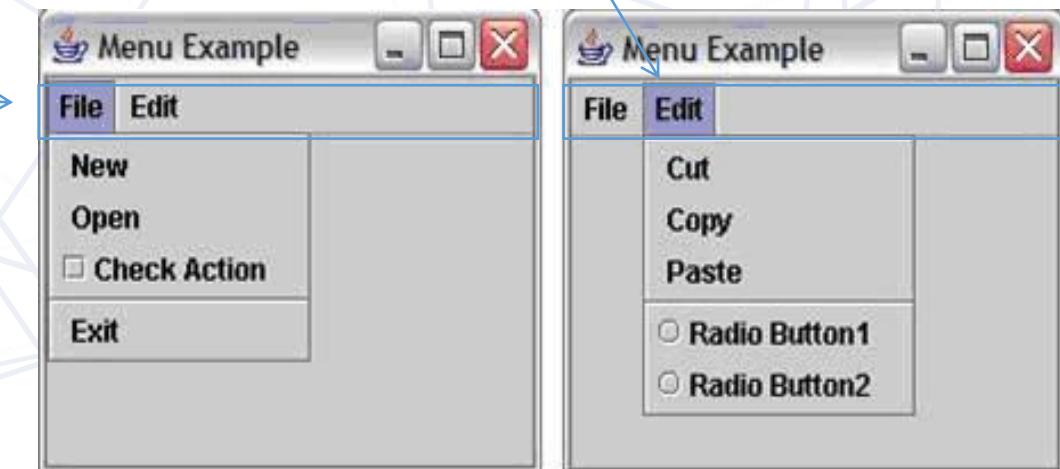
```
JScrollPane scroll = new JScrollPane(meuComponente);
```



JMenuBar

É a barra de menus colocada no topo da janela.

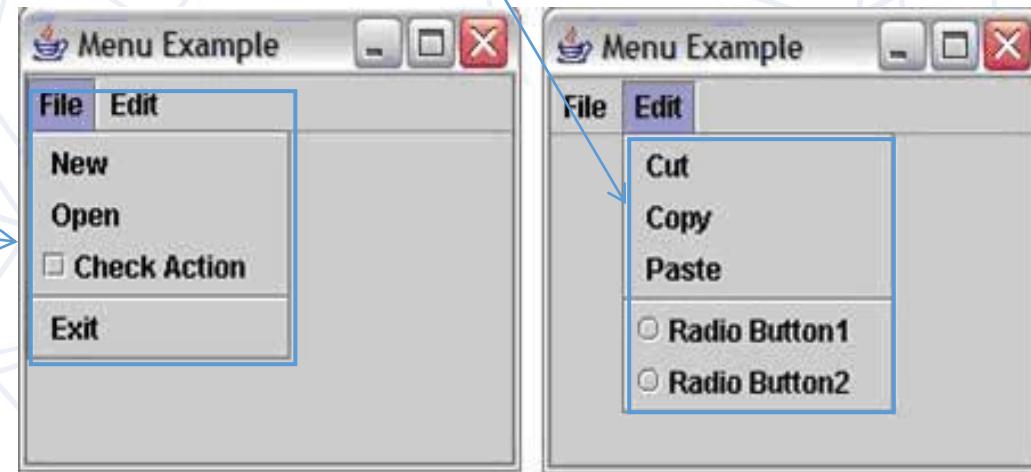
Não exibe conteúdo sozinho — ele contém os menus (JMenu).



JMenu

Um menu aberto dentro da JMenuBar.

É um container porque possui JMenuItem's dentro dele.



JMenuItem

É um item dentro de um menu no Swing.

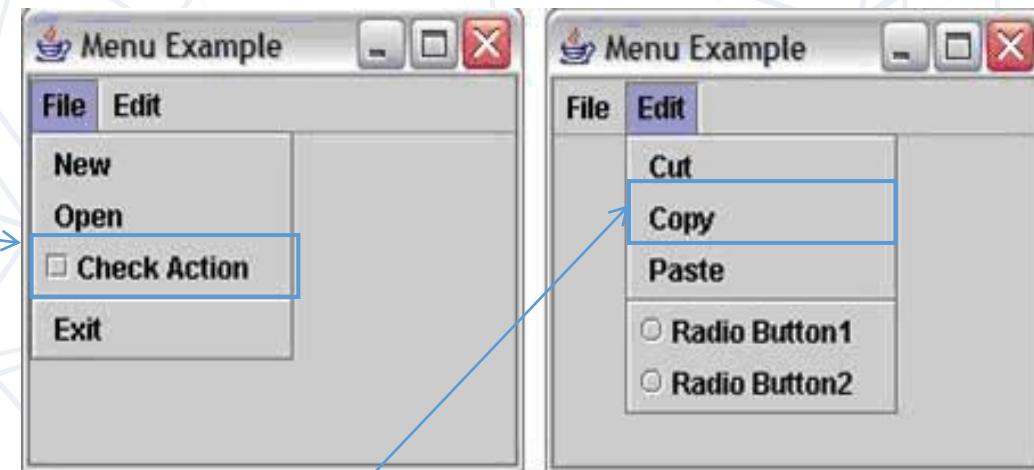
Ele representa uma opção que o usuário pode clicar dentro de um JMenu.

Ele é o equivalente a:

- "New"
- "Open"
- "Check Action"
- "Paste"
- "Exit"

Dispara ações quando selecionado

- *Pode ter atalhos (Ctrl+S, Ctrl+O etc.)*
- *Pode ter ícone*
- *Pode ser desativado (disabled)*
- *Pode pertencer a um submenu*

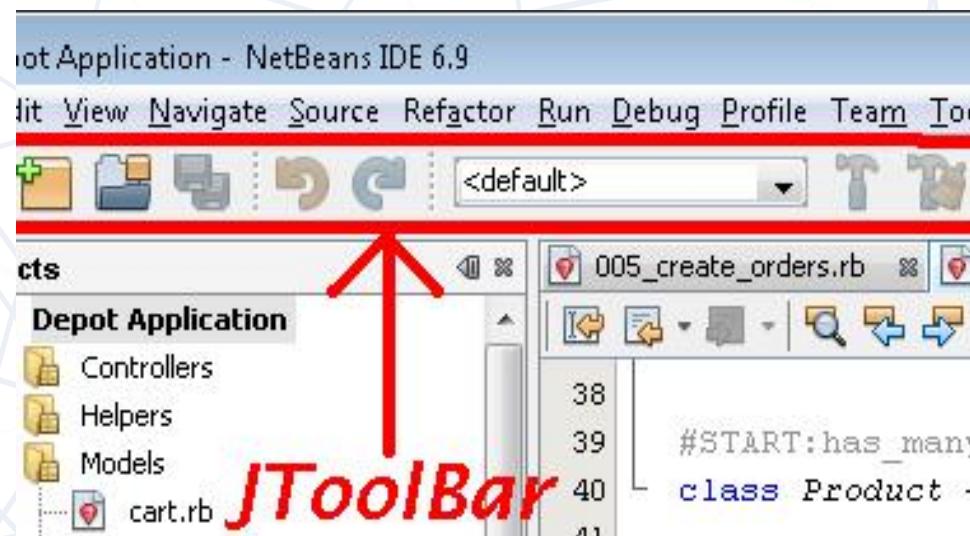


JToolBar

Uma barra de ferramentas, geralmente com:

- ícones
- botões
- atalhos de funções

Pode ser movida na tela.



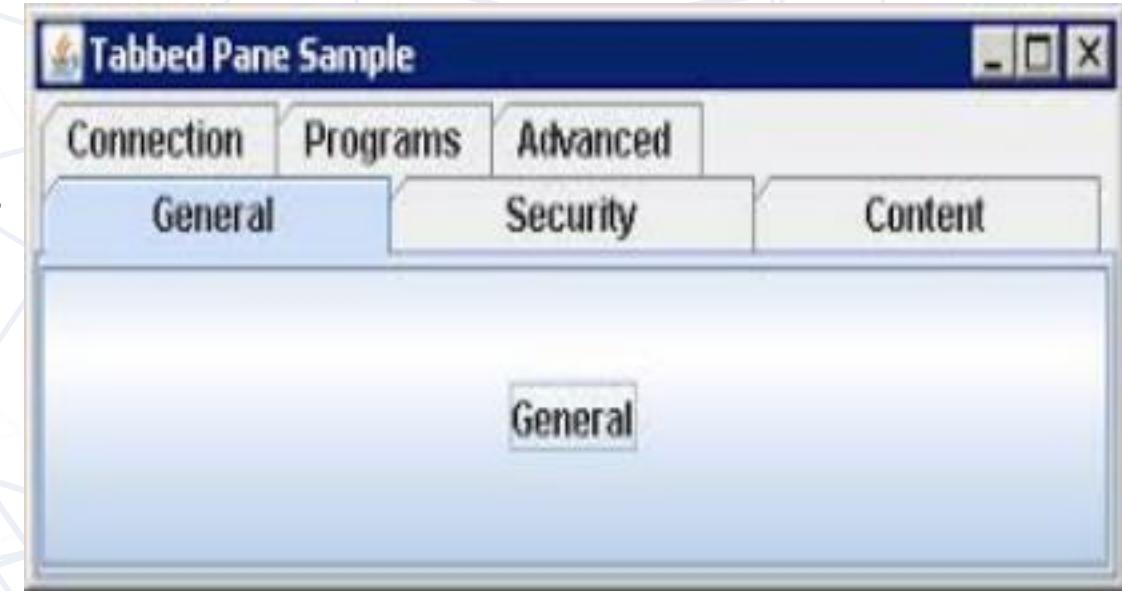
JTabbedPane

Um container que cria abas, como:

| Geral | Configurações | Ajuda |

Cada aba é um container que pode ter seus próprios componentes.

Muito usado para dividir interfaces complexas.
Pode ser movida na tela.



JSplitPane

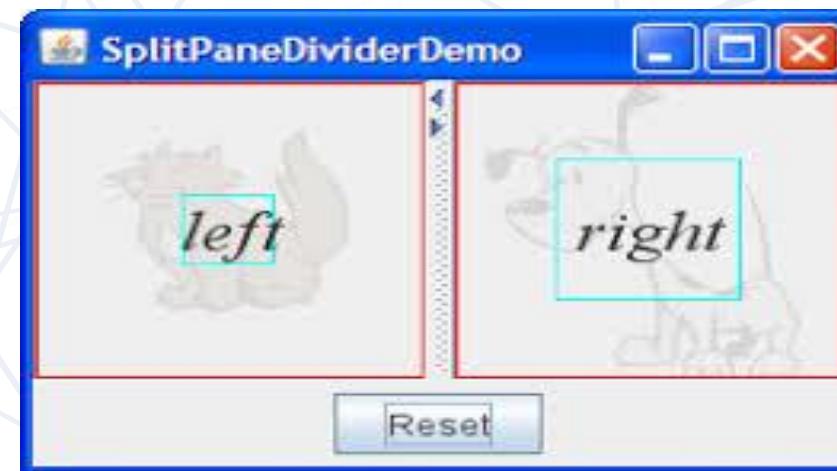
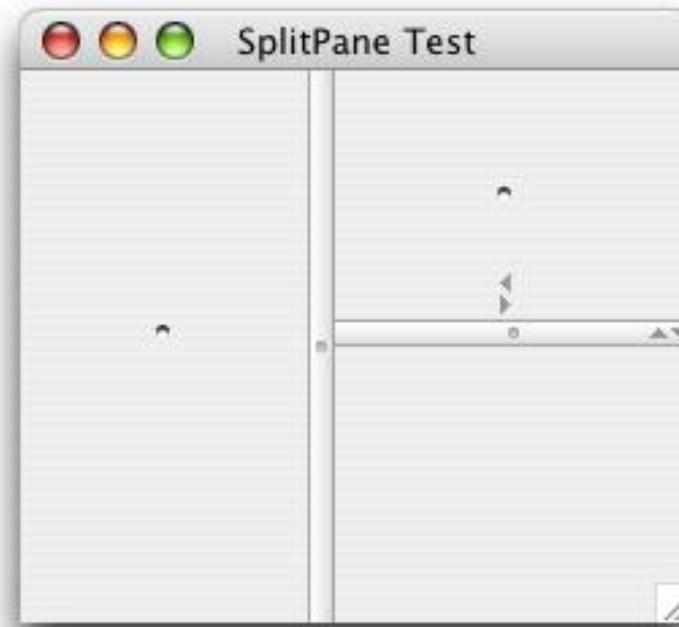
Divide a tela em duas partes:

← esquerda | direita →

ou

↑ superior | inferior ↓

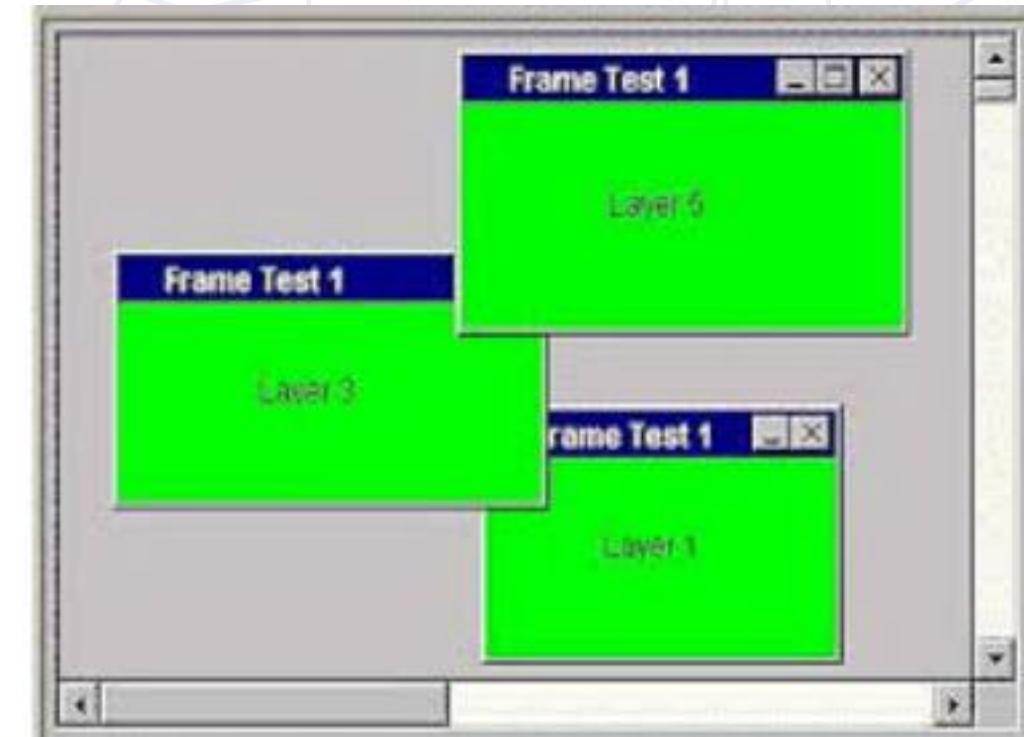
E permite arrastar a divisão.



JLayeredPane

Container que permite sobrepor componentes em camadas, como no Photoshop.

Usado quando você quer colocar algo por cima de outra coisa (pop-ups internos, imagens etc.)



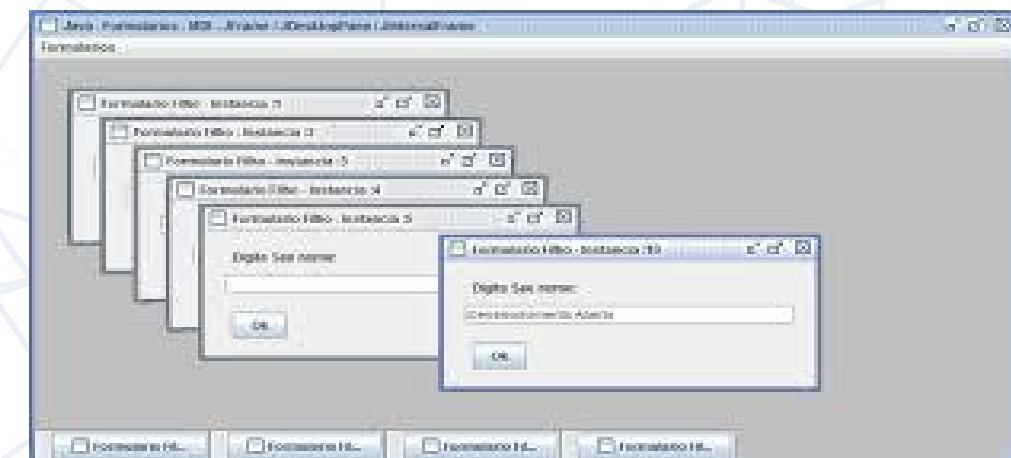
JDesktopPane

Usado para criar um ambiente de múltiplas janelas internas (MDI — Multiple Document Interface).

Usa JInternalFrame.

Exemplo:

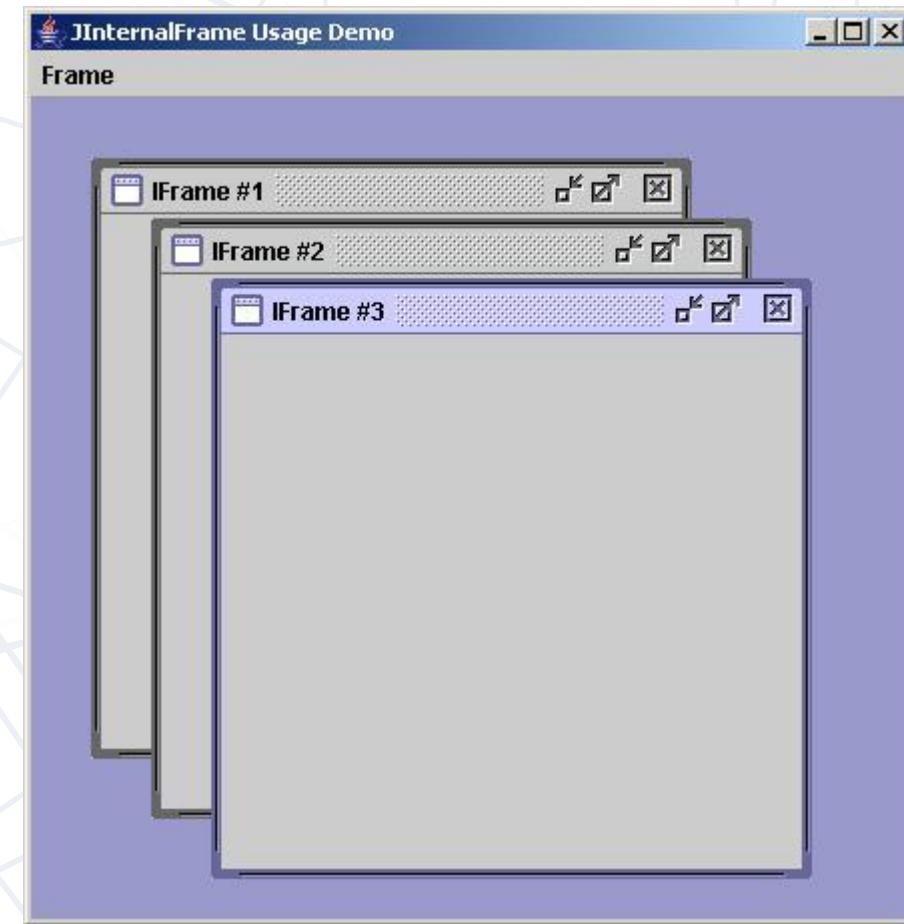
Programas como NetBeans, Photoshop,
ferramentas administrativas.



JInternalFrame

Uma “janela interna”, usada dentro de um JDesktopPane.

Parece um JFrame, mas fica dentro da aplicação.



Principais Propriedades JFrame

Rede
Fecomércio RS
de Educação

Principais Propriedades JFrame

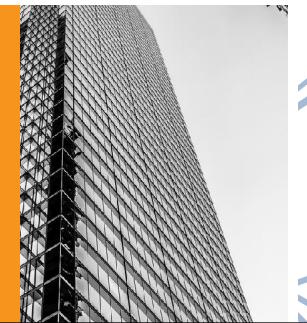
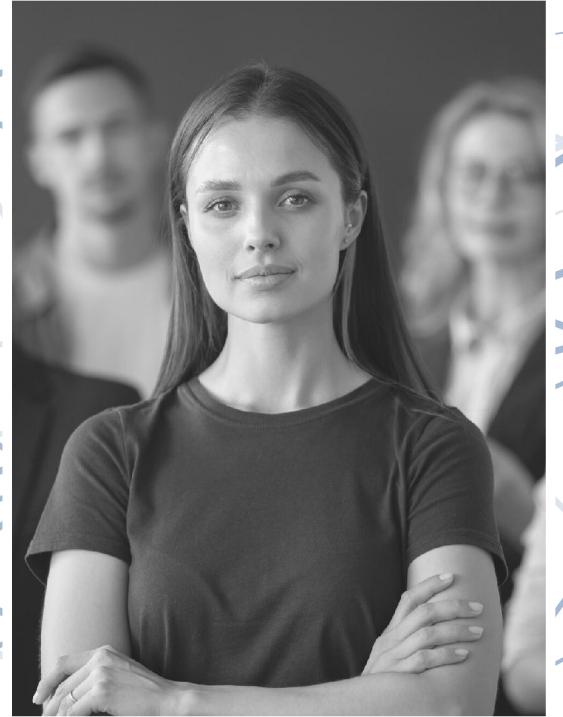
Método (set)	Para que usar	O que faz
setTitle(String)	Identificar a janela	Define o texto da barra de título
setSize(int w, int h)	Definir tamanho fixo	Ajusta largura e altura da janela
pack()	Ajustar automaticamente	Calcula o tamanho ideal conforme os componentes
setLocation(int x, int y)	Posicionar manualmente	Define a posição da janela na tela
setLocationRelativeTo(Component c)	Centralizar	Centraliza na tela ou em outra janela
setResizable(boolean)	Controlar redimensionamento	Permite ou bloqueia redimensionar
setDefaultCloseOperation(int)	Definir ação ao fechar	Controla o comportamento ao fechar
setVisible(boolean)	Mostrar janela	Exibe ou oculta o JFrame

Principais Propriedades JFrame

Método (set)	Para que usar	O que faz
setLayout(LayoutManager)	Organizar componentes	Define como os componentes são organizados
setIconImage(Image)	Identidade visual	Define o ícone da janela
getContentPane().setBackground(Color)	Aparência	Altera a cor de fundo
setEnabled(boolean)	Bloquear uso	Ativa ou desativa a janela
setFocusable(boolean)	Controle de foco	Define se recebe foco do teclado
setAlwaysOnTop(boolean)	Prioridade visual	Mantém a janela sempre na frente
setExtendedState(int)	Estado da janela	Minimiza, maximiza ou restaura
setMinimumSize(Dimension)	Límite mínimo	Define o menor tamanho permitido
setMaximumSize(Dimension)	Límite máximo	Define o maior tamanho permitido
setUndecorated(boolean)	Visual customizado	Remove barra de título e bordas

ATIVIDADE PRÁTICA GUIADA

Rede
Fecomércio RS
de Educação



Atividade Prática Guiada

Criando uma tela de cadastro de um Sistema Comercial

Objetivo

Criar uma tela de cadastro de cliente usando Java Swing, aplicando:

- JFrame
- Componentes Swing
- Propriedades (set)
- Eventos
- Validação de campos

Atividade Prática Guiada

PASSO 1 — Criar o projeto

- Abra o NetBeans
- Clique em File → New Project
- Escolha Java with Ant / Maven → Java Application
- Nome do projeto: SistemaComercial

Atividade Prática Guiada

PASSO 2 — Criar a tela principal (JFrame)

- Clique com o botão direito no projeto
- New → JFrame Form
- Nome da classe: TelaCadastroCliente

Atividade Prática Guiada

PASSO 3 — Configurar propriedades do JFrame

No modo Design, clique fora da tela (no JFrame) e configure o comportamento da janela

- Define o nome que aparece na barra da janela `setTitle("texto")`
- Define largura e altura da janela `setSize(largura, altura)`
- Faz a janela abrir centralizada na tela `setLocationRelativeTo(null);`
- Impede que o usuário redimensione a janela `setResizable(false);`

Atividade Prática Guiada

PASSO 4 — Adicionar os componentes

- No modo Design, arraste da Paleta:

Labels (JLabel) e os campos (JTextField)

Nome:	txtNome
Email:	txtEmail
Telefone:	txtTelefone
CPF:	txtCpf
Endereço:	txtEndereco

O JLabel identifica, o JTextField recebe o dado.

Atividade Prática Guiada

PASSO 5 — Adicionar os botões

- Adicione 3 JButton:

Nome do Botão	Texto
btnSalvar	Salvar
btnLimpar	Limpar
btnSair	Sair

Atividade Prática Guiada

PASSO 6 — Organizar o layout

Organizar os componentes da tela de cadastro em forma de tabela, com

GridLayout:

- Labels à esquerda
- Campos de texto à direita

FlowLayout

- Botões alinhados na parte inferior

Atividade Prática Guiada

PASSO 7 — Criar o evento do botão Salvar

- Clique no botão Salvar
- Vá em Events → actionPerformed
- O NetBeans criará o método

Porque a validação?

Antes de salvar, precisa verificar se o usuário preencheu tudo.

Adicione a validação:

```
private void btnSalvarActionPerformed(java.awt.event.ActionEvent ev  
    if (txtNome.getText().trim().isEmpty() ||  
        txtEmail.getText().trim().isEmpty() ||  
        txtTelefone.getText().trim().isEmpty() ||  
        txtCpf.getText().trim().isEmpty() ||  
        txtEndereco.getText().trim().isEmpty()) {  
  
    JOptionPane.showMessageDialog(this,  
        "Preencha todos os campos!",  
        "Erro",  
        JOptionPane.ERROR_MESSAGE);  
    return;  
}  
  
JOptionPane.showMessageDialog(this,  
    "Cliente cadastrado com sucesso!",  
    "Sucesso",  
    JOptionPane.INFORMATION_MESSAGE);  
}
```

Atividade Prática Guiada

PASSO 8 — Evento do botão Limpar

- Siga os mesmos passos do evento anterior
- Adicione o código do evento do botão limpar:

```
private void  
btnLimparActionPerformed(java.awt.event.ActionEvent evt) {  
  
    txtNome.setText("");  
    txtEmail.setText("");  
    txtTelefone.setText("");  
    txtCpf.setText("");  
    txtEndereco.setText("");  
    txtNome.requestFocus();  
}  
}
```

Atividade Prática Guiada

PASSO 9 — Evento do botão Sair

- Siga os mesmos passos do evento anterior
- Adicione o código do evento do botão sair:

```
private void  
btnSairActionPerformed(java.awt.event.ActionEvent evt) {  
    System.exit(0);  
}
```

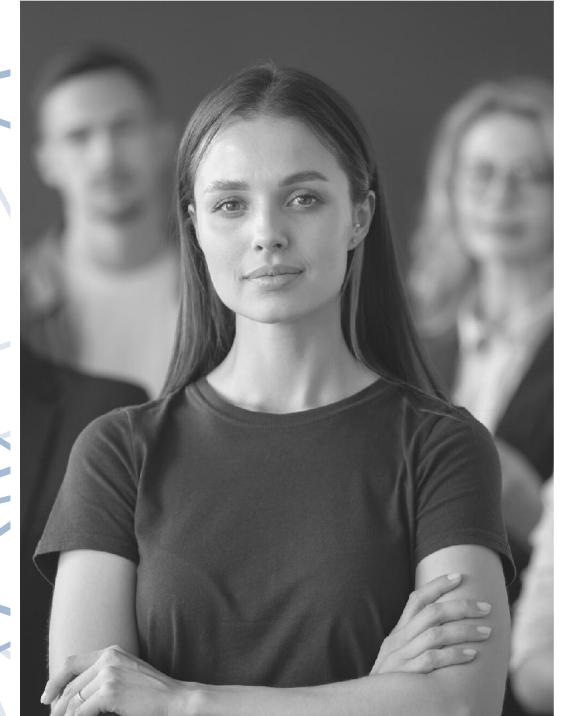
Atividade Prática Guiada

PASSO 10 — Executar o sistema

1. Clique com o botão direito na classe
2. Run File
3. Teste:
 - Campos vazios
 - Limpar
 - Sair

ATIVIDADE PRÁTICA

Rede
Fecomércio RS
de Educação



Atividade Prática

Criando o Menu Principal de um Sistema Comercial

Objetivo:

Deve criar uma interface inicial de um software comercial em Java Swing, contendo uma janela principal com barra de menus funcional.

Atividade Prática

Criando o Menu Principal de um Sistema Comercial

Tarefas da Atividade

1. Criar a janela principal (JFrame)
 - Título: Sistema Comercial – Aluno
 - Tamanho sugerido: 800 × 600
 - Abrir centralizada
 - Não precisa deixar bonita — foco é aprender a estrutura

Atividade Prática

Criando o Menu Principal de um Sistema Comercial

Tarefas da Atividade

2. Criar a barra de menus (JMenuBar) com 3 menus

Menu: Cadastro

Adicionar estes itens (JMenuItem):

- Clientes
- Produtos
- Funcionários

Quando clicar, mostrar uma mensagem tipo:

“Tela de cadastro de Clientes ainda será desenvolvida.”

Menu: Movimentações

Itens:

- Vendas
- Compras

Mensagem:

“A tela de Vendas será criada no projeto final.”

Menu: Relatórios

Itens:

- Estoque
- Financeiro

Mensagem:

“Relatório ainda não disponível.”

Atividade Prática

Criando o Menu Principal de um Sistema Comercial

Tarefas da Atividade

3. Criar mais um menu

Menu: Sistema

Itens:

- Configurações
- Sair

Ação:

Configurações → mostrar mensagem

Sair → fechar o sistema

Atividade Prática

Criando o Menu Principal de um Sistema Comercial

Tarefas da Atividade

4. Criar um painel central (JPanel)

Conteúdo:

Um texto grande: “Bem-vindo ao Sistema Comercial”

(Se quiserem, podem personalizar com cor de fundo, fonte etc.)

Atividade Prática

Criando o Menu Principal de um Sistema Comercial

ENTREGA DA ATIVIDADE

UTILIZANDO GITHUB CRIE UM REPOSITÓRIO
COM NOME:

Disciplina-Desenvolver-e-organizar-interface-
grafica-para-aplicacoes-desktop

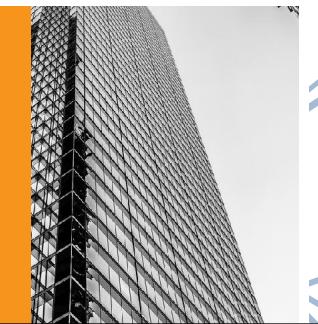
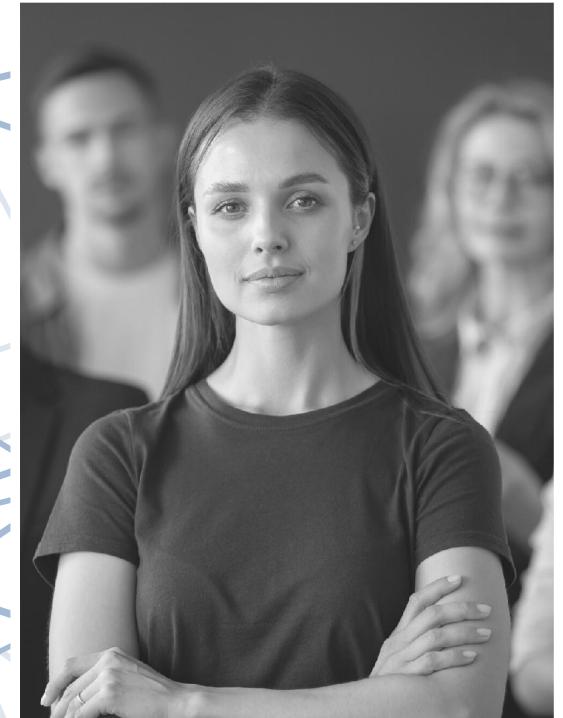
OBS: MANDAR UM EMAIL PARA

professor.jeanmax@gmail.com

com o link do repositório público

Definição do Projeto

Rede
Fecomércio RS
de Educação



Estrutura de pastas

```
SistemaLoja/
|   +-- src/
|   |   +-- MainFrame.java
|   |   +-- Controller.java
|   +-- models/
|       +-- Cliente.java
|       +-- Produto.java
|       +-- Venda.java
|   +-- views/
|       +-- ClienteFrame.java
|       +-- ProdutoFrame.java
|       +-- VendaFrame.java
+-- README.txt
```