

Minicurso Java

Java + Window Builder + JDBC

Prof. MSc. Vinícius Camargo Andrade

vcandrade@utfpr.edu.br

Departamento Acadêmico de Informática
Universidade Tecnológica Federal do Paraná



Vinícius Camargo Andrade

Professor do Magistério Superior



vcandrade@utfpr.edu.br



/vcandrade



/Vinicius_Camargo_Andrade



/prof-vcandrade

Materiais

Para obter o código-fonte:

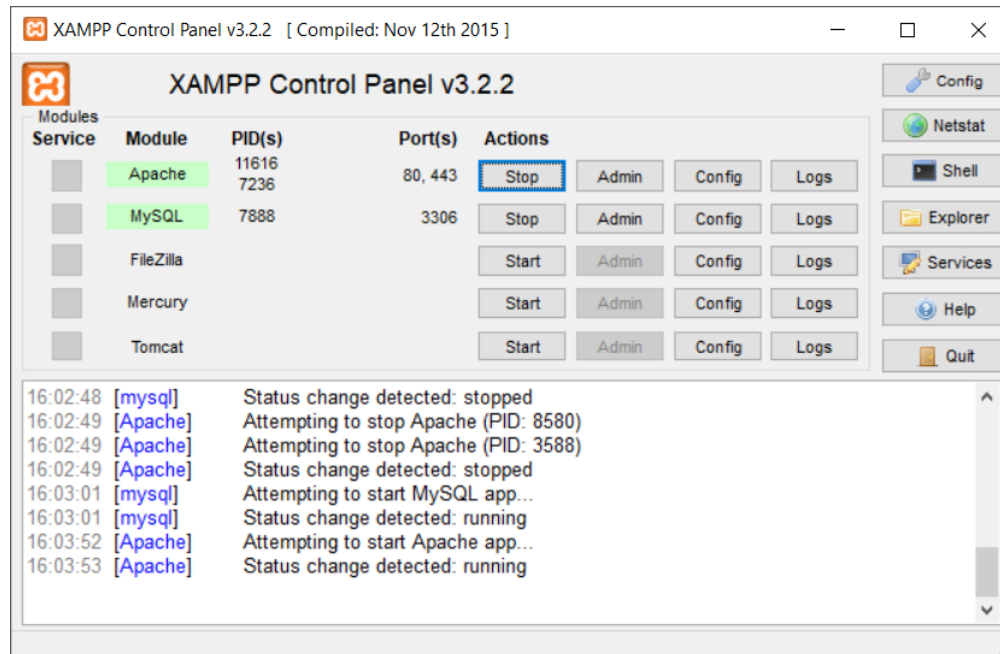
<https://github.com/vcandrade/MinicursoJavaJDBC>

Iniciar Servidor MySQL

Iniciar Servidor MySQL

Antes de conectar a um Servidor de Banco de Dados MySQL, certifique-se de que ele esteja em execução.

Iniciar Servidor MySQL

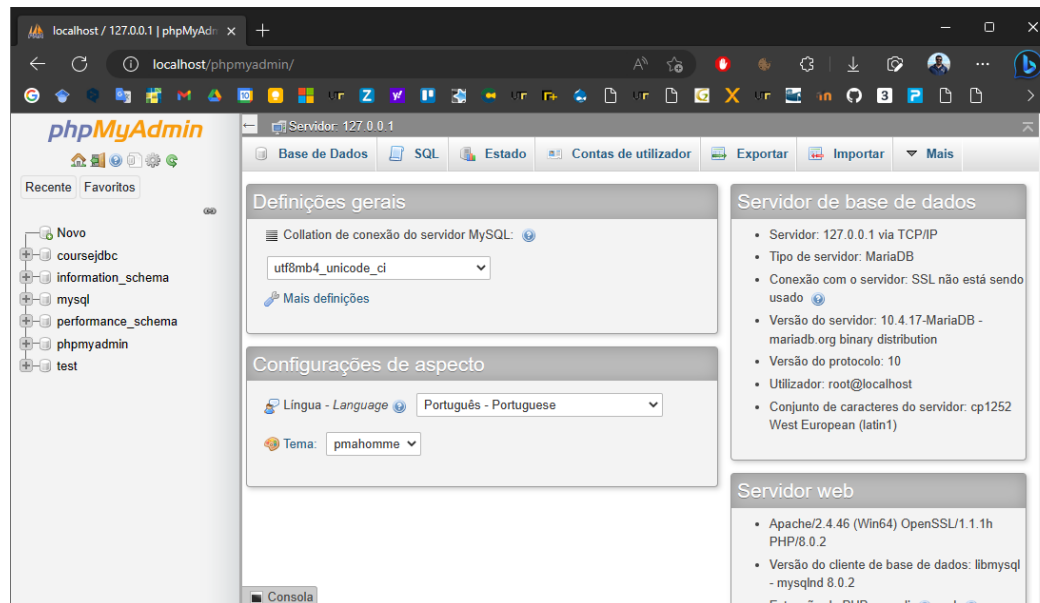


Acessar a Base de Dados

Acessar a Base de Dados

Para acessar a base, acesse o link:

<http://localhost/phpmyadmin/>



Criar Database

Criar Database

Crie um novo database:

- *Nome: minicursojava*
- *Codificação: utf8_general_ci*

Criar Database

Servidor: 127.0.0.1 » Base de Dados: minicursojava » Tabela: Aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Privilégios Operações Rastreamento Acionadores

Nome da Tabela: Aluno Adicionar 1 coluna(s) Executar

Nome	Tipo	Tamanho/Valores*	Predefinido	Agrupamento (Collation)	Atributos	Nulo	Índice	Comentário
registro_academico	INT		Nenhum			<input checked="" type="checkbox"/>	PRIMARY	
Escolher a partir das Colunas Centrais								
nome	VARCHAR	50	Nenhum			<input type="checkbox"/>	---	
Escolher a partir das Colunas Centrais								
curso	VARCHAR	50	Nenhum			<input type="checkbox"/>	---	
Escolher a partir das Colunas Centrais								
sexo	VARCHAR	20	Nenhum			<input type="checkbox"/>	---	
Escolher a partir das Colunas Centrais								

Estrutura

Comentários da tabela: Colação: Motor de armazenamento: InnoDB

Definição da PARTIÇÃO:

Partição por: (Expressão ou lista de coluna)

Partições:

Pré-visualizar SQL Guarda

Criar Database

Servidor: 127.0.0.1 » Base de Dados: minicursojava » Tabela: aluno

Procurar Estrutura SQL Pesquisar Inserir Exportar Importar Privilégios Operações Rastrear

Estrutura da tabela Visão de relação(ões)

#	Nome	Tipo	Agrupamento (Collation)	Atributos	Nulo	Predefinido	Comentários	Extra	Acções
<input type="checkbox"/>	1 registro_academico	int(11)			Não	Nenhum			Muda Elimina Mais
<input type="checkbox"/>	2 nome	varchar(50)	utf8_general_ci		Não	Nenhum			Muda Elimina Mais
<input type="checkbox"/>	3 curso	varchar(50)	utf8_general_ci		Não	Nenhum			Muda Elimina Mais
<input type="checkbox"/>	4 sexo	varchar(20)	utf8_general_ci		Não	Nenhum			Muda Elimina Mais

↑ ☐ Marcar todos Com os seleccionados: Procurar Muda Elimina Primária Único Índice Texto Completo

Adicionar à(s) coluna(s) central(is) Remover da(s) coluna(s) central(is)

Imprimir Propor uma estrutura de tabela Acompanhar tabela Mover campo(s) Normalizar

Adicionar 1 campo(s) após sexo Executar

Criar uma Biblioteca de Usuário

MySQL Connector

Para mantermos uma organização das bibliotecas Java, crie uma pasta chamada “java-libs”, no diretório de sua preferência. Crie um subpasta denominada “jdbc-connectors”.

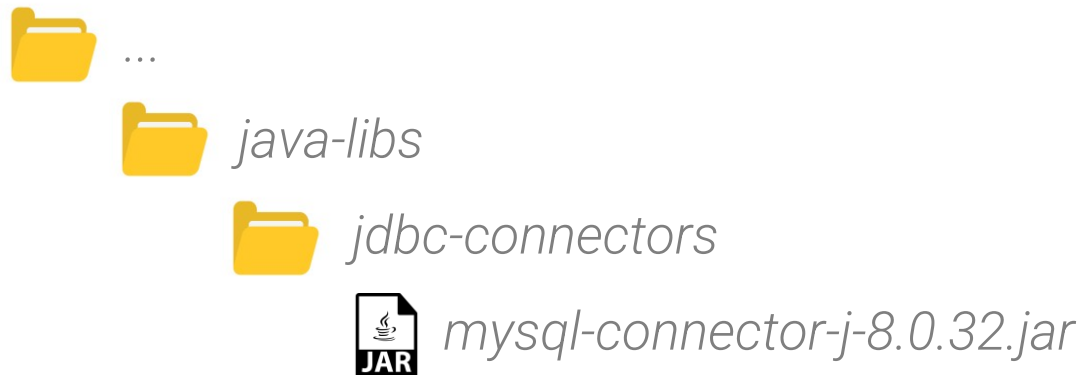
MySQL Connector

*Para que a aplicação se comunique com um banco de dados, é necessário utilizar o conector correto. Para conectar **sistemas Java** com o **MySQL**, o conector correto pode ser obtidos nos links:*

- *<https://dev.mysql.com/downloads/connector/j/>*
- *<https://moodle.utfpr.edu.br/mod/resource/view.php?id=1421507>*

MySQL Connector

Armazene o arquivo *“mysql-connector-j-8.0.32.jar”* na pasta *“jdbc-connectors”*.



Cria uma Biblioteca de Usuário

*Para utilizar o conector MySQL em uma aplicação Java, é necessário **adicioná-lo ao projeto**. Uma maneira organizada de fazer esse processo é por meio de **biblioteca**.*

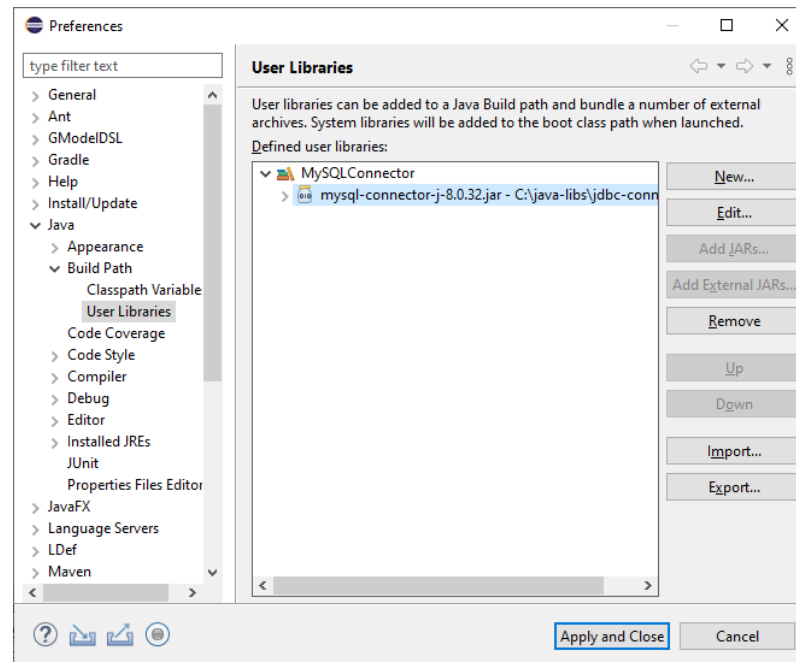
Cria uma Biblioteca de Usuário

Para criar uma nova biblioteca de usuário no Eclipse, acesse:

- *Window > Preferences > Java > Build Path > User Libraries > New;*
- *Nome da Biblioteca: MySQLConnector*

Cria uma Biblioteca de Usuário

Após criar a biblioteca, selecione-a e pressione o botão “Add External JARs...”, localizando e selecionando o arquivo “mysql-connector-j-8.0.32.jar” da pasta “jdbc-connectors”.



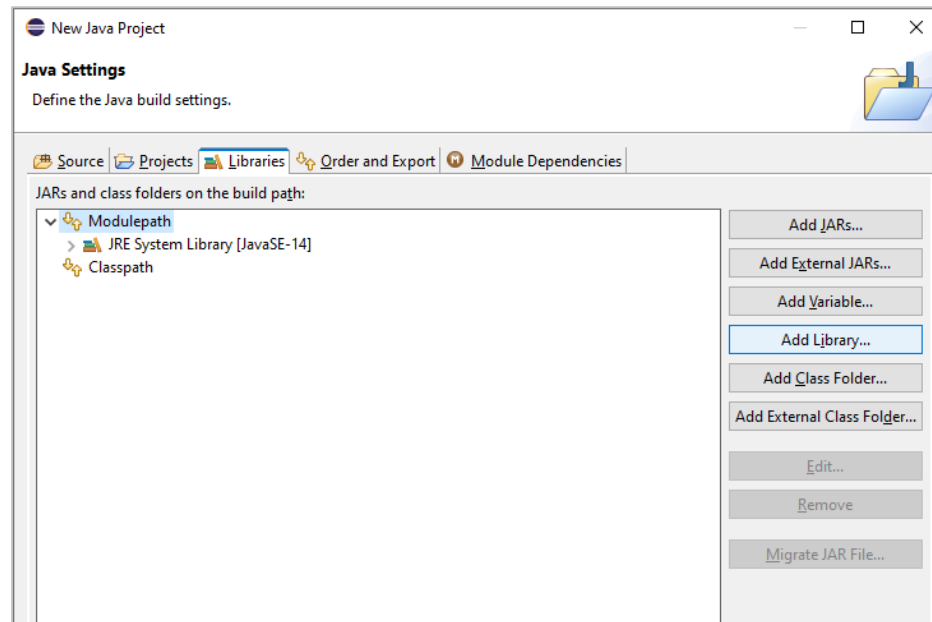
Criar Projeto Java

Criar Projeto Java

Crie um novo Projeto Java com o nome “MinicursoJava” e pressione em “Next”.

Criar Projeto Java

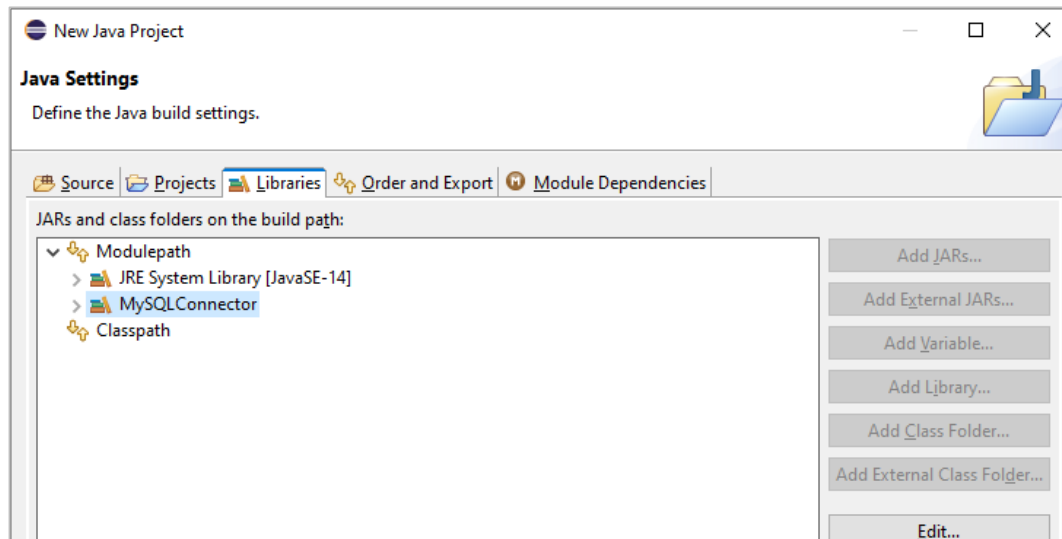
Na aba “*Libraries*”, selecione o “*Modulepath*” e pressione o botão “*Add Library*”.



Criar Projeto Java

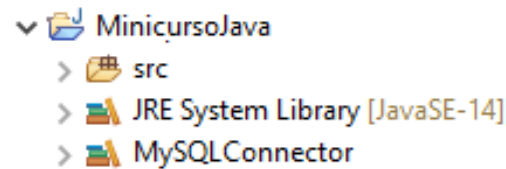
Na próxima janela, selecione:

- *User Library > MySQLConnector, pressione o botão “Finish”.*



Criar Projeto Java

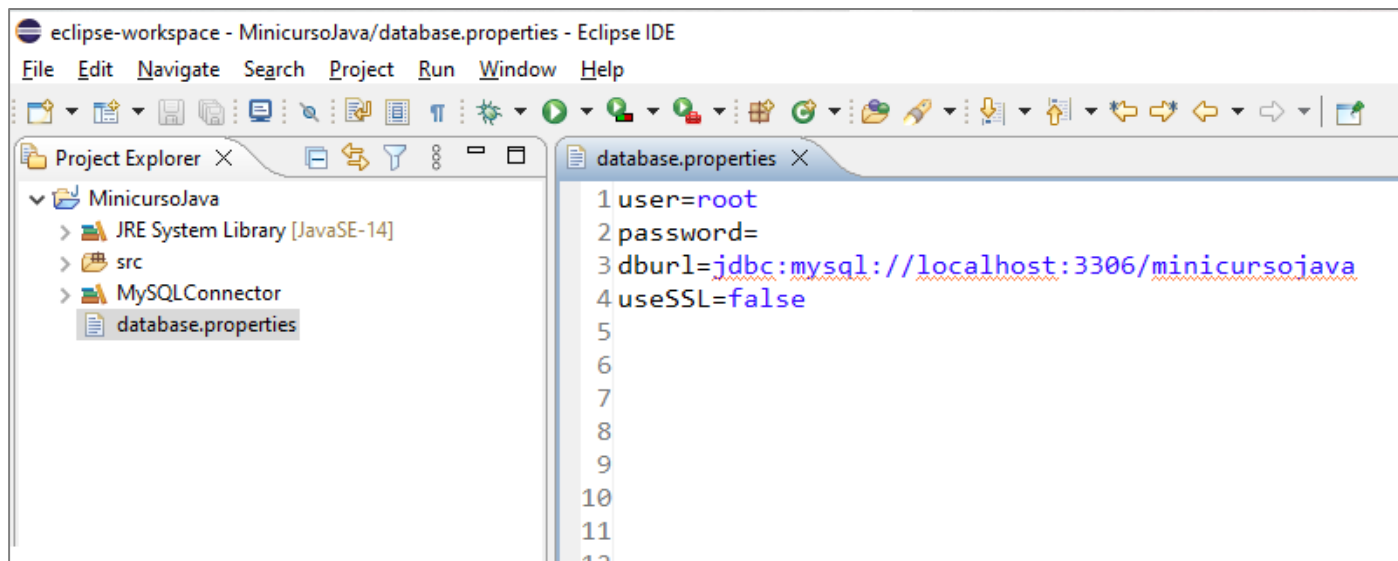
*O Projeto Java **MinicursoJava** deve estar devidamente criado com a seguinte estrutura:*



Criar Arquivo com as
Propriedades da Base de
Dados

Arquivos de Propriedades

Na pasta raiz do projeto, crie um novo arquivo, denominado “database.properties” com o seguinte conteúdo:



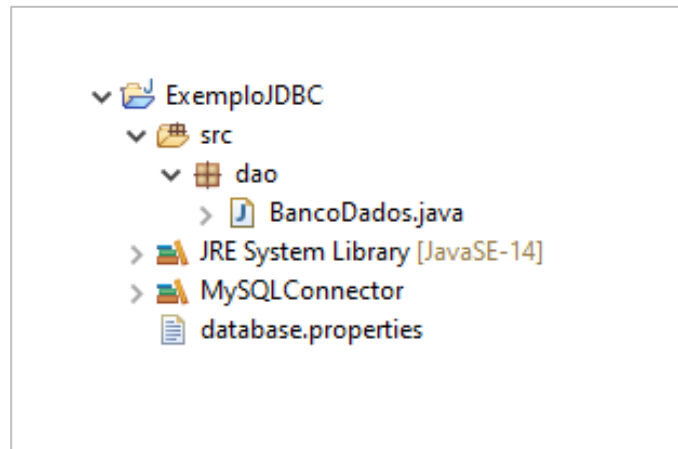
The screenshot shows the Eclipse IDE interface. The title bar reads "eclipse-workspace - MinicursoJava/database.properties - Eclipse IDE". The menu bar includes File, Edit, Navigate, Search, Project, Run, Window, and Help. The toolbar contains various icons for file operations and development tools. The Project Explorer on the left shows the project structure: MinicursoJava (expanded) containing JRE System Library [JavaSE-14], src, MySQLConnector, and database.properties (selected). The editor on the right shows the content of database.properties:

```
1 user=root
2 password=
3 dburl=jdbc:mysql://localhost:3306/minicursojava
4 useSSL=false
5
6
7
8
9
10
11
12
```

Classe de Conexão

Classe de Conexão

Primeiramente, deve-se criar o pacote “dao”, que conterá todas as classes que interagirão com a base de dados. Após a criação do pacote, deve-se criar uma nova classe, denominada “BancoDados”.

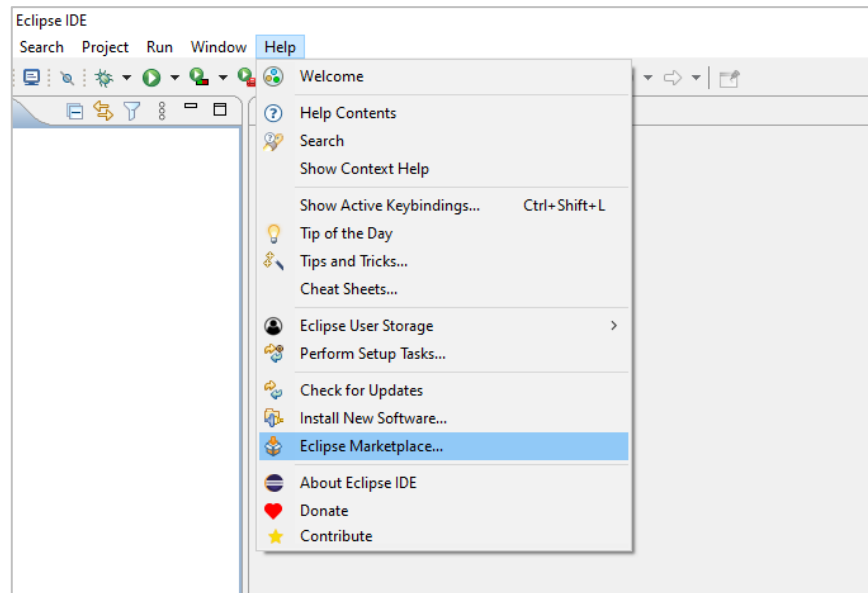


Instalação

Window Builder

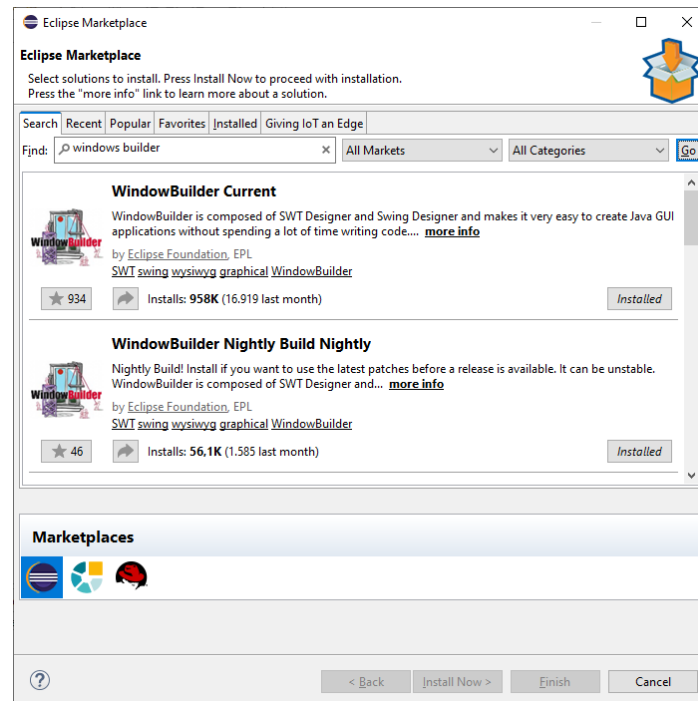
Instalação do Windows Builder

No Eclipse IDE, selecione as opções “Help > Eclipse Marketplace”.



Instalação do Windows Builder

No marketplace, pesquise por “Window Builder” e após encontrar, pressione o botão “Install”.



Instalação do Windows Builder

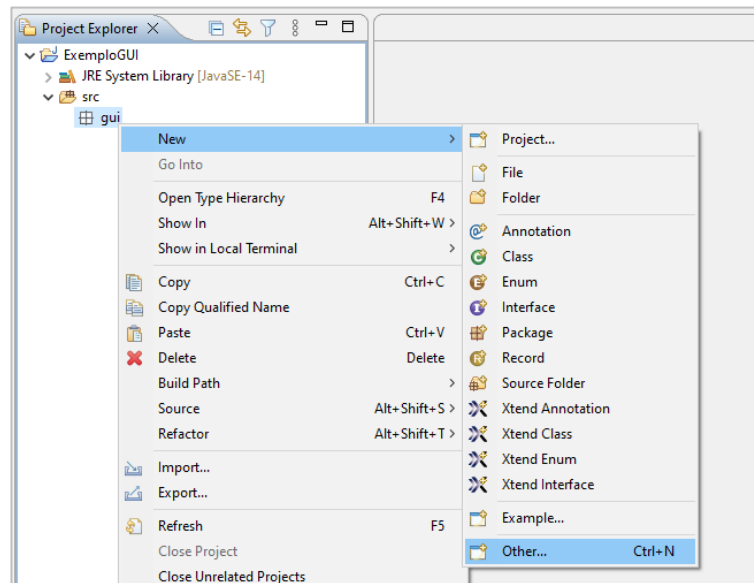
Nas próximas janelas, confirme a instalação e aceite os termos de uso da ferramenta.

Criar JFrame

Window Builder

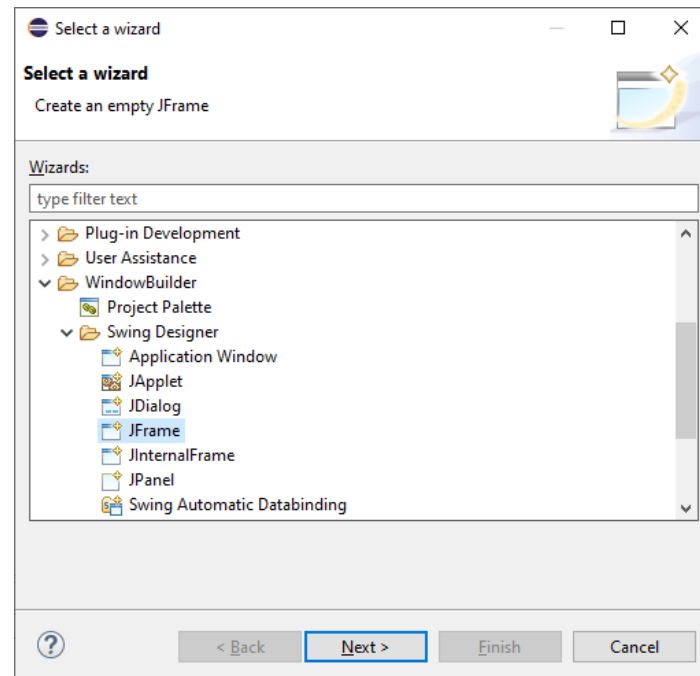
Criar JFrame

*Para criar um novo JFrame utilizando o Window Builder, clique com o botão direito no pacote e selecione as opções: **New > Other...***



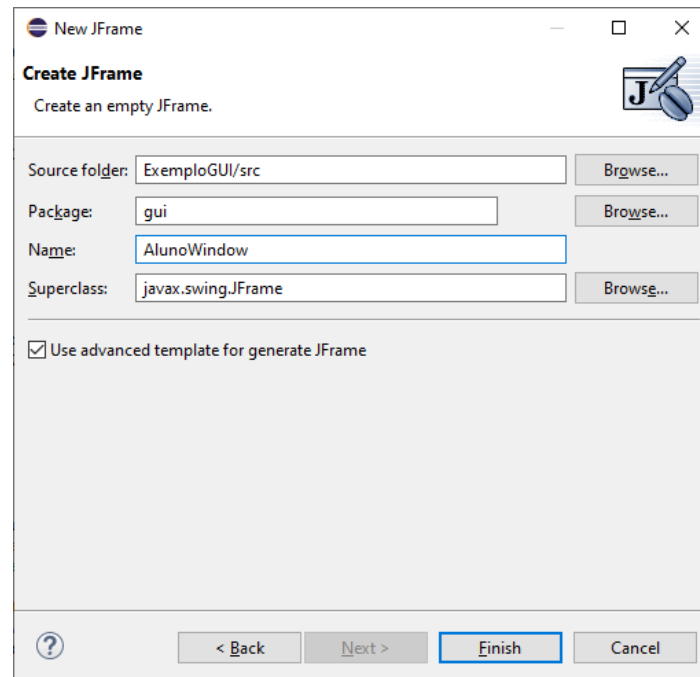
Criar JFrame

*Na janela de diálogo, expanda a opção **WindowBuilder** > **Swing Designer**, e selecione a opção **JFrame**.*



Criar JFrame

*Para este exemplo, nomeou-se o JFrame como **AlunoWindow**.*



Projeto

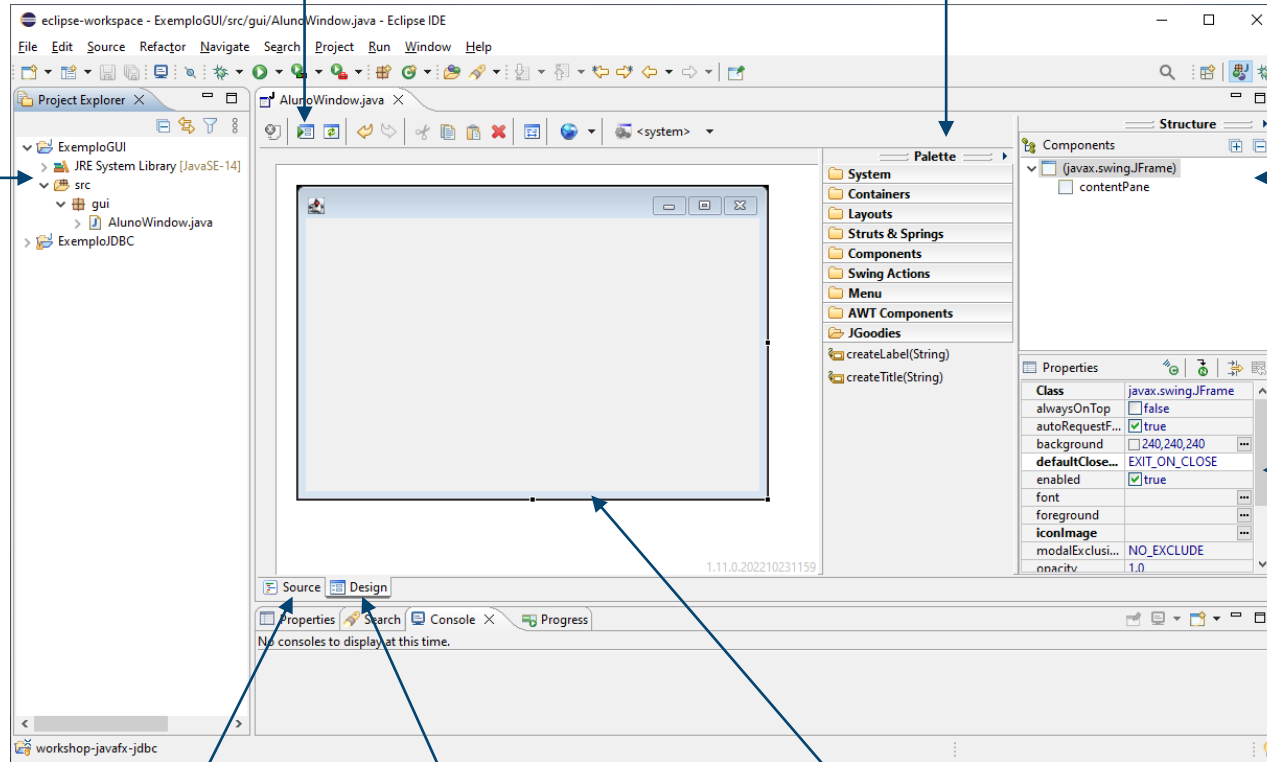
Window Builder

Estrutura do Projeto

Prévia da GUI

Paleta de Componentes

Estrutura do JFrame



Código-Fonte

Projeto da GUI

JFrame

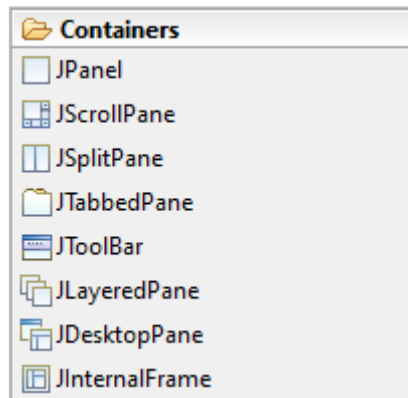
Propriedade de um Componente

Paleta

Window Builder

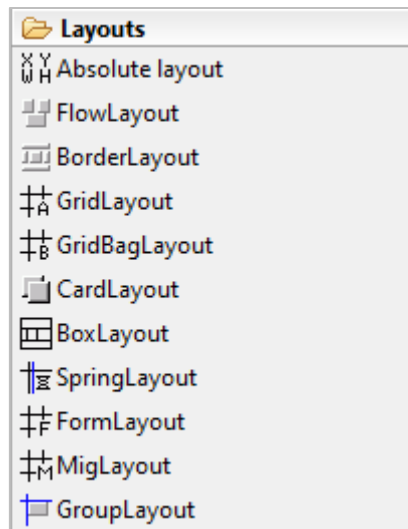
Containers

Qualquer programa que ofereça uma interface gráfica possuirá ao menos um container.



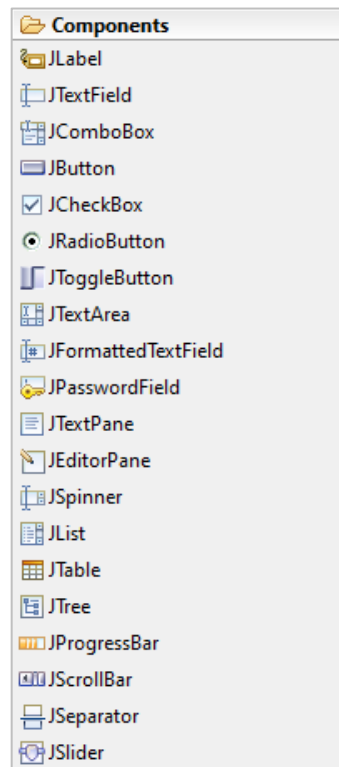
Layouts

Os layout definem como os componentes serão dispostos no container.



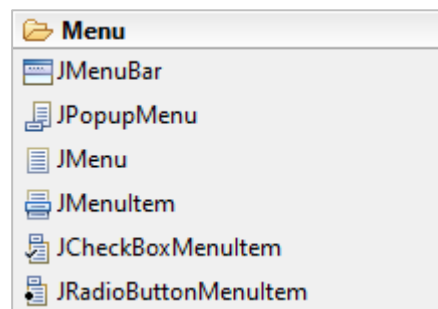
Components

O componente são fundamentais na interface com o usuário.



Menu

Para organização dos containers e seus componentes, os aplicativos podem ser organizados por meios dos Menus

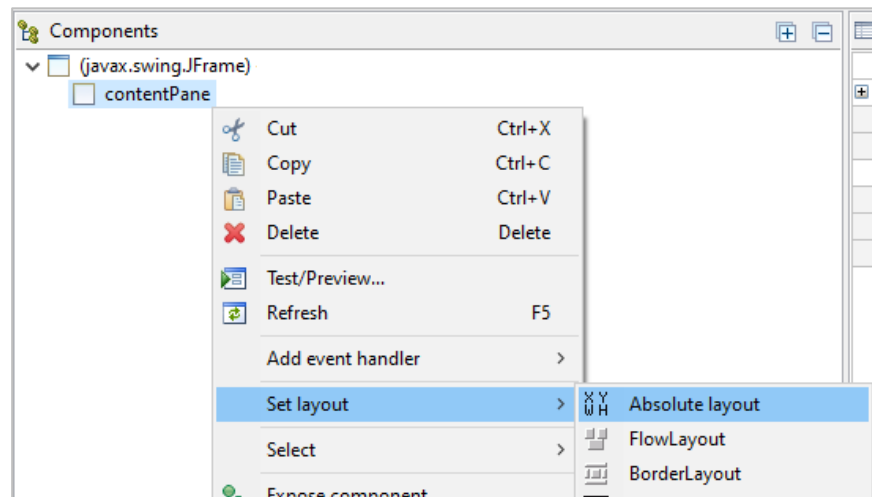


Exemplo GUI

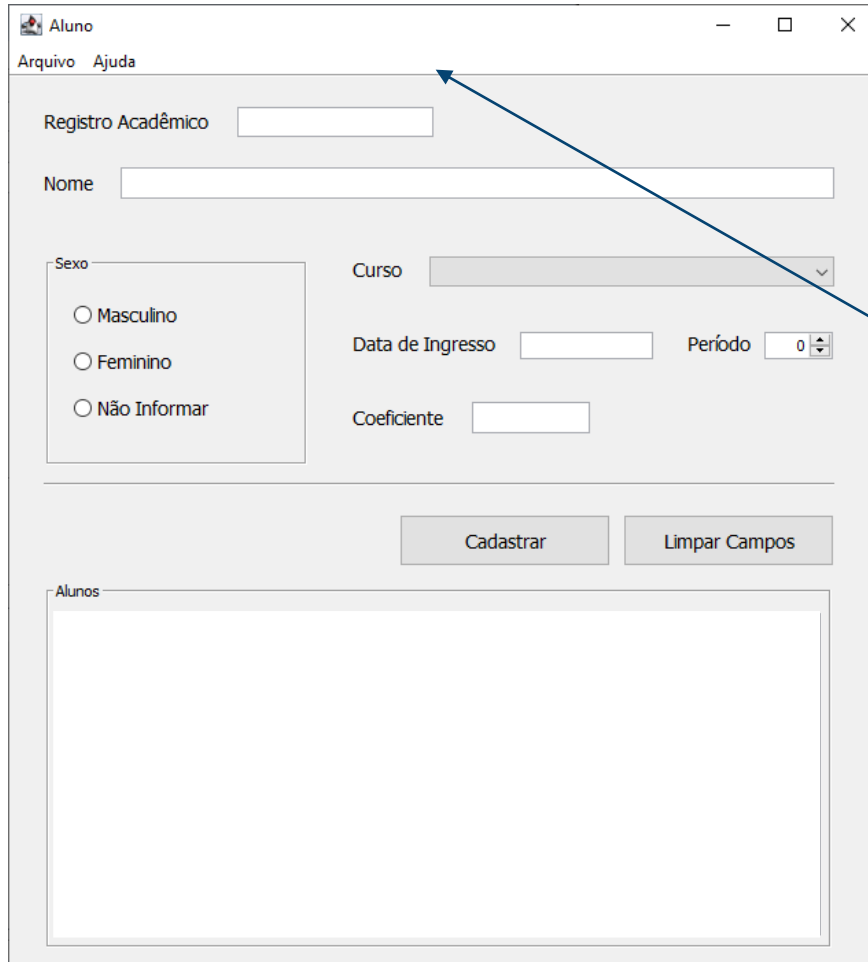
Window Builder

Exemplo GUI

Para adicionar os componentes em qualquer local do Jframe, na estrutura do JFrame, defina o layout do contentPane como Absolute Layout.



Exemplo GUI



The image shows a Java Swing window titled "Aluno" with a standard Mac OS-style title bar (minimize, maximize, close buttons). Below the title bar is a menu bar with two items: "Arquivo" and "Ajuda". The main content area is divided into several sections:

- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A spinner box with the value "0".
- Coeficiente:** A text input field.

Below these fields are two buttons: "Cadastrar" and "Limpar Campos". At the bottom of the window is a scrollable area labeled "Alunos" which is currently empty.

JMenuBar

Exemplo GUI

The image shows a Java Swing window titled "Aluno" with a standard Mac OS-style title bar (minimize, maximize, close buttons). The window contains a menu bar with two items: "Arquivo" and "Ajuda". Below the menu bar, the form is organized into several sections:

- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A dropdown menu.
- Coeficiente:** A text input field.

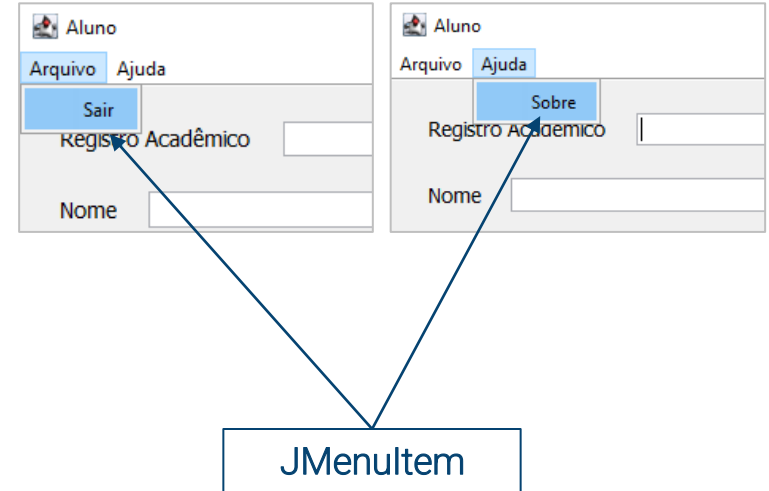
Below these fields are two buttons: "Cadastrar" and "Limpar Campos". At the bottom of the window is a large, empty rectangular area labeled "Alunos" in its top-left corner, likely intended for a list of students.

JMenu

Exemplo GUI

A Java Swing window titled "Aluno" with a standard Mac OS-style title bar (red, yellow, green buttons and window controls). The menu bar contains "Arquivo" and "Ajuda". The main content area is a form for student registration with the following fields and controls:

- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A spinner control with a value of 0.
- Coeficiente:** A text input field.
- Buttons:** "Cadastrar" and "Limpar Campos" are located at the bottom right of the form area.
- Alunos:** A list box at the bottom of the window, currently empty.



Exemplo GUI

The image shows a Java Swing window titled "Aluno" with a menu bar containing "Arquivo" and "Ajuda". The window contains several input fields and a list area. Arrows point from a box labeled "JLabel" to the following fields:

- Registro Acadêmico
- Nome
- Curso
- Data de Ingresso
- Período
- Coeficiente

The form also includes a section for "Sexo" with radio buttons for "Masculino", "Feminino", and "Não Informar". At the bottom, there are two buttons: "Cadastrar" and "Limpar Campos". A list area at the bottom is labeled "Alunos".

Exemplo GUI

The image shows a Java Swing window titled "Aluno" with a menu bar containing "Arquivo" and "Ajuda". The window contains several input fields and a list of students. Three blue arrows point from a box labeled "JTextField" to three specific input fields: "Registro Acadêmico", "Nome", and "Coeficiente".

Registro Acadêmico

Nome

Sexo

☐ Masculino

☐ Feminino

☐ Não Informar

Curso

Data de Ingresso Período

Coeficiente

Cadastrar Limpar Campos

Alunos

Exemplo GUI

The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS X title bar (red, yellow, and green buttons). The window contains a menu bar with "Arquivo" and "Ajuda". The main content area is divided into several sections:

- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A spinner box with the value "0".
- Coeficiente:** A text input field.

Below these fields are two buttons: "Cadastrar" and "Limpar Campos". At the bottom of the window is a scrollable area labeled "Alunos" which is currently empty.

JFormattedTextField

Exemplo GUI

The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS X title bar (red, yellow, and green buttons). The window contains the following elements:

- Menu Bar:** "Arquivo" and "Ajuda".
- Form Fields:**
 - "Registro Acadêmico": A text input field.
 - "Nome": A text input field.
 - "Sexo": A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
 - "Curso": A JComboBox (dropdown menu) with a downward arrow.
 - "Data de Ingresso": A text input field.
 - "Período": A spinner field with the value "0".
 - "Coeficiente": A text input field.
- Buttons:** "Cadastrar" and "Limpar Campos".
- List Area:** A scrollable area titled "Alunos" at the top, which is currently empty.

JComboBox

Exemplo GUI

The image shows a Java Swing window titled "Aluno" with a standard Mac OS X title bar (close, maximize, and zoom buttons). The window contains a menu bar with "Arquivo" and "Ajuda". The main content area is divided into several sections:

- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A JSpinner control showing the value "0".
- Coeficiente:** A text input field.

Below these fields are two buttons: "Cadastrar" and "Limpar Campos". At the bottom of the window is a scrollable area labeled "Alunos" which is currently empty.

JSpinner

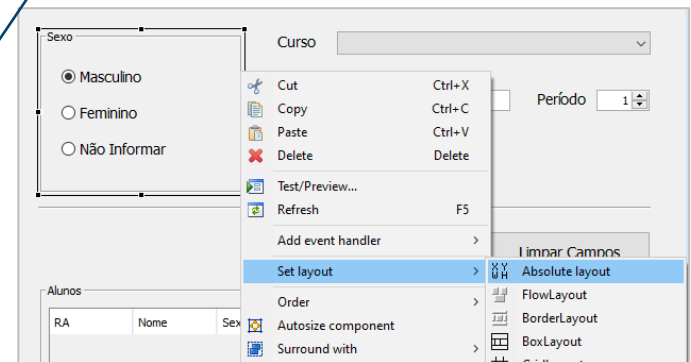
Exemplo GUI

The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS-style title bar (close, maximize, and zoom buttons). The window contains a menu bar with "Arquivo" and "Ajuda". The main area is a registration form with the following elements:

- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino" (selected), "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A spinner field with the value "0".
- Coeficiente:** A text input field.
- Buttons:** "Cadastrar" and "Limpar Campos" at the bottom right.
- Alunos:** A table at the bottom with columns "RA", "Nome", and "Sex". The table is currently empty.

JPanel

ao adicionar ambos os
JPanels, atribua layout
como absoluto



Exemplo GUI

The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS-style title bar (red, yellow, green buttons). The window contains the following elements:

- Menu Bar:** "Arquivo" and "Ajuda".
- Form Fields:**
 - "Registro Acadêmico": A text input field.
 - "Nome": A text input field.
 - "Sexo": A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
 - "Curso": A dropdown menu.
 - "Data de Ingresso": A text input field.
 - "Período": A spinner box with the value "0".
 - "Coeficiente": A text input field.
- Buttons:** "Cadastrar" and "Limpar Campos".
- List View:** A panel titled "Alunos" containing a large empty rectangular area, likely for a list of students.

JRadioButton

Exemplo GUI

Aluno

Arquivo Ajuda

Registro Acadêmico

Nome

Sexo

☐ Masculino

☐ Feminino

☐ Não Informar

Curso

Data de Ingresso Período

Coeficiente

Cadastrar Limpar Campos

Alunos

JSeparator

Exemplo GUI

The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS-style title bar (red, yellow, and green buttons). The window contains the following elements:

- Menu Bar:** "Arquivo" and "Ajuda".
- Form Fields:**
 - "Registro Acadêmico": A single-line text input field.
 - "Nome": A single-line text input field.
 - "Sexo": A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
 - "Curso": A dropdown menu.
 - "Data de Ingresso": A date input field.
 - "Período": A spinner field with the value "0".
 - "Coeficiente": A single-line text input field.
- Buttons:** Two buttons at the bottom: "Cadastrar" and "Limpar Campos".
- List Area:** A large empty rectangular area at the bottom labeled "Alunos".

JButton

Exemplo GUI

The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS X title bar (red, yellow, and green buttons). The window contains a menu bar with "Arquivo" and "Ajuda". The main content area is a registration form with the following fields and controls:

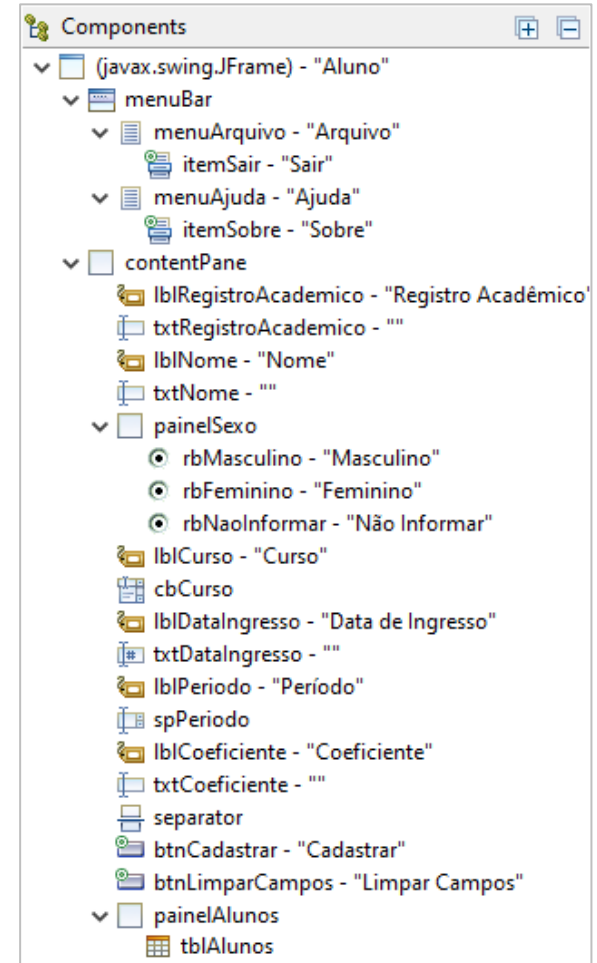
- Registro Acadêmico:** A text input field.
- Nome:** A text input field.
- Sexo:** A group box containing three radio buttons: "Masculino", "Feminino", and "Não Informar".
- Curso:** A dropdown menu.
- Data de Ingresso:** A text input field.
- Período:** A spinner box with the value "0".
- Coefficiente:** A text input field.

Below the form are two buttons: "Cadastrar" and "Limpar Campos". At the bottom of the window is a table area labeled "Alunos" which contains an empty rectangular frame, representing a `JTable`.

JTable

Exemplo GUI

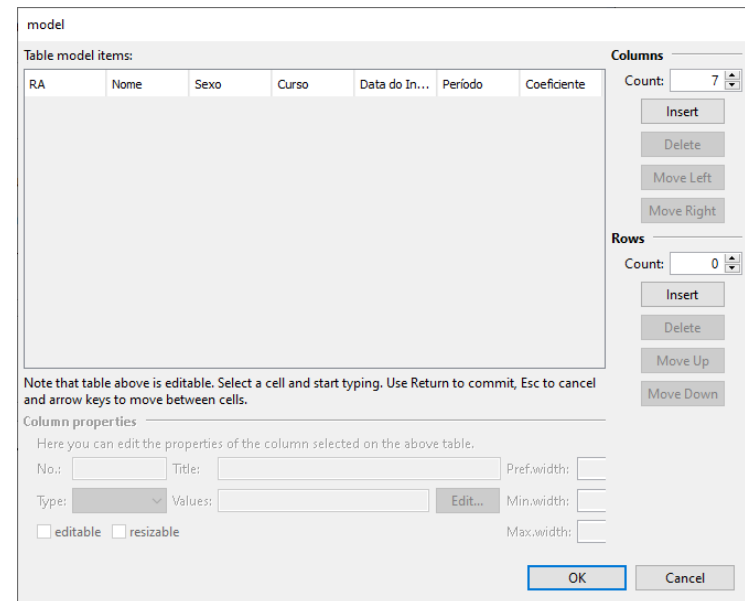
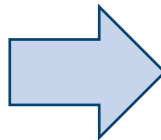
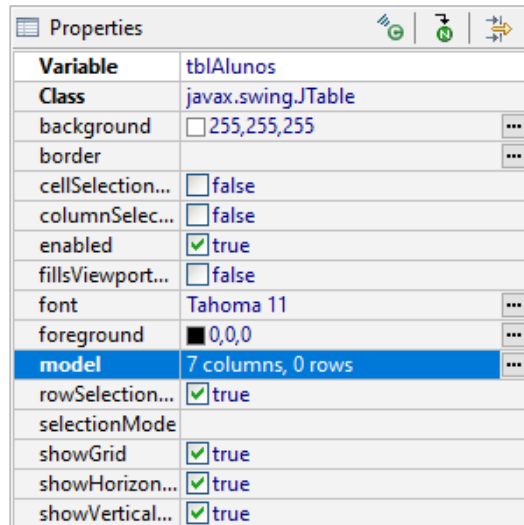
The screenshot shows a Java Swing window titled "Aluno" with a standard Mac OS-style title bar (close, maximize, and zoom buttons). The window has a menu bar with "Arquivo" and "Ajuda". The main content area is a registration form for a student. It includes a "Registro Acadêmico" text field, a "Nome" text field, a "Sexo" group box with three radio buttons ("Masculino", "Feminino", "Não Informar"), a "Curso" dropdown menu, a "Data de Ingresso" text field, a "Período" spinner field (set to 0), and a "Coeficiente" text field. At the bottom right are two buttons: "Cadastrar" and "Limpar Campos". At the bottom left is a panel titled "Alunos" containing a large empty rectangular area, likely for a table.



JTable

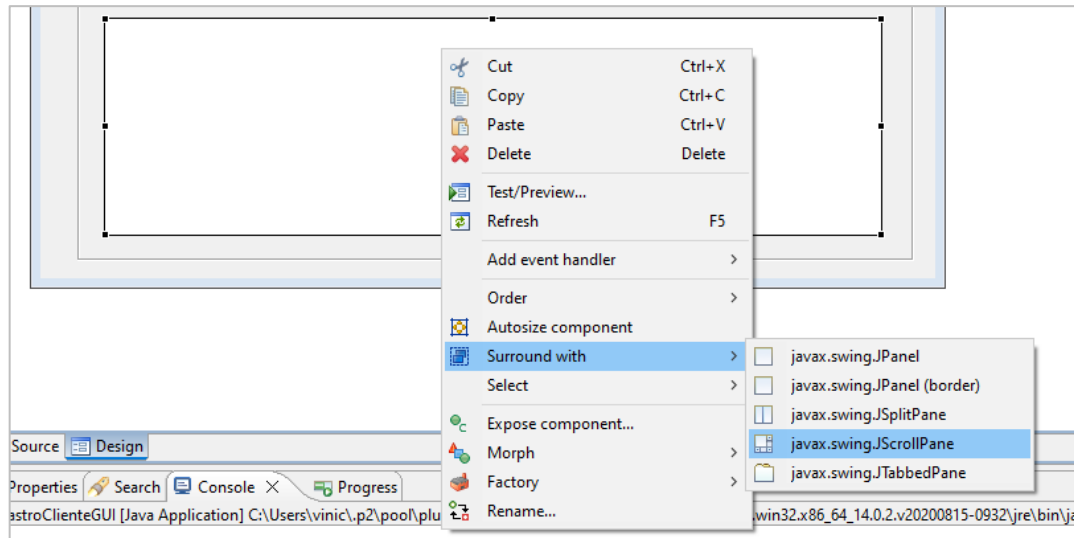
JTable

*Para inserir os cabeçalhos no **JTable**, selecione-o e modifique a propriedade **Model**.*



JTable

*Para conseguir visualizar a modificação, pressione com o botão direito no **JTable** e selecione as opções: **Surround with** > **JScrollPane**.*



JTable

Caso os nomes ou a ordem das colunas não estiverem corretas, os detalhes podem ser modificados no código-fonte.

```
tblAlunos = new JTable();
scrollPane.setViewportView(tblAlunos);
tblAlunos.setModel(new DefaultTableModel(
    new Object[][] {
    },
    new String[] {
        "RA", "Nome", "Sexo", "Curso", "Data do Ingresso", "Período", "Coeficiente"
    }
));
```

Radio Buttons

Radio Buttons

*Para impossibilitar que o usuário do sistema selecione mais de um sexo, é necessário agrupar os **RadioButtons** em um grupo de botões.*

```
rbNaoInformar.setBounds(18, 97, 109, 31);
painelSexo.add(rbNaoInformar);

// agrupa os RadioButton relacionados ao sexo
btnGroupSexo = new ButtonGroup();
btnGroupSexo.add(rbMasculino);
btnGroupSexo.add(rbFeminino);
btnGroupSexo.add(rbNaoInformar);

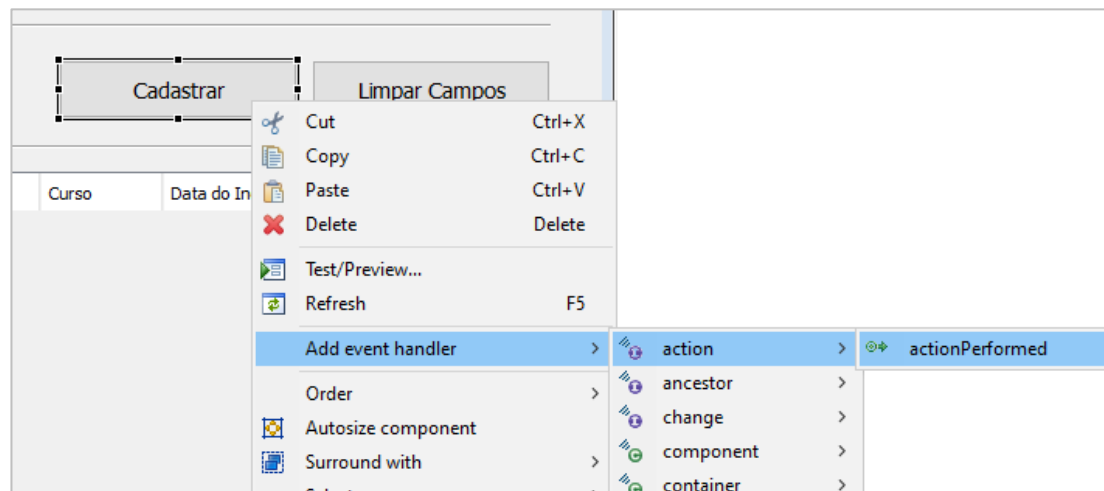
lblCurso = new JLabel("Curso");
lblCurso.setFont(new Font("Tahoma", Font.PLAIN, 14));
```

Inserir Eventos

Botão Cadastrar

Cadastrar

*Para inserir eventos aos componentes, pressiona com o botão direito do mouse no componente e escolhe o evento a ser inserido, por exemplo, o evento **ActionPerformed** ao botão **Cadastrar**.*



Cadastrar

A própria ferramenta adiciona o método `actionPerformed()` ao botão, porém, para manter a organização, cria-se um novo método que será invocado pelo método `actionPerformed()`

```
btnCadastrar = new JButton("Cadastrar");
btnCadastrar.addActionListener(new ActionListener() {
    public void actionPerformed(ActionEvent arg0) {

        btnCadastrarActionPerformed();
    }
});
btnCadastrar.setFont(new Font("Tahoma", Font.PLAIN, 14));
btnCadastrar.setBounds(284, 319, 154, 38);
contentPane.add(btnCadastrar);
```