# Java swing Roteiro para criar uma Tela

Prof. Richard Henrique de Souza, Dr.

# Crie o Projeto

- No Netbeans
  - Menu File  $\rightarrow$  New Project

Source

Ctrl+Shift+N

Ctrl+Shift+O

Ctrl+N

New Project.

Open Project...

P New File..

- Escolha Java with Ant
  - Java Applications

Steps	Choose Project	
1. Choose Project 2	🤍 Filter:	
	Categories: Java with Maven Java with Gradle Java with Ant JavaFX Java Web Java Enterprise NetBeans Modules HTML5/JavaScript C/C++ PHP Samples	Projects: Java Application Java Class Library Java Project with Existing Sources Java Modular Project Java Free-Form Project
	Description:	
	Creates a new Java SE application class in the project. Standard project and debug your project.	in a standard IDE project. You can also generate a main ts use <b>an IDE-generated Ant build script</b> to build, run,

- Digite o nome do projeto
  - Exemplo: aula14
  - Clique em Finish

			Name and Location	teps	
		ila14	Project Name:	<ol> <li>Choose Project</li> <li>Name and Location</li> </ol>	
Browse	ts\NetBeansProjects	\Users\RIHS\Docum	Project Location:		
	ts\NetBeansProjects\aula14	\Users\RIHS\Docum	Project Folder:		
		ler for Storing Librari	Use Dedicated F		
Browse			Libraries Folder:		
	cts can share the same : Help for details).	fferent users and pro mpilation libraries (s			
		aula14.Aula14	Create Main Cla		
	t> Finish Cancel	< Back N			

# Crie os pacotes

- Botão direito em "Source Packages"
  - New → Java Package



- Package Name: digite Principal
  - Clique em finish

Steps 1. Choose File Type 2. Name and Location	Name and Locat	tion	
<ol> <li>Choose File Type</li> <li>Name and Location</li> </ol>	Package Name:		
	r octage Hamer	Principal	
	Project:	aula14	
	Location:	Source Packages	~
	Created Folder:	$\label{eq:c:UsersRIHS} C: \label{eq:c:UsersRIHS} C: eq:c:User$	

< Back

Next >

Finish

Cancel

Help

# Crie os pacotes

• Crie o pacote view



- Agora temos 2 pacotes: Principal e View
- Se quiser já crie os pacotes Model e DAO

# Vamos Criar a TelaPrincipal

- Botão direito no pacote View
  - Selecione Jframe Form...



• Classe Name: TelaPrincipal

칮 New JFrame Form		×
Steps	Name and Loc	cation
<ol> <li>Choose File Type</li> <li>Name and Location</li> </ol>	Class <u>N</u> ame:	TelaPrincipal
	<u>P</u> roject:	aula14
	Location:	Source Packages
	Pac <u>k</u> age:	View
	<u>C</u> reated File:	's\richard\Documents\NetBeansProjects\aula14\src\View\TelaPrincipal.java
	<u>S</u> uperclass:	Browse
	Interfaces:	Browse
		< Back Next > Finish Cancel Help

# Tela criada

Projects × Services Files	va 🐼 Pessoa.java 🗙 🗟 GerenciaAluno.java 🗙 🗟 TelaPrincipal.jav	a x 🔹 🕨 🗗	Palette ×		
🔻 🤓 aula14			Swing Containers	6	
Gource Packages			Panel 🚺	📕 Tabbed Pane 🛛 🔟 Split P	ane 🛛 🔜 Scroll Pane
Principal View	Sector of the context menu to access available useful actions for the selected of the selected	ed components. x	💷 Tool Bar 🚦	Ӓ Desktop Pane 🛛 🕅 Interna	I Frame 🔳 Layered Pane
TelaPrincipal java			Swing Controls		
▶ 🝙 Libraries			label Label	Interpretation	Toggle Button
Aula2     Aula2			Image: Image: Image: Box		°⊂ Button Group
<ul> <li>SavaAppication to</li> <li>Projeto_Estoque</li> </ul>			📧 Combo Box	🧮 List	Text Field
			t× Text Area	Scroll Bar	ѿ Slider
			💷 Progress Bar	Formatted Field	Password Field
			💷 Spinner	Separator	T Text Pane
			🖆 Editor Pane	ਵਿੀ Tree	Table
			<ul> <li>Swing Menus</li> <li>Swing Windows</li> <li>Swing Fillers</li> <li>AWT</li> <li>Beans</li> </ul>		

- Selecione a tela
  - Vá em Properties
    - No title coloque "Interface de Teste"

Properties Events	Code	
<ul> <li>Properties</li> </ul>		
defaultCloseOperation	EXIT_ON_CLOSE	▼ (
title	Interface de Teste	C.
<ul> <li>Other Properties</li> </ul>		

# Vamos adicionar os Menus

- Primeiro o JMenuBar
  - No palette  $\rightarrow$  Swing Menus  $\rightarrow$  Menu Bar

		r ano			
¥.	Swing	Menus			
File	Menu	Bar		🗄 Menu	
	Menu	ltem		🖂 Menu I	tem / CheckBox
⊗	Menu	javax.swing.JMe A container for r	enuBar nenus and me	enu items.	Menu
	Separ	ator			I

• Arraste e solte em cima da Tela

File Edit			

 Selecione o primeiro Menu, dê um clique para habilitar a edição e mude o texto para Arquivo



 Selecione o segundo Menu, dê um clique para habilitar a edição e mude o texto para Sobre



- Agora vamos adicionar os itens de Menu
  - Vá em palette  $\rightarrow$  Swing Menus  $\rightarrow$  Menu Item

<ul> <li>Swing Menus</li> </ul>	
File Menu Bar	Ē
🥅 Menu Item	
💽 Menu Item / RadioButton	E
E Separator	

• Arraste para dentro no Menu Arquivo



• Repita mais duas vezes



• Selecione o primeiro JMenultem e mude o texto para Cadastrar Aluno

Arquivo	Sobre		
N	Cadastra	r Aluno	
jN	/lenultem:	shortcut	
jN	/lenultem	shortcut	

 Selecione o segundo JMenultem e mude o texto para Gerenciar Alunos



• Selecione o terceiro JMenultem e mude o texto para Sair



#### Vamos Testar

#### • Crie a classe Principal no pacote Principal

v 🂩 aula14	l Irea Backagan		Source Design History	
	New Find	► Ctrl+F	Arquivo_Sobre Folder JFrame Form	Name and Location
► 🍻 Aula2 ► 🧽 JavaAj ► 🍉 Projeti	Cut Copy Paste Delete Refactor	CtrI+X CtrI+C CtrI+V Delete	Java Package     Java Class     Java Exception     Java Interface     JPanel Form	<u>P</u> roject: aula Location: Sou
	Compile Package Test Package Run Selenium Tests History	F9 Ctrl+F6	<ul> <li>Entity Class</li> <li>Entity Classes from Database</li> <li>Web Service Client</li> <li>Other</li> </ul>	Pac <u>k</u> age: Prin <u>C</u> reated File: rs\rin Superclass:
	Tools	•		Interfaces:

Name and Loc	cation
Class <u>N</u> ame:	Principal
<u>P</u> roject:	aula14
Location:	Source Packages
Pac <u>k</u> age:	Principal
<u>C</u> reated File:	rs\richard\Documents\NetBeansProjects\aula14\src\Principal\Principal.java
<u>S</u> uperclass:	Browse
Interfaces:	Browse
	< Back Next > Finish Cancel Help

# Crie o código para executar a Tela

```
package Principal;
import View.TelaPrincipal;
public class Principal {
      public static void main(String args[]) {
                                                                                        Execute e veja
            TelaPrincipal objetotela = new TelaPrincipal();
                                                                                          o resultado
            objetotela.setVisible(true);
                                                                                     package Principal;
                                                                                             Navidate
                                                                                    import Vi
                                                                                             Show Javadoc
                                                                                                         Alt+F1
                                                                                             Find Usages
                                                                                                         Alt+F7
                                                                                             Call Hierarchy
                                                                                     public cl
                                                                                             Insert Code..
                                                                                                         Alt+Insert
                                                                                             Fix Imports
                                                                                                         Ctrl+Shift+I
                                                                                        publi
                                                                                             Refactor
                                                                                                         Alt+Shift+F
                                                                                             Format
```

Run File

Toot File

Debug File

Shift+F6

Otd-E6

Ctrl+Shift+F5

# Resultado até agora

- Criamos a Tela
- Mas ainda não faz nada....

	hfar / /nhhaa	+ ( Creation Di 1	- Creation	- [
-	실 Interface de Teste		_	×
7	Arquivo Sobre			
1	Cadastrar Aluno			
	Gerenciar Alunos			
e	Sair			
а				
d				
e				
r				
Ĩ				

#### Vamos Cadastrar Aluno

- No pacote view
  - New  $\rightarrow$  JFrame Form





#### Classe Name: CadastroAluno

Name and Loc	lame and Location					
Class <u>N</u> ame:	CadastroAluno					
<u>P</u> roject:	aula14					
Location:	Source Packages					
Pac <u>k</u> age:	View					
<u>Created File:</u>	<pre>\richard\Documents\NetBeansProjects\aula14\src\View\CadastroAluno.java</pre>					
Superclass:	Browse					
Interfaces:	Browse					
	< Back Next > Finish Cancel Help					

• Selecione a Tela e mude o title

[JFrame] - Properties ×			
Properties Events	Code		
Properties			
defaultCloseOperation		EXIT_ON_CLOSE	
title		Cadastro de Aluno	
Other Properties			

• Em palette, procure Swing Controls→ Label.

Swing Containers					
Panel	🛅 Tabbed Pane	📃 Split Pane	E Scroll P		
💷 Tool Bar	Desktop Pane	🚞 Internal Frame	🔳 Layered		
Swing Controls	\$				
label Label	💷 📧 Button	ION TO	oggle Button		
Image: Check E javax.s A disp	wing.JLabel lay area for a short te	xt string or an image	, or both.		
Combo Box	El LISI		extrieid		
<sup>t</sup> ≍ Text Area	💷 Scroll Ba	ır 🗘 Si	lider		
Dragrada Dar	E. Formette	d Field - D	accurated Fiel		

#### • Arraste 4 Labels



- Mude os nomes dos Labels.
- Selecione o Label e dê um clique, e troque o texto para Nome:



• Idade:

Nome:

- Curso:
- Fase:



- Procure Por TextField.
  - Arraste 4 vezes o componente TextField para a tela de Cadastro

Swing Controls		
label Label	🔤 Button	Toggle Button
Interpretation Interpretatio Interpretatio Interpretation Interpretation Inte		<sup>®</sup> <sup>_−</sup> Button Group
🖃 Combo Box	🧮 List	Text Field
t× Text Area	Scroll Bar	🗘 Slider
💷 Progress Bar	Form javax.swing.JTe	xtField mponent that allows the edit
	of a single line of	of text.

Nome:	jTextField1	
Idade: 🗠	jTextField2	
Curso	iTextField3	
	Jiona Iorao	
Errer	Teatral	
Fase:	I ext⊢leid4	

- Agora vamos mudar os nomes das "variáveis" (atributos) de forma a identificar melhor os componentes no código e possibilitar o acesso ao texto digitado pelo usuário.
- Com o botão direito no primeiro JTextField1 e selecione Change Variable Name...
  - Mude para c\_nome
- Mude os outros para
  - c\_idade
  - c\_curso
  - c\_fase



🟮 Rename			×
<u>N</u> ew Name:	c_nome		
		ОК	Cancel

• Clique em source e veja como o nome mudou.

164

FextField4	[JTextField] -	Navigator 🔉
------------	----------------	-------------

🔻 🖻 Form CadastroAluno

- 🕨 🔄 Other Components
- 🔻 📰 (JFrame)
  - <sup>⊯bei</sup>jLabel1 (JLabel)
  - 🔤 jLabel2 (JLabel)
  - <sup>label</sup> jLabel3 (JLabel)
  - <sup>iabei</sup> jLabel4 (JLabel)
  - c\_nome [JTextField]
  - c\_idade [JTextField]
  - c\_curso [JTextField]
  - c\_fase [JTextField]

Source	🛛 Design History 🔀 🖓 🕶 🐺 🔍 🧖 😓 🦂 🥐 😓 🗞
.53	
.54	<pre>// Variables declaration - do not modify</pre>
.55	<pre>private javax.swing.JTextField c_curso;</pre>
.56	<pre>private javax.swing.JTextField c_fase;</pre>
.57	<pre>private javax.swing.JTextField c_idade;</pre>
.58	<pre>private javax.swing.JTextField c_nome;</pre>
.59	<pre>private javax.swing.JLabel jLabel1;</pre>
.60	<pre>private javax.swing.JLabel jLabel2;</pre>
.61	<pre>private javax.swing.JLabel jLabel3;</pre>
.62	<pre>private javax.swing.JLabel jLabel4;</pre>
.63	<pre>// End of variables declaration</pre>

- Clique em design.
  - Selecione o TextField, dê um clique e apague o texto.
  - Repita a operação em todos

Source	Design	History	<b>(</b> ]	¥	
Nom	ne:				]
Idad	e: jTextF	ield2			
Curs	so: jTextF	ïeld3			
Fase	e: jTextF	ield4			

Nome:	
Idade: 👓	
Curso:	
Fase:	



- Procure o componente Button.
  - E arraste 2 para a tela de CadastroAluno



- Selecione o primeiro botão e dê um clique.
  - Mude o Texto para Cancelar



- Selecione o segundo botão e dê um clique.
  - Mude o Texto para Cadastrar

Nome:			
Idade:			
Curso:			
Fase:			
	Cancelar	Cadastrar	

# Voltamos para a TelaPrincipal

 No item de Menu Cadastrar Aluno, use o botão direito e selecione Events→ Action→ actionPerformed.

Arquivo Sob	re				
Cadas Gereno Soir	Edit Text Change Variable Name				
Jaii	Events	►	Action	•	oction Derformed
	Align	•	Ancestor	•	actionPenormed
	Anchor		Change	•	

 Vai abrir o método para ser inserido o código que queremos que execute quando o usuário clicar no Menu Cadastrar Aluno

• Digite o código para Abrir a Tela de Cadastro de aluno.

<pre>private void jMenuItem1ActionPerformed(java.awt.event.ActionEvent evt) {</pre>
CadastroAluno objeto = new CadastroAluno();
<pre>objeto.setVisible(true);</pre>
1

#### Vamos Testar

• Execute a classe Principal

실 Interface de Teste	—	×
Arquivo Sobre		
Cadastrar Aluno		
Gerenciar Alunos		
Sair		

• Depois clique em Cadastrar Aluno

📓 Interface de Teste 🛛 — 🗆 🗙								
+ Arquivo Sobre								
🙆 Cadastro de Aluno	_	- [		(				
Nome:			]					
Idade:								
Curso:			]					
Fase:								
Cancelar	Cadastrar							

# Por enquanto apenas criamos as Telas

- Agora vamos fazer funcionar algo....
  - Primeiro copie as classes já feitas nas aula anteriores para os pacotes Model e DAO.
  - Então primeiro crie os pacotes...



#### Agora Cria as Classes Pessoa e Aluno



# Lembrando: Classe Pessoa

```
package Model;
public abstract class Pessoa {
  private int id;
  private String nome;
  private int idade;
  public Pessoa() {
  public Pessoa(int id, String nome, int idade) {
    this.id = id;
    this.nome = nome;
    this.idade = idade;
```

```
public int getId() {
    return id;
 public void setId(int id) {
    this.id = id;
public String getNome() {
    return nome;
 public void setNome(String nome)
    this.nome = nome;
 public int getIdade() {
    return idade;
 public void setIdade(int idade) {
    this.idade = idade;
```

# Relembrando Classe Aluno

```
package Model;
import java.util.*;
import DAO.AlunoDAO;
public class Aluno extends Pessoa {
  private String curso;
  private int fase;
  public Aluno() {
  public Aluno(String curso, int fase) {
    this.curso = curso;
    this.fase = fase;
  public Aluno(String curso, int fase, int id, String nome,
int idade) {
    super(id, nome, idade);
    this.curso = curso;
    this.fase = fase;
```

```
public String getCurso() {
    return curso;
  public void setCurso(String curso) {
    this.curso = curso;
  public int getFase() {
    return fase;
  public void setFase(int fase) {
    this.fase = fase;
  @Override
  public String toString() {
    return "\n ID: " + this.getId()
         + "\n Nome: " +
this.getNome()
         + "\n Idade: " + this.getIdade()
         + "\n Curso: " + this.getCurso()
         + "\n Fase:" + this.getFase()
         + "\n -----";
```

# Crie a Classe AlunoDAO no pacote DAO



package DAO;

import Model.Aluno; import java.util.\*;

```
public class AlunoDAO {
  public static ArrayList<Aluno> MinhaLista = new
ArrayList<Aluno>();
  public static int maiorID() {
    int maiorID = 0;
    for (int i = 0; i < MinhaLista.size(); i++) {</pre>
       if (MinhaLista.get(i).getId() > maiorID) {
         maiorID = MinhaLista.get(i).getId();
    return maiorID;
```

# Relembrando a classe AlunoDAO

```
public ArrayList getMinhaLista() {
    return AlunoDAO.MinhaLista;
```

public boolean InsertAlunoBD(String curso, int fase, String
nome, int idade) {

```
int id = this.maiorID() + 1;
```

Aluno objeto = new Aluno(curso, fase, id, nome, idade); AlunoDAO.MinhaLista.add(objeto); return true;

```
}
```

public boolean DeleteAlunoBD(int id) {
 int indice = this.procuraIndice(id);
 AlunoDAO.MinhaLista.remove(indice);
 return true;

```
}
```

public boolean UpdateAlunoBD(String curso, int fase, int id, String nome, int idade) {

Aluno objeto = new Aluno(curso, fase, id, nome, idade); int indice = this.procuraIndice(id);

AlunoDAO.MinhaLista.set(indice, objeto); return true;

```
Agora que temos a classe AlunoDAO
Voltamos na Classe Aluno e
adicionamos os métodos
  private int procuralndice(int id) {
     int indice = -1;
     for (int i = 0; i < AlunoDAO.MinhaLista.size();</pre>
 i++) {
        if (AlunoDAO.MinhaLista.get(i).getId() ==
 id) {
          indice = i;
     return indice;
   public Aluno carregaAluno(int id) {
     int indice = this.procuralndice(id);
      return AlunoDAO.MinhaLista.get(indice);
   public int maiorID(){
     return AlunoDAO.maiorID();
```

# Agora voltamos para a Classe (Tela) CadastroAluno

- Vá em source
- E adicione os códigos:



# Volte em Design

- Botão direito no botão Cadastrar
  - Events  $\rightarrow$  Action  $\rightarrow$  action Performed

	Edit Text Change Variable Name				Button2 [JButton] - Pro Properties Event:
	Events	►	Action	►	actionPortformed
Cadast	Align	- ►	Ancestor	►	
- <u>-</u> - <u>-</u> - <u>-</u>	Anchor	•	Change	►	ackground
	Auto Resizing	•	Component	•	pnt

# Vai abrir o método

- Digite o código
- Note vai dar erro em Mensagens
- Vamos adicionar a classe Mensagens

```
try {
   String nome = "";
   int idade = 0;
   String curso = "";
   int fase = 0;
```

```
if (this.c_nome.getText().length() < 2) {
    throw new Mensagens("Nome deve conter ao menos
2 caracteres.");
    } else {
        nome = this.c_nome.getText();
    }</pre>
```

```
} catch (Mensagens erro) {
    JOptionPane.showMessageDialog(null,
erro.getMessage());
    } catch (NumberFormatException erro2) {
        JOptionPane.showMessageDialog(null, "Informe um
número.");
    }
```

# Crie a Classe Mensagens



# Digite o código na Classe Mensagens

package View;

public class Mensagens extends Exception {

Mensagens(String msg) {
 super(msg);

```
Volte em
cadastroAluno e
termine de digitar o
código
```

```
private void
jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
    try {
      String nome = "";
       int idade = 0;
       String curso = "";
       int fase = 0;
      if (this.c_nome.getText().length() < 2) {</pre>
         throw new Mensagens("Nome deve conter ao menos
2 caracteres.");
       } else {
         nome = this.c_nome.getText();
      if (this.c_idade.getText().length() <= 0) {</pre>
         throw new Mensagens("Idade deve ser número e
maior que zero.");
       } else {
         idade = Integer.parseInt(this.c_idade.getText());
```

Continua....

```
if (this.c_curso.getText().length() < 2) {</pre>
        throw new Mensagens("Curso deve conter ao menos 2 caracteres.");
      } else {
        curso = this.c_curso.getText();
        if (this.c_fase.getText().length() <= 0) {</pre>
        throw new Mensagens("Fase deve ser número e maior que zero.");
      } else {
        fase = Integer.parseInt(this.c_fase.getText());
        if (this.objaluno.InsertAlunoBD(curso, fase, nome, idade)) {
        JOptionPane.showMessageDialog(rootPane, "Aluno Cadastrado com
```

Sucesso!");

```
this.c_nome.setText("");
this.c_idade.setText("");
this.c_curso.setText("");
this.c_fase.setText("");
```

Continua....

# Parte 3 do código de cadastro

System.out.println(this.objaluno.getMinhaLista().toString());

```
} catch (Mensagens erro) {
    JOptionPane.showMessageDialog(null, erro.getMessage());
} catch (NumberFormatException erro2) {
    JOptionPane.showMessageDialog(null, "Informe um número.");
}
```

#### Teste o sistema

- Execute a Classe Principal
  - E cadastre um Aluno

실 Cadastro de Aluno	—		×
Nome: Richard			
Message		×	
iua i) Aluno Cadastrado co	m Sucesso	!	
Си			
Fase: 2			
Cancelar Cadas	strar		System



#### Volte na Tela de CadastroAluno

• Vá no botão cancelar



# Adicione o código



# Observação

- No material da aula passada a Classe a seguir foi definida como
  - View.GerenciaAluno.java
- Aqui foi criada como
  - View.GerenciamentoAluno.java

# Vamos Criar a Classe GerenciamentoAluno







# Coloque o title

[JFrame] - Properties ×		
Properties Events	Code	
Properties		
defaultCloseOperation		EXIT_ON_CLOSE
title		GerenciamentoAluno
Other Properties		

# Arraste o componente JTable para a Tela de GerenciamentoAluno



# Mude a Tabela para os campos do Aluno

• Botão direito na tabela, selecione Table Contents...



#### Vá em Columns

• Mude os Titles

Columns

Туре

Object

Object

Object

Object

Rows

Resizable

 $\overline{\mathbf{V}}$ 

 $\overline{\mathbf{V}}$ 

Editable

くくく

Customizer Dialog

Table Model

0

Title

Nome

Idade

Curso

ID

#### 🟮 Customizer Dialog

Х

4 🔺

Insert

Delete

Move Up

Move Down

Count:

Table Mo	del Columns F	Rows				
Title	Туре	Resizable	Editable	Count: 4		
Title 1 Title 2 Title 3 Title 4	Object Object Object Object			Insert Delete		
				Move <u>Up</u> Move Do <u>w</u> n		
<u>T</u> itle:	(No Property Edito	r) 🗌 Res	i <u>z</u> able 🗌 <u>E</u> di	table		
Туре:	Object	Pre <u>f</u> . Wie	dth: Default	•		
Edit <u>o</u> r:	(No Property Edito	r) Mi <u>n</u> . Wid	th: Default	•		
<u>R</u> enderer:	(No Property Edito	or) Ma <u>x</u> . Wie	th: Default	<b>v</b>		
Selection Model: Not Allowed						
Allow to reorder columns by drag and drop						
				Close		

 $\times$ 



#### Adicione os outros componentes

ID	Nome	Idade	Curso	Fase
Nome:				
Idade:				
Curso:				
Fase				
	Cancelar	Alterar	Apagar	

# Mude o nome da Tabela



Agora vamos fazer em GerenciaAluno o mesmo que em CadastroAluno

• Vá em source

package View;

import Model.Aluno;

public class GerenciamentoAluno extends javax.swing.JFrame {

```
private Aluno objaluno;
public GerenciamentoAluno() {
    initComponents();
    this.objaluno = new Aluno();
}
```

Adicione o Método carregarTabela() na tela GerenciamentoAluno @SuppressWarnings("unchecked")
public void carregaTabela() {
 DefaultTableModel modelo =
(DefaultTableModel) this.jTableAlunos.getModel();
 modelo.setNumRows(0);

ArrayList<Aluno> minhalista = new ArrayList<>(); minhalista = objaluno.getMinhaLista(); for (Aluno a : minhalista) { modelo.addRow(new Object[]{ a.getId(), a.getNome(), a.getIdade(), a.getCurso(), a.getFase() });

# Volte no Design da tela GerenciamentoAluno

• Vamos adicionar localizar o método mouseClicked



# Adicione o código Para funcionar mude o nome dos JTextFields...

private void jTableAlunosMouseClicked(java.awt.event.MouseEvent evt) {

```
if (this.jTableAlunos.getSelectedRow() != -1) {
```

String nome = this.jTableAlunos.getValueAt(this.jTableAlunos.getSelectedRow(), 1).toString(); String idade = this.jTableAlunos.getValueAt(this.jTableAlunos.getSelectedRow(), 2).toString(); String curso = this.jTableAlunos.getValueAt(this.jTableAlunos.getSelectedRow(), 3).toString(); String fase = this.jTableAlunos.getValueAt(this.jTableAlunos.getSelectedRow(), 4).toString();

this.c\_nome.setText(nome); this.c\_idade.setText(idade); this.c\_curso.setText(curso); this.c\_fase.setText(fase);

# Agora Vamos iniciar a Tela GerenciamentoAluno

- Vá na TelaPrincipal
  - E adicione o código no evento do Menu Gerenciar Aluno



private void jMenuItem2ActionPerformed(java.awt.event.ActionEvent evt) {
 GerenciamentoAluno objeto = new GerenciamentoAluno();
 objeto.setVisible(true);
 objeto.carregaTabela();
}

#### Vamos Testar ?

- Execute a Classe Principal
  - Cadastre um Aluno
  - Entre em Gerenciar Aluno

🔮 GerenciamentoAluno — 🗆 🗙						
ID	Nome	Idade	Curso	Fase		
1	Richard	44	Sis	4		
Nome:	Richard					
ldade:	44					
Curso:	Sis					
Fase	4					
	Cancelar	Altera	ar Apa	gar		

#### Volte em GerenciamentoAluno

• Vá no Botão alterar e vamos adicionar o evento



# Agora adicione o código

```
try {
```

```
int id = 0;
String nome = "";
int idade = 0;
String curso = "";
int fase = 0;
```

```
if (this.c_nome.getText().length() < 2) {
    throw new Mensagens("Nome deve
conter ao menos 2 caracteres.");
    } else {
    nome = this.c_nome.getText();</pre>
```

```
if (this.c_idade.getText().length() <= 0) {</pre>
         throw new Mensagens("Idade deve ser número
e maior que zero.");
      } else {
         idade = Integer.parseInt(this.c_idade.getText());
       if (this.c_curso.getText().length() < 2) {</pre>
         throw new Mensagens("Curso deve conter ao
menos 2 caracteres.");
      } else {
         curso = this.c_curso.getText();
       if (this.c_fase.getText().length() <= 0) {</pre>
         throw new Mensagens("Fase deve ser número e
maior que zero.");
       } else {
         fase = Integer.parseInt(this.c_fase.getText());
```

```
if (this.jTableAlunos.getSelectedRow() == -1) {
         throw new Mensagens("Primeiro Selecione um Aluno
para Alterar");
      } else {
         id =
Integer.parseInt(this.jTableAlunos.getValueAt(this.jTableAlunos.
getSelectedRow(), 0).toString());
if (this.objaluno.UpdateAlunoBD(curso, fase, id, nome, idade)) {
         // limpa os campos
         this.c nome.setText("");
         this.c idade.setText("");
         this.c curso.setText("");
         this.c fase.setText("");
         JOptionPane.showMessageDialog(rootPane, "Aluno
Alterado com Sucesso!");
```

```
} catch (Mensagens erro) {
    JOptionPane.showMessageDialog(null,
erro.getMessage());
    } catch (NumberFormatException erro2) {
        JOptionPane.showMessageDialog(null,
        "Informe um número.");
     } finally {
        carregaTabela(); // atualiza a tabela.
     }
```

}

System.out.println(this.objaluno.getMinhaLista().toString());

# Teste o código

- Cadastre um aluno
- Edite o aluno

#### Vamos Apagar

• No botão Apagar, selecione o evento



```
try {
```

```
int id = 0;
if (this.jTableAlunos.getSelectedRow() == -1) {
    throw new Mensagens("Primeiro Selecione um Aluno para APAGAR");
} else {
    id = Integer.parseInt(this.jTableAlunos.getValueAt(this.jTableAlunos.getSelectedRow(), 0).toString());
}
```

```
int resposta_usuario = JOptionPane.showConfirmDialog(null, "Tem certeza que deseja APAGAR este Aluno ?");
```

```
if (resposta_usuario == 0) {// clicou em SIM
```

if (this.objaluno.DeleteAlunoBD(id)) {

```
// limpa os campos
this.c_nome.setText("");
this.c_idade.setText("");
this.c_curso.setText("");
this.c_fase.setText("");
JOptionPane.showMessageDialog(rootPane, "Aluno Apagado com Sucesso!");
```



System.out.println(this.objaluno.getMinhaLista().toString());

```
} catch (Mensagens erro) {
    JOptionPane.showMessageDialog(null,
erro.getMessage());
    finally {
        // atualiza a tabela.
        carregaTabela();
    }
```

#### Teste

- Cadastre um aLuno
- E depois apague

# Agora o Botão Cancelar



Insira o código:

```
private void b_cancelarActionPerformed(java.awt.event.ActionEvent evt) {
    this.setVisible(false);
  }
```

# Agora o Menu Sair Volte na TelaPrincipal

Arquivo	Sobre				🗌 🖃 — Check Box
N	lovo Aluno shortcut				Combo Box
G	air chortout				t× Text Area
	Edit Text				Progress Bar
	Change Variable Name				
	Events	►	Action	►	actionDerformed [iMenuItem3ActionDerformed]
	Align	►	Ancestor	►	

• Adicione o código

private void jMenuItem3ActionPerformed(java.awt.event.ActionEvent evt) {
 System.exit(0);

# Faça o Exercício passado na aula passada

- Primeiro, execute o guia passo a passo apresentado neste documento.
- Depois tente implementar as interfaces de ao menos 1 dos módulos solicitados para a A3.