

Técnicas de Programação II

Aula 03 – GUI e Swing

Edirlei Soares de Lima <edirlei.lima@uniriotec.br>

GUI – Graphical User Interface

• A API Java fornece diversas classes destinadas a criação de interfaces gráficas.

Name								
First Name:					Last N	ame:		
Title:					Nickn	ame:		
Format:	ltem 1							•
	ltem 1							
	ltem 2							
	ltem 3							
E-mail	ltem 4							
E-mail Add	ress:							Add
ltem 1								Edit
ltem 2								
Item 3								Remove
Item 4								As Default
Mail Forma	it: _ ○ Pla	ain Tex	at 🖲 Cu	stom				
_ · · · · ·								

AWT e Swing

• AWT (Abstract Window Toolkit):

- API padrão para criação de componentes GUI no início da plataforma Java (entre 1995 até 1998);
- Os objetos AWT são construídos sobre objetos de código nativo do Sistema Operacional em uso;
- Os componentes GUI originais do pacote java.awt estão diretamente associados com as capacidades de GUI da plataforma local.

• Swing:

- API escrita puramente em Java (padrão desde 1998);
- JLabel, JButton, JTextField, etc, são componentes GUI do pacote javax.swing.
- Mais flexível que o java.awt porque é implementada toda em Java, enquanto que java.awt é implementada em código nativo.

AWT e Swing



Containers e Componentes

- Uma interface gráfica em Java é baseada em dois elementos:
 - **Containers**: servem para agrupar e exibir outros componentes
 - Componentes: botões, labels, scrollbars, etc.
- Todo programa Java que ofereça uma interface possuí pelo menos um container.
- Uma janela de nível mais alto (que não fica contida dentro de outra janela) é um Frame ou, na versão Swing, um <u>JFrame</u>;
- O JFrame é um container. Isso significa que ele pode conter outros componentes de interface com o usuário.

Netbeans GUI Builder

File Edit View Navigate Source Refactor Rur	n Debug Profile Team Tools Window Help	🔍 - Search (Ctrl	+I)					
110.7/223.5MB 🟵 👔								
Projects × Files Services	📄 Antenna.java ×	Palette ×	-					
GUIFormExamples	Source Design History	Swing Containers	A					
🛱 🔂 Source Packages		Panel 📋	Tabbed Pane					
	♀ Use the Connection Mode button (in the toolbar) to establish a connection between components. ×	Split Pane	Scroll Pane					
		💷 Tool Bar 🛛 🚍	Desktop Pane					
ContactEditor.java	r Position/Direction	🔚 Internal Frame	Layered Pane					
ContactEditor.properties	Direction [9]: 140.000	Swing Controls						
Find.java		label Label	Button					
NewJFrame.java	Height [m]: 110.000	ON Toggle Button	Check Box					
			Button Group					
	System	💽 Combo Box	List					
jLabel2 [JLabel] - Navigator ×	Channels: 2 Watts: 12.000 Adjust	Tavt Field tx	Tevt Area					
🖻 Form Antenna		jLabel2 [JLabel] - Propertie	s X 🖃					
United Components	Antenna Type: Kathrein 742151	Properties	Binding					
irramej	Electrical Downtilt From [°]: 0.000 To: 10.000 Adjust	Events	Code					
·····iabei jLabel1 [JLabel]	Polarization: X +45°	Properties						
·····label jLabel2 [JLabel]		background	[212,208,200]					
JTextField1 [JTextField]	Frequency From [MHz]: 1943.000 To: 1951.000 Adjust	displayedMnemonic						
JTextField2 [JTextField]		font Tah	ioma 11 Plain					
jCheckBox1 [JCheckBox]	OK Cancel	foreground	[0,0,0]					
jPanel2 [JPanel]		horizontalAlignment LEA	DING The second					
JButton3 [JButton]		Icon IshalFar						
JBUTTON4 [JBUTTON]		tavt Hei	oht [m]:					
		toolTinText	grie (m):					
		verticalAlignment CEN	VTER VIER					
		Other Properties	V					

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



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



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":



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

🕥 HelloGUI - Net	Beans IDE 8.0								
File Edit View N	lavigate Source Refactor	Run Deb	ug P	rofile Team Tools Window Help)				
1	₽ 9 0 <def< th=""><th>ault config></th><th></th><th>- 7 👸 🕨 - 🚯 - (</th><th></th><th></th><th></th><th></th><th></th></def<>	ault config>		- 7 👸 🕨 - 🚯 - (
Projects & Files	Services	Start Pag		8 Folder JFrame Form Java Class Java Package Jova Interface JPanel Form Entity Class)	New Java Pa Steps 1. Choose File 2. Name and	ckage Type d Location	Name and Leo Package <u>N</u> ame:	Notion HelloInterface
	Debug Profile Test Set Configuration Open Required Projects	Alt+F6		Entity Classes from Database Other Teste				Project: Location: Created Folder:	HelloGUI Source Packages C:\Users\Edirlei\Documents\NetBeansProjects\HelloGUI\src\HelloInterface
Teste - Navig	Rename			ORACLE					
									< Back Next > Finish Cancel Help

5) Crie um novo "JFrame Form":

🕥 HelloGUI - NetBeans II	DE 8.0						
File Edit View Navigate	Source Refactor R	un Debug P	Profile Team Tools Window Help				
1 🔁 🔁 🕒 🛙	🤊 🦪 🛛 🖂	t config>	🗔 🕆 🦉 🕨 - 🎼 - 🕀	ş -			
Projects 22 Files Ser Source Packager 	s New	Start Page 8	Se Folder				
	Find Cut Copy Paste	Ctrl+F Ctrl+X Ctrl+C Ctrl+V	Java Package JFrame Form Java Class Java Interface JPanel Form		New JFrame Form Steps Choose File Type Name and Location	Name and L Class <u>N</u> ame:	HelloJFrame
	Delete Refactor	Delete 🛃	Image: Second			Project: Location:	HelloGUI Source Packages
	Compile Package Test Package History	F9 Ctrl+F6	Other			Package: <u>C</u> reated File:	HelloInterface Jsers\Edirlei\Documents\WetBeansProjects\HelloGUI\src\HelloInterface\HelloJFrame.java
Teste - Navigator 🕷	Tools	•	ORACLE				
							<back next=""> Finish Cancel Help</back>

Netbeans GUI Builder

HelloGUI - NetBeans IDE 8.0						x
<u>File Edit View Navigate Source Refactor</u>	<u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp			Q Search ((Ctrl+I)	
: 🔁 🚰 😫 🖣 : 🍤 🏈 : 🗠	ault config> 🖃 🍟 陵 🔹 🌆 🔹 🕕 🔹					
Projects % Files Services	Start Page 🛛 🔂 Hello JFrame. java 🖇		Palette %			-
🖃 🍥 🎒 HelloGUI	Source Design History	\$ B	Swing Contain	ers		-
Source Packages			Panel	Tabbed Pane	Split Pane	
	Y The Preview Design button (in the toolbar) enables you to test the design of the	he form. x	Scroll Pane	💷 Tool Bar	📇 Desktop Pane	-
		_	Internal Frame	👅 Layered Pane		=
			Swing Controls	5		
			label Label	OK Button	IN Toggle Button	
			Check Box	Radio Button	8 Button Group	
			💽 Combo Box	E List	Text Field	
			t× Text Area	I Scroll Bar	💭 Slider	
			Des este Des	Concentration of Circle	d C Deserved Cold	*
			[JFrame] - Proper	rties %		
			Properties Bind	ing Events	Code	
			Properties	etion EVIT Of		-6
[JFrame] - Navigator 🕺 📃			title		I_CLUGL ↓ []	- I
🖻 Form HelloJFrame			Other Properties		6	
🕀 🖓 Other Components			alwaysOnTop			
[⊡[JFrame]			alwaysOnTopSuppo	orted 🗸		-
			[JFrame]			•
	Output - Teste (run) 🕺					
						-
	975 1910					
						-
					1:1	INS

Componentes Básicos - Label

• Componente para exibição de texto não-editável ou ícones.

HelloGUI - NetBeans IDE 8.0					x			
<u>File E</u> dit <u>V</u> iew <u>N</u> avigate <u>S</u> ource F	Ref <u>actor Run D</u> ebug Profile Tea <u>m</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp		Q- Search	ı (Ctrl+I)				
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1								
8 Files Servi 🖃	Start Page 🛛 🔂 HelloJFrame.java 🖇 🖉		Palette %					
⊟… 🧽 HelloGUI	Source Design History	·	Swing Containers		-			
🖨 🚡 Source Packages			Panel	Tabbed Pane				
🖮 📑 HelloInterface	$\displaystyle \bigcirc$ To select multiple components in an area hold Shift and drag mouse over the comp	ponen	Split Pane	Scroll Pane				
HelloJFrame.java			💷 Tool Bar	📇 Desktop Pane	=			
			internal Frame	Layered Pane				
			Swing Controls					
		\subseteq	label Label	OK Button				
			In Toggle Button	Image: Image				
			◎- Radio Button	[®] [−] Button Group				
			Combo Box	List				
	jLabel1		Text Field	tx Text Area				
			I Scroll Bar	💭 Slider	-			
iLabel1 [JLabel] - Nay 🕷 🔳			jLabel1 [JLabel] - Pro	operties %				
Form Hello JFrame			Properties Binding	Events Code				
🗄 🖓 Other Components			Properties		*			

Componentes Básicos - Label

 Containers e componentes e estrutura da interface gráfica;

 Todos os elementos que fazem parte da interface gráfica são **objetos**;

 Todos os objetos possuem um <u>nome</u> (variable name) que pode (e deve!) ser alterado.





Componentes Básicos - Label

- Principais Propriedades (JLable):
 - text;
 - foreground;
 - background;
 - font;
 - icon;
 - toolTipText;
 - border;

jLabel1 [JLabel] - Pro	perties %	-
Properties Binding	Events	Code
Properties		*
background	[240,240,240]	
displayedMnemonic		
font	Tahoma 11 Plain	🗉
foreground	[0,0,0]	
horizontalAlignment	LEADING	▼
icon		▼
labelFor	<none></none>	▼
text	jLabel 1	
toolTipText		
verticalAlignment	CENTER	▼
Other Properties		
UIClassID	LabelUI	
alignmentX	0.0	
alignmentY	0.5	
autoscrolls		
	CONTRO OFFICET	
jLabel1 [JLabel]		•

Componentes Básicos – TextField

• Componente para entrada, edição e exibição de texto.

HelloGUI - NetBeans IDE 8.0								
<u>F</u> ile <u>E</u> dit <u>V</u> iew <u>N</u> avigate <u>S</u> ource f	ile <u>E</u> dit <u>V</u> iew <u>N</u> avigate <u>Source Refactor Run Debug Profile Team Tools Window H</u> elp							
1 1 1 2 4 5 6	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1							
Files Servi 🖃	Start Page 🗱 🔂 HelloJFrame.java 📾	Palette 8						
HelloGUI	Source Design History	Tool Bar	Desktop Pane					
HelloInterface	$^{ m Q}$ To change layout manager of a container use Set Layout submenu from its context menu.	Internal Frame	Layered Pane					
HelloJFrame.java		Swing Controls						
Logoff.png		label Label	OK Button					
E: bibraries		In Toggle Button	Image: Image					
	Nones iTextField1		8 Button Group					
		Combo Box	List					
		🔲 Text Field	tx Text Area					
A 1 1		💷 Scroil Bar	💭 Slider					
		Progress Bar	Formatted Field					
		···· Password Field	🗢 Spinner 👻					
iTextField1 [JTextFiel 8		jTextField1 [JTextFiel	d] - Properties 🕷 📃					
🖻 Form HelloJFrame		Properties Binding	Events Code					
🖶 🔄 Other Components		Properties	*					
. [JFrame]		editable	V 🗆					
label lableNome [JLabel]		background	[255,255,255]					
JTextField1 [JTextField]		columns	0					

Componentes Básicos – TextField

- Principais Propriedades (JTextField):
 - text;
 - editable;
 - foreground;
 - background;
 - font;
 - toolTipText;
 - border;
 - enabled;

JTextField1 [JTextFiel	d] - Properties 🕷	
Properties Binding	Events Co	ode
Properties		
editable	V	
background	[255,255,255]	
columns	0	
document	<default></default>	
font	Tahoma 11 Plain	
foreground	[0,0,0]	
horizontalAlignment	LEADING 🗖	
text	jTextField1	
toolTipText		
Other Properties		
UI	<default></default>	•
UIClassID	TextFieldUI	
action	<none></none>	
alignmentX	0.5	
alignmentY	0.5	
autoscrolls	V	
baselineResizeBehavior	CENTER_OFFSET	-
border	[XPFillBorder]	
caret	<default></default>	
caretColor	[0,0,0]	
15 11	4.4	

Componentes Básicos – Button

• Componente que representa um botão.

🕥 HelloGUI - NetBeans IDE 8.0								
<u>File Edit View N</u> avigate <u>S</u> ource I	ile <u>E</u> dit <u>V</u> iew <u>N</u> avigate <u>Source Refactor Run Debug Profile Team</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp							
1 🔁 🔚 🖶 🏷 🦿	1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1							
8 Files Servi	Start Page 🛛 🔂 HelloJFrame.java 📽	Palette 🕷						
	Source Design History	Swing Containers						
🖨 🚹 Source Packages		Panel Tabbed Pane						
Hello IFrame java	Y To change layout manager of a container use set Layout submenu from its context menu.	Split Pane 🔜 Scroll Pane						
		💷 Tool Bar 🛛 📇 Desktop Pane						
🗄 🔓 Libraries		🛅 Internal Frame 🕒 Layered Pane						
-		Swing Controls						
	Nome:	label 🛛 🗰 Button						
		In Toggle Button In Check Box						
		Image: Image						
		💽 Combo Box 📑 List						
		T						
		jButton1 [JButton] - Properties 🕺 🔳						
iButton1 [IButton] N %		Properties Binding Events Code						
E Form Hello Frame		Properties						
🚊 🖓 Other Components		action <none> </none>						
jToolBar1 [JToolBar]		background [240,240,240]						
[JFrame]								
www.abei lableNome [JLabel]								

Componentes Básicos – Button

- Principais Propriedades (JButton):
 - text;
 - foreground;
 - background;
 - font;
 - icon;
 - toolTipText;
 - border;
 - enabled;

Events me> [240,240,240 noma 11 Plain [0,0,0] tton 1	Code	* II
ne> [240,240,240 ioma 11 Plain [0,0,0] tton1	▼] 	*
one> [240,240,240 noma 11 Plain [0,0,0] tton1	▼] ▼ 	=
[240,240,240 ioma 11 Plain [0,0,0] tton1] 	
tton 1	····	
[0,0,0] tton 1		
tton 1	▼ 	
tton 1		
tton 1		
tonUl		
tton1		
ITER_OFFSET	•	
		Ŧ
	v TER_OFFSET	V ITER_OFFSET

Componentes Básicos – List

• Componente que exibe uma lista de itens e permite que o usuário possa seleciona-los.

HelloGUI - NetBeans IDE 8.0				
<u>File Edit View Navigate Source Re</u>	ef <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m T</u> ools <u>W</u> indow <u>H</u> elp		Q- Search	(Ctrl+I)
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	(default config>) T T T T T T T T T T T T T T T T T T T			
8 Files Servi 🖃	Start Page 🛛 🔂 HelloJFrame.java 🖇		Palette %	
E B	Source Design History 💦 🖶 😭 🗮 🚔 🚛 🌆 🖶	⇔ \$ ⊞	💷 Tool Bar	🖻 Desktop Pane 👘
Source Packages	O The Navigator window displays a tree hierarchy of components in the operation of components in the operation of the second	ened form. x	Internal Frame	🖲 Layered Pane
HelloJFrame.java			Swing Controls	
Logoff.png	T		iabei Label	OK Button
🗄 🕞 Libraries			Toggle Button	Image: Image
		_	Image: Image	Butten Group
	Nome: Adicionar		🖃 Combo Box 🤇	List
	Item 1	\sim	Text Field	t× Text Area
	Item 2		Scroll Bar	💭 Slider 🔶
	Item 3		il ist1 [II ist] - Proper	ties %
	Item 5		Properties Binding	Events Code
		\sim	Properties	
jList1 [JList] - Navigator 🕺 📃			background	[255,255,255] =
🖻 Form HelloJFrame			border	(No Border)
Other Components			font	Tahoma 11 Plain 🛄
			foreground	[0,0,0]
taxtNama [ITaxtEid]			model	Item 1, Item 2, Ite
			selectionMode	MULTIPLE_INT 🔻 🛄

Componentes Básicos – List

- Principais Propriedades (JList):
 - model;
 - selectionMode;
 - selectedIndex;
 - visibleRowCount;
 - foreground;
 - background;
 - font;
 - toolTipText;
 - border;
 - enabled;

jList1 [JList] - Proper	ties 🕺	
Properties Binding	Events Co	de
Properties		
background	[255,255,255]	
border	(No Border)	=
font	Tahoma 11 Plain	
foreground	[0,0,0]	
model	Item 1, Item 2, Ite	•
selectionMode	MULTIPLE_INT 👻	
tooITipText		
Other Properties		
UIClassID	ListUl	
alignmentX	0.5	
alignmentY	0.5	
anchorSelectionIndex	-1	
autoscrolls	V	
baselineResizeBehavior	CONSTANT_ASCENT	•
cellRenderer	<default></default>	
		\cap
jList1 [JList]		?

Netbeans GUI Builder – Código Gerado



Eventos – Button

- Principais Eventos (JButton):
 - <u>actionPerformed;</u>
 - mouseClicked;
 - mousePressed;
 - mouseRelesead;
 - mouseMoved;
 - mouseEntered
 - mouseExited;
 - focusGained;
 - focusLost;

buttonAdd [JButton] -	Properties 🕷		-
Properties Binding	Events	Code	
focusGained	<none></none>	▼	-
focusLost	<none></none>		
hierarchyChanged	<none></none>		
inputMethodTextChanged	<none></none>		
itemStateChanged	<none></none>		
keyPressed	<none></none>		
keyReleased	<none></none>	- 🛨 📖	
keyTyped	<none></none>	- 🖵 🛄	
mouseClicked	<none></none>	- 🛨 📖	
mouseDragged	<none></none>	- 🛨 📖	
mouseEntered	<none></none>	- 🛨 📖	
mouseExited	<none></none>	- 🛨 📖	
mouseMoved	<none></none>	- 🛨 📖	=
mousePressed	<none></none>	- 🖵 🛄	
mouseReleased	<none></none>		
mouseWheelMoved	<none></none>		
propertyChange	<none></none>		
stateChanged	<none></none>	▼	-
buttonAdd [JButton]		(?

• Criando um evento de ativação para o botão (actionPerformed):

buttonAdd [JButton]	Properties 8	-	HelloGUI - NetBeans IDE 8.0		
Properties Binding	Events	Code	<u>File Edit View Navigate Source</u>	e Ref <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Team <u>I</u> ools <u>W</u> indow <u>H</u> elp Q Search (Ctrl+I)	
Events		*	🕆 🞦 🔛 🖣 🏓 🤇	📲 🛛 <default config=""> 🕞 🍟 խ 🔹 🌇 🍷 🕕 🔹</default>	
actionPerformed	<none></none>	•	8 Files Servi 🖃	Start Page 🗱 🔂 HelloJFrame.java 🕷	
ancestorAdded	buttonAddAction	Perf	HelloGUI	Source Design History 🔯 👼 + 🐺 + 💐 🖓 🖓 🖓 🖶 📪 🖗 😓 🖄 😂 🖄 🔴 🔛 🖉 📑	
ancestorMoved	<none></none>	▼	Source Packages	29 🖓 /**	Т
ancestorMoved	<none></none>	▼	HelloJFrame.java	a 30 * This method is called from within the constructor to initialize the form.	
ancestorRemoved	<none></none>	-	🗄 🖓 🍃 🔓	31 * WARNING: Do NOT modify this code. The content of this method is always 32 * regenerated by the Form Editor.	
ancestorResized	<none></none>	T		33 4 */	
caretPositionChanged	<none></none>	-		34 @SuppressWarnings("unchecked") 35 H Generated Code	
componentAdded	<none></none>	T		117	T
componentHidden	<none></none>	-		118 - private void buttonAddActionPerformed(java.awt.event.ActionEvent evt) {	
amponentMoved				119 // TODO add your handling code here:	
componentwoved		- 0		121	
componentRemoved	<none></none>				
componentResized	<none></none>	•	Members V <e th="" v<=""><th>123 * Gparam args the command line arguments</th><th></th></e>	123 * Gparam args the command line arguments	
componentShown	<none></none>	•	HelloJFrame :: JFrame	125 - public static void main(String args[]) {	
ocusGained	<none></none>	▼	HelloJFrame()	126 /* Set the Nimbus look and feel */	
ocusLost	<none></none>	▾	buttonAddActionPerformer	148	
ierarchyChanged	<none></none>	▼	main(String] args)	149 /* Create and display the form */	
nputMethodTextChanged	<none></none>	▼	buttonAdd : JButton	java.awt.EventQueue.invokeLater(new Runnable() {	
temStateChanged	<none></none>	T	iRadioButton1: JRadioBut	152 new HelloJFrame().setVisible(true);	
	I		jRadioButton2 : JRadioBut	153 - }	
ctionPerformed		?			
java.awt.event.ActionLi	stener			Hello JFrame > button AddAction Performed >	P
			ත 🔁 Output	119:1	

• **Exemplo 1 –** Mostrar mensagem com o conteúdo do TextField:

private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
{
 JOptionPane.showMessageDialog(this, "Hello " + textNome.getText());
}



- Exemplo 2 Adicionar o conteúdo do TextField na List:
 - a) Crie um objeto do tipo DefaultListModel para armazenar os elementos da lista:

```
...
public class HelloJFrame extends javax.swing.JFrame {
    private DefaultListModel listModel = new DefaultListModel();
...
```

b) No método construtor da classe, associe o listModel a lista:

```
...
public HelloJFrame() {
    initComponents();
    listNomes.setModel(listModel);
}
```

• Exemplo 2 – Adicionar o conteúdo do TextField na List:

. . .

c) No evento actionPerformed, adicione o conteúdo do TextField na List:

```
private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
{
    listModel.addElement(textNome.getText());
    textNome.setText("");
}
....
```

Componentes Básicos – RadioButton e ButtonGroup

• Componentes que permitem a seleção de opções.

🕥 HelloGUI - NetBeans IDE 8.0			
<u>File E</u> dit <u>V</u> iew <u>N</u> avigate <u>S</u> ource F	Ref <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp	Q- Search	(Ctrl+I)
12 12 12 13 10 10 10 10 10 10 10 10 10 10 10 10 10	<default config=""> 😱 🏠 🔖 🖌 👘 🗸</default>		
8 Files Servi 🖃	Start Page 🛛 🔂 HelloJFrame.java 🕺	Palette 8	
🖃 🍥 🧽 HelloGUI	Source Design History 🛄 🖶 🖺 🗎 🛄 井 🕅 🗰 😂 🔅 🖼	Swing Containers	<u>^</u>
🖃 🖟 Source Packages		Panel	Tabbed Pane
	$lash$ Use the ButtonGroup component and the buttonGroup property to create a group of exclusion $rac{1}{2}$	^{sir}][Split Pane	📑 Scroll Pane
HelloJFrame.java		Tool Bar	🖻 Desktop Pane 🗧
		internal Frame	Layered Pane
		Swing Controls	
	Nome: Adicionar	label Label	OK Button
	jRadioButton1	ON Toggle Button	- Check Box
	jRadioButton2	◎— Radio Button	[®] ⊟ Button Group
		Combo Box	E List
		Text Field	tx Text Area 👻
		jRadioButton1 [JRadi	oButton] - Pro 🕺 🔳
jRadioButton1 [JRadi 🕺 🖃		Properties Binding	Events Code
Form HelloJFrame		Properties	A
Other Components		action	<none> 💌 🛄 🗐</none>
buttonGroup1 [ButtonGrou		background	[240,240,240]
🖆 📲 [JFrame]		model	<default> 💌 🛄</default>
label lableNome [JLabel]		buttonGroup	<none> 💌 🛄</none>

Componentes Básicos – RadioButton

- Principais Propriedades (JRadioButton):
 - text;
 - buttonGroup;
 - Selected;
 - foreground;
 - background;
 - font;
 - icon;
 - toolTipText;
 - border;
 - enabled;

jRadioButton1 [JRadio	Button] - Pro	88	
Properties Binding	Events	Code	
Properties			4
action	<none></none>	▾ 📖	Γ
background	[240,240,240]		
model	<default></default>		1
buttonGroup	<none></none>		
font	Tahoma 11 Plain		L
foreground	[0,0,0]		
mnemonic			
selected			
text	jRadioButton1		
toolTipText			
Other Properties			
UIClassID	RadioButtonUl		
actionCommand	jRadioButton1		
alignmentX	0.0		
alignmentY	0.5		
autoscrolls			
baselineResizeBehavior	CENTER_OFFSET	•	
border	[CompoundBorder	UIF	
hand an Datate d		\square	

Usando o RadioButton

 Exemplo 3 – Adicionar o conteúdo do TextField em duas List de acordo com a opção selecionada no RadioButton.

– Interface:

Nome:		Adicionar	
	O Lista 1	🔘 Lista 2	

Usando o RadioButton

 Exemplo 3 – Adicionar o conteúdo do TextField nas Lists de acordo com a opção selecionada no RadioButton.

. . .

```
private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
ł
  if (radioBt1.isSelected())
    listModel1.addElement(textNome.getText());
  else if (radioBt2.isSelected())
    listModel2.addElement(textNome.getText());
                                                                             <u>≰</u>,
  textNome.setText("");
                                                           Nome: Mario
                                                                           Adicionar
                                                              Lista 1
                                                                       Lista 2
                                                              Edirlei
                                                                       Pedro
                                                              Maria
```

Componentes Básicos – CheckBox

 Componente que permite a seleção de opções (marcado ou não marcado).

HelloGUI - NetBeans IDE 8.0				
<u>File E</u> dit <u>V</u> iew <u>N</u> avigate <u>S</u> ource F	Ref <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp		Q- Search	(Ctrl+I)
1 🔁 🚰 🖶 🦻 🦿	<default config=""></default>			
8 Files Servi	Start Page 🛛 🔂 HelloJFrame.java 🖇		Palette %	
🖃 🎂 HelloGUI	Source Design History	>\$} ⊞	Split Pane	Scroll Pane 🔺
🖨 🍓 Source Packages			💷 Tool Bar	🖶 Desktop Pane
HelloInterface	Y To select multiple components in an area hold shift and drag mouse over the	components. :	Internal Frame	🖲 Layered Pane
Libraries	•		Swing Controls	E
			label Label	OK Button
	Nome		IN Toggle Button	☑— Check Box
			Radio Button	S- Button Group
			💽 Combo Box	Eist
	Lista 1 C Lista 2		Text Field	tx Text Area
			I Scroll Bar	🗘 Slider
			JUNECKBOX1 [JUNECKE	soxj - Properti »
jCheckBox1 [JCheck 🕺 🖃			Properties binding	Evenus Code
buttonGroup1 [Button			actionPerformed	<none></none>
in [JFrame]			ancestorAdded	<none></none>
label lableNome [JLabel]			ancestorMoved	<none></none>
textNome [JTextField]			ancestorMoved	<none> 💌 🛄</none>
I DK buttonAdd [JButton]		I	ancestorDemoved	<none></none>

Componentes Básicos – CheckBox

- Principais Propriedades (JCheckBox):
 - text;
 - buttonGroup;
 - Selected;
 - foreground;
 - background;
 - font;
 - icon;
 - toolTipText;
 - border;
 - enabled;

jCheckBox1 [JCheckB	Box] - Properti	88		_
Properties Binding	Events	Cod	le	
Properties				
action	<none></none>	•		
background	[240,240,240]			=
buttonGroup	<none></none>	•		
font	Tahoma 11 Plain			L
foreground	[0,0,0]			
model	<default></default>	•		
mnemonic				
selected				
text	jCheckBox1			
toolTipText				
Other Properties				
UIClassID	CheckBoxUI			
actionCommand	jCheckBox1			
alignmentX	0.0			
alignmentY	0.5			
autoscrolls				
	CENTED OFFICET		_	
jCheckBox1 [JCheckB	lox]		(?

Usando o CheckBox

- Exemplo 4 Permitir ao usuário escolher se ele deseja limpar automaticamente o conteúdo do TextField após adiciona-lo à List.
 - Interface:

lome:		Adicionar
	Limpar Automaticamer	ite
	O Lista 1	🔘 Lista 2

Usando o CheckBox

 Exemplo 4 – Permitir ao usuário escolher se ele deseja limpar automaticamente o conteúdo do TextField após adiciona-lo à List.

```
private void buttonAddActionPerformed(java.awt.event.ActionEvent evt)
  if (radioBt1.isSelected())
    listModel1.addElement(textNome.getText());
  else if (radioBt2.isSelected())
                                                                              _ D X
    listModel2.addElement(textNome.getText());
                                                            Nome: Mario
                                                                            Adicionar
  if (checkBoxLimpar.isSelected())
                                                               Limpar Automaticamente
                                                               C Lista 1
                                                                        Lista 2
    textNome.setText("");
                                                               Edirlei
                                                                        Pedro
                                                                        Maria
```

Componentes Básicos – MenuBar

• Componente que permite a criação de uma barra de menu.

🕞 HelloGUI - NetBeans IDE 8.0					
ile <u>E</u> dit <u>V</u> iew <u>N</u> avigate <u>S</u> ource Ref <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Team <u>T</u> ools <u>W</u> indow <u>H</u> elp					
👚 🔚 🛃 🆻 🦿 🔄 <default config=""> 🕞 🚏 🦉 🕨 - 🎼 - 🛞 -</default>					
🛛 🗱 Files Servi 🖃 Start Page 🗱 📑 HelloJFrame.java 🕿	Palette 🕷 🔳				
HelloGUI Source Design History History	Separator T Text Pane				
HelloInterface 💡 The Preview Design button (in the toolbar) enables you to test the design of the form. x	🖄 Editor Pane 🖸 Tree				
HelloJFrame.java	Table				
🗄 🔓 Libraries 🖉 File Edit	Swing Menus				
	File Menu Bar				
Nemo	E Menu				
	- Menu Item E				
Limpar Automaticamente	Menu Item / CheckBox				
ista 1 Dista 2	Menu Item / RadioButton				
	Ropup Menu				
	jMenuBar1 [JMenuBar] - Properties 🕺 🖃				
jMenuBar1 [JMenuBa 📽 🔲	Properties Binding Events Code				
E Form Hello JFrame	Properties				
Cher Components	background [240,240,240] =				
	border [MenuBarBorder]				
Urramej	foreground [0,0,0]				
	toolTipText				

Componentes Básicos – Menultem

Componente que permite a criação de itens para a barra de menu.

HelloGUI - NetBeans IDE 8.0						x
<u>File Edit View N</u> avigate <u>S</u> ource I	Ref <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u>	<u>T</u> ools <u>W</u> indow <u>H</u> elp		Q- Search	(Ctrl+I)	
1 🔁 🔚 🖶 🏷 🦿	<default config=""></default>	🥸 🕨 - 🌇 - 🕦 -				
% Files Servi S HelloGUI S Source Packages HelloInterface S HelloJFrame.java S Libraries	Start Page 8 HelloJFrame.java Source Design History 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	a 8 allable useful actions for the selected component Adicionar ente C Lista 2	ents. x	Palette % Separator Editor Pane Editor Pane Table Swing Menus Menu Bar Menu Menu Item Menu Item Menu Item Menu Item / Check Menu Item / Radiot	T Text Pane	• •
jMenuItem1 [JMenuIt %				jMenultem1 [JMenu Properties Binding Properties action accelerator background font	Item] - Prope 8 Events (<none> null [240,240,240] Segoe UI 12 Plain</none>	Code

• Exemplo 5 – Criando e abrindo uma nova janela.

a) Crie um novo JFrame;





- **Exemplo 5 –** Criando e abrindo uma nova janela.
 - b) Crie um <u>novo Package</u> e <u>arraste o arquivo de uma imagem</u> para o projeto;

HelloGUI - NetBeans IDE 8.0	A CONTRACT OF	4		×
<u>File Edit View Navigate Source</u>	Ref <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp	Q Searc	h (Ctrl+I)	
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	(default config>) T Y Y > T · · · · · · · · · · · · · · · · · ·			
Files Servi 🖃	Start Page 🛛 🔂 Hello JFrame, java 🖇 💼 About Panel, java 📽	□ Palette ×		-
HelloGUI	Source Design History	🗄 💷 Tool Bar	🚰 Desktop Pane	*
Source Packages	O To add a component multiple times, select it via cick in palette and then Shift-click on der	ian (🛅 Internal Frame	👅 Layered Pane	
AboutPanel.iava	g To add a component multiple times, select it via dick in palette and then shintedick on des	Swing Controls		
HelloJFrame.jave		label Label	DK Button	=
Resources		IN Toggle Button	Check Box	
Liberaire		Radio Button	8 Button Group	
		💽 Combo Box	Eist List	
		Text Field	t× Text Area	
		Scroll Bar	ѿ Slider	
		Progress Bar	Formatted Field	
		Doonword Field	TA Coincor	Ŧ
		Resources - Prope	erties %	-
Logounirio.png - Navi 🕷 🖃		Name	Resources	
		Sorting		
		Sort Mode	By Name (Package	•
UNIRIO				
		-		
		Resources	(Deseursee)	
Dimensions: 297 x 264		Java Source Package	(resources)	
ති 🔁 Output				

- **Exemplo 5 –** Criando e abrindo uma nova janela.
 - c) Adicione um novo <u>label</u> ao formulário, colocando a imagem previamente selecionada como icone. Adicione também um <u>botão</u> para fechar a janela.



- Exemplo 5 Criando e abrindo uma nova janela.
 - d) Na janela principal, crie um novo evento para criar e abrir a nova janela quando o usuário clicar no item "About" do menu "Help":

```
private void menuAboutActionPerformed(java.awt.event.ActionEvent evt)
{
    AboutJFrame aboutJFrame = new AboutJFrame();
    aboutJFrame.setVisible(true);
}
```

- Exemplo 5 Criando e abrindo uma nova janela.
 - e) Crie um evento "actionPerformed" para o botão fechar, ocultando e destruindo a janela quando o usuário clicar no botão:

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

Fechar

Componentes Básicos – Frame

- Principais Propriedades (JFrame):
 - defaultCloseOperation;
 - title;
 - background;
 - awaysOnTop;
 - iconImage;
 - resizable;
 - undecorated;
 - type;

Properties Binding	Events	Code
- Properties		
defaultCloseOperatio	EXIT_ON_CLOSE	
title		
Other Properties		
alwaysOnTop		
alwaysOnTopSupported	\checkmark	
autoRequestFocus	V	
background	[240,240,240]	
bounds	<not set=""></not>	
cursor	Default Cursor	
enabled	V	
extendedState	0	
focusCycleRoot	V	
focusTraversalPolicy	<default></default>	
focusTraversalPolicyPro		
focusable		
focusableWindowState		
font	null	
foreground	null	
	<none></none>	-

Componentes Básicos – Tabbed Pane

• Componente que permite a criação de telas com abas.

HelloGUI - NetBeans IDE 8.0	1	
<u>File Edit View Navigate Source Refactor Run D</u> ebug Profile Tea <u>m</u> Tools <u>W</u> indow <u>H</u> elp	Q Search	(Ctrl+I)
👚 🔚 🖶 🌗 🌔 🤇 🔄 🕞 🕞 🕞 🕞 🕞		
🗱 Files Servi 🖬 Start Page 🛚 📄 HelloJFrame.java 🕿 🔂 AboutJFrame.java 🕿	Palette %	
HelloGUI Source Design History History	Swing containers Panel	Tabbed Pane
in the location in the second component/container to create the second tab.	Split Pane	Scroll Pane 🗉
AboutJFrame.java	💷 Tool Bar	📇 Desktop Pane
tab1	Tinternal Frame	🖲 Layered Pane
LogoUnirio.png	Swing Controls	
🗄 - 🔓 Libraries	label Label	OK Button
	IN Toggle Button	☑— Check Box
	®- Radio Button	[®] [−] Button Group
	Combo Box	Eist
	Text Field	tx Text Area 👻
	jTabbedPane2 [JTab	bedPane] - Pr 🕺 📃
jTabbedPane2 [JTabb 🕺 🔲	Properties Binding	Events Code
🖻 Form HelloJFrame	Properties	*
Dther Components	background	[240,240,240] =
IFrame]	border	(No Border)
⊡ jTabbedPane2 [JTabbedPa	foreground	[0,0,0]
· IPanel2 [JPanel]	selectedIndex	0

Componentes Básicos – Tabbed Pane

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help C Search (CtH+1) C Search (CtH+2) C Search (CtH+2) Files Servi	🕥 HelloGUI - NetBeans IDE 8.0						
Image: Image	<u>File Edit View N</u> avigate Source F	ef <u>a</u> ctor <u>R</u> un <u>D</u> ebug <u>P</u> rofile Tea <u>m</u> <u>T</u> ools <u>W</u> indow <u>H</u> elp	Q- Searc	:h (Ctrl+I)			
* Files Servi * HelloUI • Source Packages • HelloUI • Source Packages • HelloUI • HelloTrame.java • HelloTrame.java • HelloTrame.java • Besources • LogoUnirio.prog • Libraries Principal Opcões Ajuda Principal Opcõ	👚 🚰 🚂 🦻 🍘 Kalefault config> 🐨 🍞 🎇 🕨 - 🌇 - 🔞 -						
Image: Control of the components Image: Control of the control	Files Servi HeloGUI Source Packages HeloThrerface MootTFrame.java Resources LogoUlniro.png Libraries	Start Page Image: Hello JFrame.java [jTabbedPane2] Image: About JF Source Design History Image: Base of the table of the table of the table of table	Palette # B Swing Containe Panel Split Pane Tool Bar Internal Frame Swing Controls How Label Of Toggle Button e Radio Button I Combo Box	Tabbed Pane			
Image: Screener Lipscreament Lipscreamen			JTabbedPane2 [JT. Properties] Bindi Properties] Bindi Properties background border foreground selectedIndex tabLayoutPolicy tabPlacement toOTIpText Other Properties JTabbedPane2 [JTab	Image: Prime Image: Prima Image: Prime Image: Prime<			

<u></u>			
Principal	Opções	Ajuda	
Nome:			Adicionar
🗌 I	Limpar Auto	omaticar	nente
01	Lista 1		🔾 Lista 2

Exercícios

Lista de Exercícios 03 – GUI e Swing

http://uniriodb2.uniriotec.br