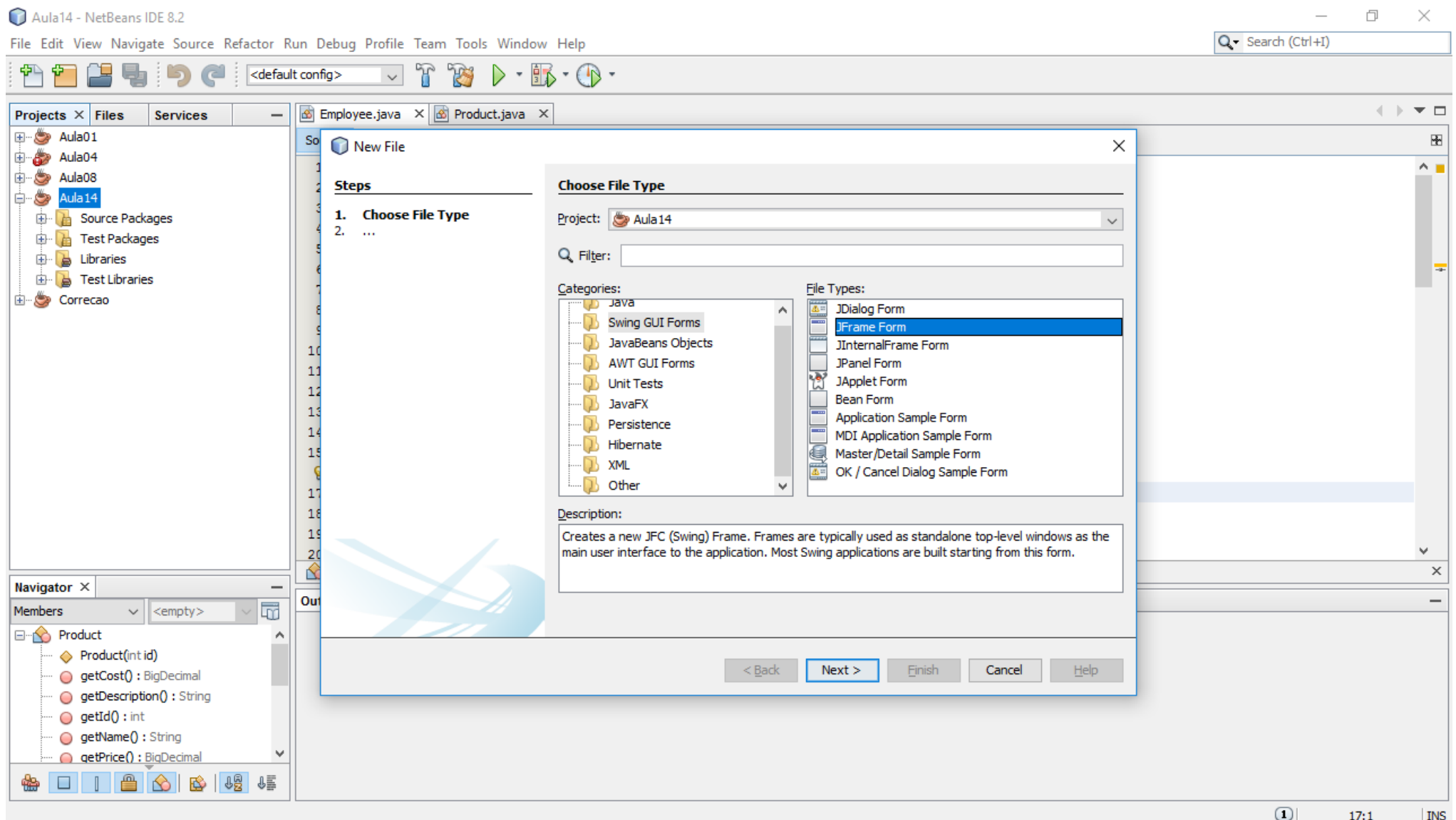


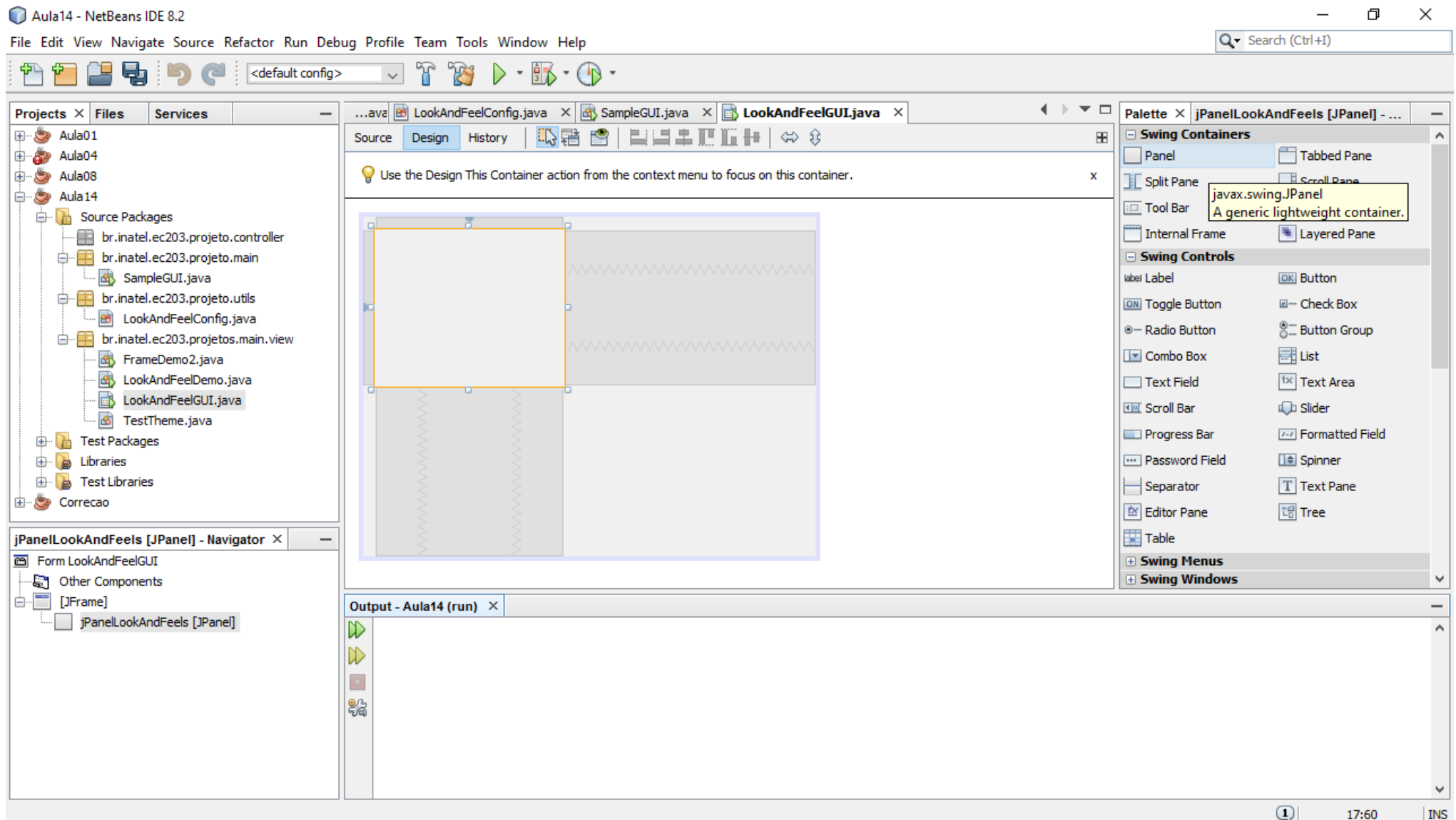
# Interface Gráfica do Usuário (GUI) em Java

## Capítulo XIII

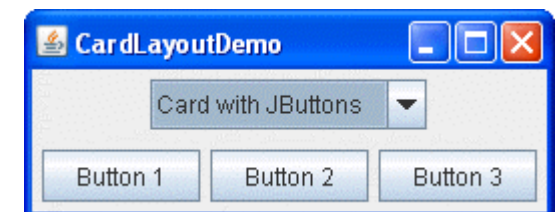
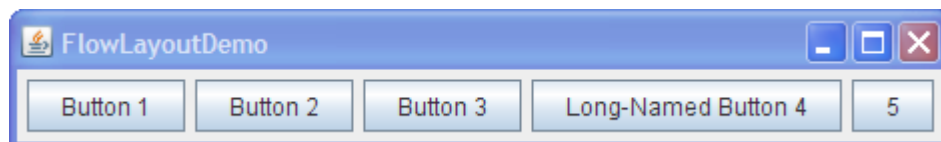
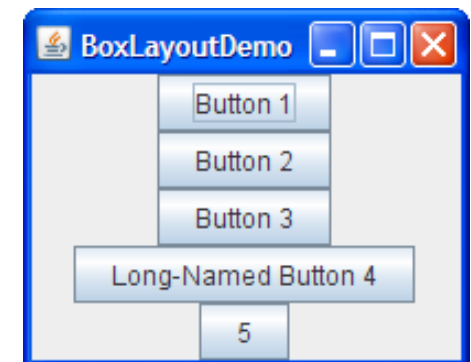
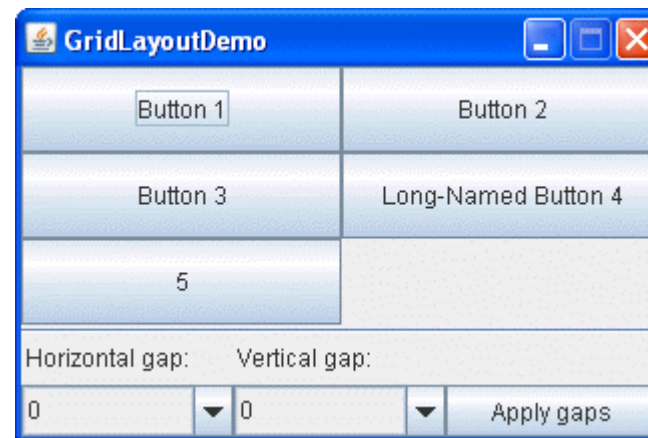
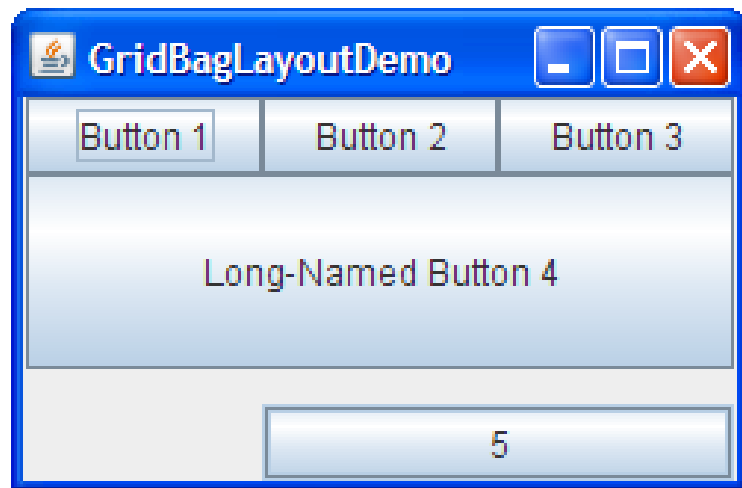
### Janela Principal usando JFrame



## Adicionando o painel (container)



## Layout Manager



## Adicionando o ButtonGroup

Aula14 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Files Services

Source Packages

- br.inatel.ec203.projeto.controller
- br.inatel.ec203.projeto.main
- SampleGUI.java
- br.inatel.ec203.projeto.utils
- LookAndFeelConfig.java
- br.inatel.ec203.projetos.main.view
- FrameDemo2.java
- LookAndFeelDemo.java
- LookAndFeelGUI.java
- TestTheme.java

Test Packages

Libraries

Test Libraries

Correcao

buttonGroup1 [ButtonGroup] - Navigator

- Form LookAndFeelGUI
- Other Components
- buttonGroupLookAndFeel [ButtonGroup]
- [JFrame]
- JPanelLookAndFeels [JPanel]

LookAndFeelConfig.java

SampleGUI.java

LookAndFeelGUI.java

Design

The Tools>Palette>Swing/AWT Components menu item allows you to modify the content of the Palette.

Palette

Swing Containers

- Panel
- Split Pane
- Tool Bar
- Internal Frame
- Tabbed Pane
- Scroll Pane
- Desktop Pane
- Layered Pane

Swing Controls

- Label
- Toggle Button
- Radio Button
- Button
- Check Box
- Button Group

Cor

Text

Scroll Bar

Progress Bar

Password Field

Separator

Editor Pane

Table

Slider

Formatted Field

Spinner

Text Pane

Tree

Swing Menus

Swing Windows

Output - Aula14 (run)

17:60

INS

# Adicionando o JRadioButton

Aula14 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Files Services

...ava LookAndFeelConfig.java SampleGUI.java LookAndFeelGUI.java

Source Design History

Use the ButtonGroup component and the buttonGroup property to create a group of exclusive JRadioButtons.

Default

Palette jRadioButtonLookAndFeelDefau...

Swing Containers

- Panel
- Split Pane
- Tool Bar
- Internal Frame
- Tabbed Pane
- Scroll Pane
- Desktop Pane
- Layered Pane

Swing Controls

- Label
- Toggle Button
- Radio Button
- ComboBox
- Text Field
- ScrollPane
- ProgressBar
- Password Field
- Separator
- Editor Pane
- Table
- OK Button
- Check Box
- Button Group
- Spinner
- Text Pane
- Tree

javax.swing.JRadioButton

An item that can be selected or deselected. Used with a ButtonGroup object to create a group of buttons in which only one button at a time can be selected.

jRadioButtonLookAndFeelDefault [JRadioButton]

Form LookAndFeelGUI

Other Components

- buttonGroupLookAndFeel [ButtonGroup]
- [JFrame]
- jPanelLookAndFeel [JPanel]
- jRadioButtonLookAndFeelDefault [JRadioButton]

Output - Aula14 (run)

17:60 INS



## Incluindo o JRadioButton no ButtonGroup

Aula14 - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Files Services

Aula01  
Aula04  
Aula08  
Aula14

Source Packages

- br.inatel.ec203.projeto.controller
- br.inatel.ec203.projeto.main
- SampleGUI.java
- br.inatel.ec203.projeto.utils
- LookAndFeelConfig.java
- br.inatel.ec203.projetos.main.view
- FrameDemo2.java
- LookAndFeelDemo.java
- LookAndFeelGUI.java
- TestTheme.java

Test Packages

Libraries

Test Libraries

Correcao

jRadioButtonLookAndFeelDefault [JRadioButton] - Properties

Properties Binding Events Code

Use the ButtonGroup

LookAndFeelGUI

Default

Property	Value
action	<none>
background	[240,240,240]
model	<default>
buttonGroup	<none>
font	<none>
foreground	buttonGroupLookAndFeel
mnemonic	
selected	
text	Default
toolTipText	
Other Properties	
UIClassID	RadioButtonUI
actionCommand	Default
alignmentX	0.0
alignmentY	0.5
autoscrolls	
baselineResizeBehavior	CENTER_OFFSET
border	[CompoundBorderUIResource]
borderPainted	
componentPopupMenu	<none>
contentAreaFilled	

buttonGroup

Group of buttons in which this button belongs

Close Help

Palette jRadioButtonLookAndFeelDefault...

Swing Containers

- Panel
- Split Pane
- Tool Bar
- Internal Frame
- Tabbed Pane
- Scroll Pane
- Desktop Pane
- Layered Pane

Swing Controls

- Label
- Toggle Button
- Radio Button
- Combo Box
- Text Field
- Scroll Bar
- Progress Bar
- Password Field
- Separator
- Editor Pane
- Table
- OK Button
- Check Box
- Button Group
- List
- Text Area
- Slider
- Formatted Field
- Spinner
- Text Pane
- Tree

Swing Menus

Swing Windows

Output - Aula14 (run)

Form LookAndFeelGUI

- Other Components
- buttonGroupLookAndFeel [ButtonGroup]
- [JFrame]
- JPanelLookAndFeel [JPanel]
- jRadioButtonLookAndFeelDefault [JRadioButt]

17:60 INS

## Adicionando a borda no painel

The screenshot shows the NetBeans IDE 8.2 interface. The 'Projects' pane on the left displays a project structure for 'Aula14'. The 'Files' pane shows the 'Source Packages' and 'Test Packages' folders. The 'Services' pane is empty. The 'Output - Aula14 (run)' pane at the bottom shows the execution output.

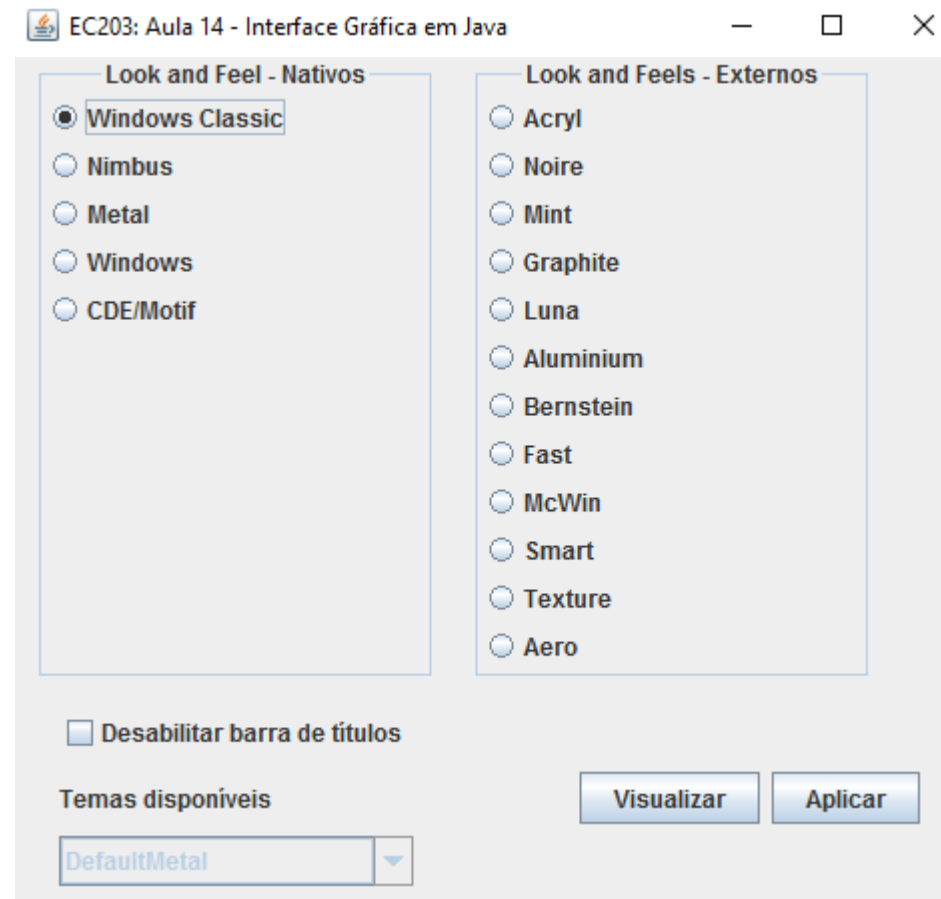
The 'jPanelLookAndFeels [JPanel] - Properties' dialog is open, showing the 'Properties' tab. The 'border' property is selected, and its value is set to '(javax.swing.border.Border) The component's border.' The 'Other Properties' section shows various properties like 'UIClassID', 'alignmentX', 'alignmentY', 'autoscrolls', 'baselineResizeBehavior', 'componentPopupMenu', 'cursor', 'debugGraphicsOptions', 'doubleBuffered', 'enabled', 'focusCycleRoot', 'focusTraversalPolicy', 'focusTraversalPolicyProvider', 'focusable', 'font', and 'inheritsPopupMenu'.

The 'jPanelLookAndFeels [JPanel] - Navigator' pane shows the 'Form LookAndFeelGUI' and 'Other Components' sections. The 'jPanelLookAndFeels [JPanel]' component is selected in the 'Other Components' section.

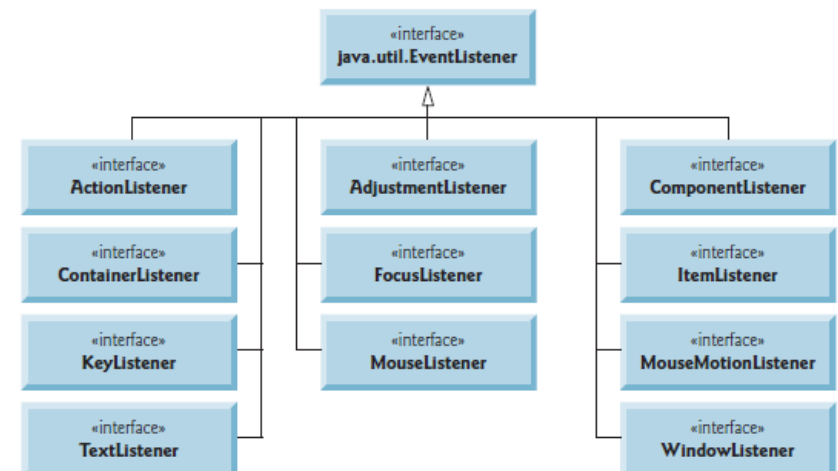
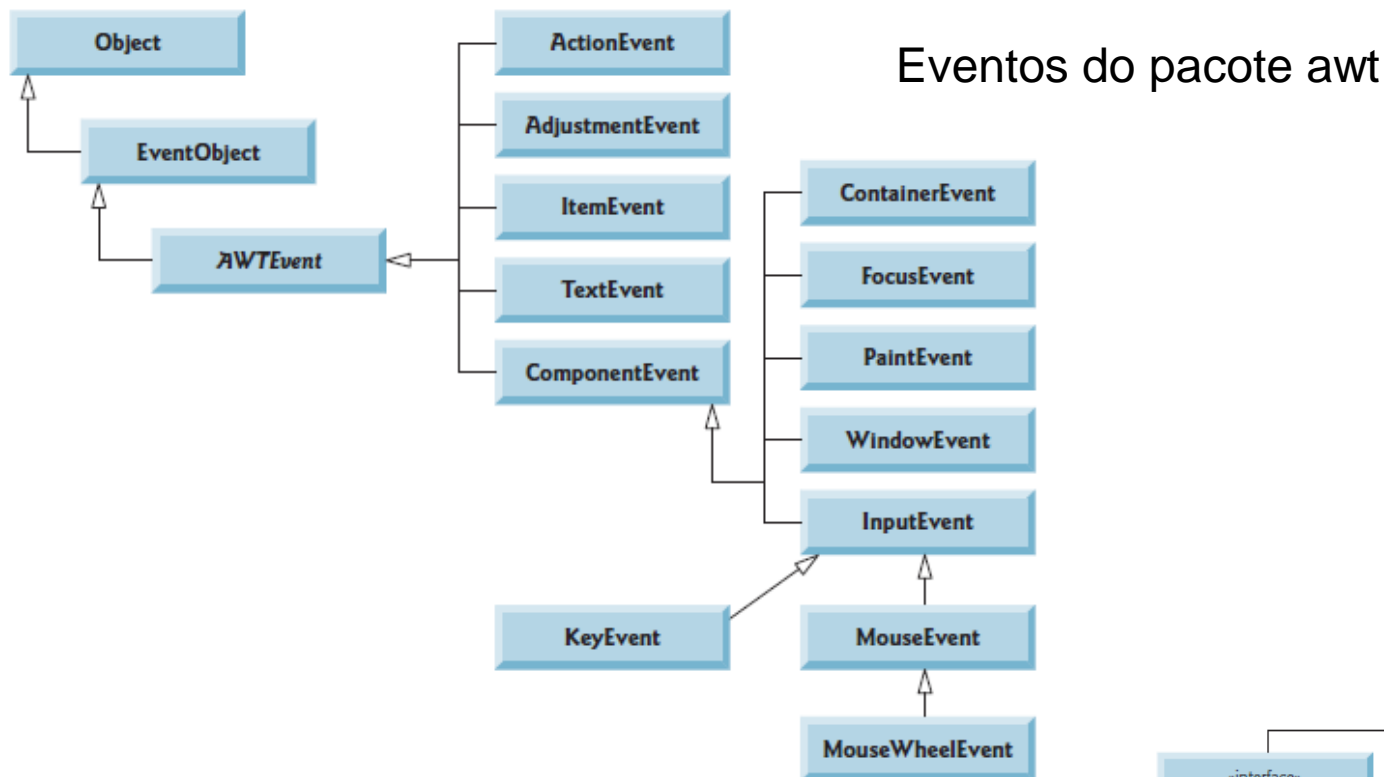
The 'Output - Aula14 (run)' pane shows the execution output, including the text 'The Tools>Palette' and 'The component's border.'



## Exemplo de interface



## Tratamento de eventos



Interfaces para tratamento de eventos do pacote awt

## Tratamento de eventos

```
public void setButtonApplyEvent(LookAndFeelApplyButtonHandler event) {
    jButtonApplication.addActionListener(event);
}

public void setButtonPreviewEvent(LookAndFeelPreviewButtonHandler event) {
    jButtonPreview.addActionListener(event);
}

public void setRadioButtonEvent(LookAndFeelRadioButtonHandler event) {
    for(JRadioButton jRadio : jRadioButtonsLookAndFeelList)
        jRadio.addActionListener(event);
}
```

Registrando os eventos nos componentes

```
import br.inatel.ec203.projeto.controller.MainController;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

/**
 *
 * @author Felipe Miranda
 */
public class LookAndFeelApplyButtonHandler implements ActionListener {

    @Override
    public void actionPerformed(ActionEvent e) {
        MainController control = MainController.getInstance();
        control.displayGUIComponents();
    }
}
```

Exemplo do tratamento de um evento

## Aplicando Look And Feels e o tema Metal

```
public void applyLookAndFeel(String name) throws LookAndFeelException {  
    try {  
        String className = getAvailableLookAndFeels().get(name);  
        if (className.isEmpty()) {  
            UIManager.setLookAndFeel(UIManager.getCrossPlatformLookAndFeelClassName());  
        } else {  
            UIManager.setLookAndFeel(className);  
        }  
    } catch (ClassNotFoundException | InstantiationException  
            | IllegalAccessException | UnsupportedLookAndFeelException ex) {  
        throw new LookAndFeelException();  
    }  
}
```

```
public void applyMetalTheme(String name) {  
    switch (name) {  
        case "Ocean":  
            MetalLookAndFeel.setCurrentTheme(new OceanTheme());  
            break;  
        default:  
            MetalLookAndFeel.setCurrentTheme(new DefaultMetalTheme());  
            break;  
    }  
}
```

### Exercícios

#### **Faça crítica das entradas de dados**

1. Implemente um programa para o controle de inventário de equipamentos da sua empresa. Neste primeiro momento serão levantados notebooks e smartphones
    - i. Notebooks: Marca, modelo, matrícula do responsável e número de série do aparelho
    - ii. Smartphone: Marca, modelo, IMEI, Centro de Custo e matrícula do responsável
  
  2. O departamento de Relações com os Investidores da sua empresa gostaria de controlar os relatórios enviados e as comunicações com a imprensa que foram realizados. Para isso a TI foi contata e você o designado para desenvolver um sistema que automatize esse cadastro. As informações que deverão ser inseridas/exibidas são:
    - i. Relatórios: tema, funcionário responsável, funcionário revisor e data do envio
    - ii. Comunicações: tema, funcionário responsável, tipo de mídia e veículo (nome do jornal ou site)
- Obs.: para os funcionários armazenar nome e matrícula