

INSTITUTO FEDERAL DE EDUCAÇÃO, CIÊNCIA E  
TECNOLOGIA DO RIO GRANDE DO NORTE  
CAMPUS JOÃO CÂMARA

# MANIPULAÇÃO DE STRINGS, JCHECKBOX E JTEXTAREA

Nickerson Fonseca Ferreira  
[nickerson.ferreira@ifrn.edu.br](mailto:nickerson.ferreira@ifrn.edu.br)

# Manipulação de Strings

2

- O que é uma String ???
  - É uma classe Java que herda da classe Object.
- Podem ser criados de 2 formas:
  - `String s = new String("Teste");`
  - `String s = "Teste";`

# Manipulação de Strings

3

- Podemos fazer algumas operações com as Strings. A principal é a **concatenação**.
  - Utiliza-se o operador **+** para concatenar Strings.
- Como é um objeto, possui diversos métodos que podemos utilizar. Os principais são:
  - `equals(String outra)`
  - `length()`
  - `toLowerCase()` e `toUpperCase()`
  - `trim()`
  - `replace()`
  - `substring(int inicio, int fim)`
  - `indexOf(String pesquisar)`

# Exemplos

4

```
String t = ">>Nickerson Fonseca Ferreira>>";
System.out.println("Sem espaços: " + t.trim());
System.out.println("Todas maiúsculas: " + t.toUpperCase());
System.out.println("Todas minúsculas: " + t.toLowerCase());
System.out.println("Tamanho da String: " + t.length());
System.out.println("Substituído: " + t.replace(">", "#"));
System.out.println("Trecho: " + t.substring(0, 11));
```

```
Sem espaços: >>Nickerson Fonseca Ferreira>>
Todas maiúsculas: >>NICKERSON FONSECA FERREIRA>>
Todas minúsculas: >>nickerson fonseca ferreira>>
Tamanho da String: 37
Substituído: ##Nickerson Fonseca Ferreira##
Trecho: >>Nickerson
```

# JCheckBox (Caixa de seleção)

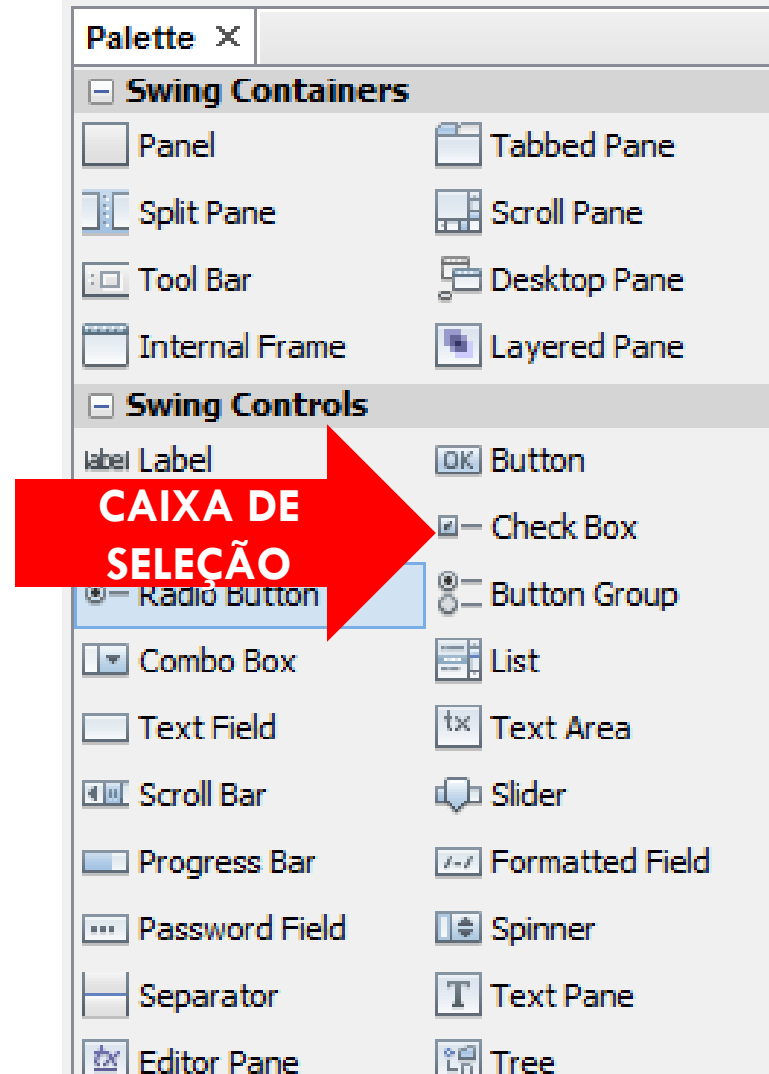
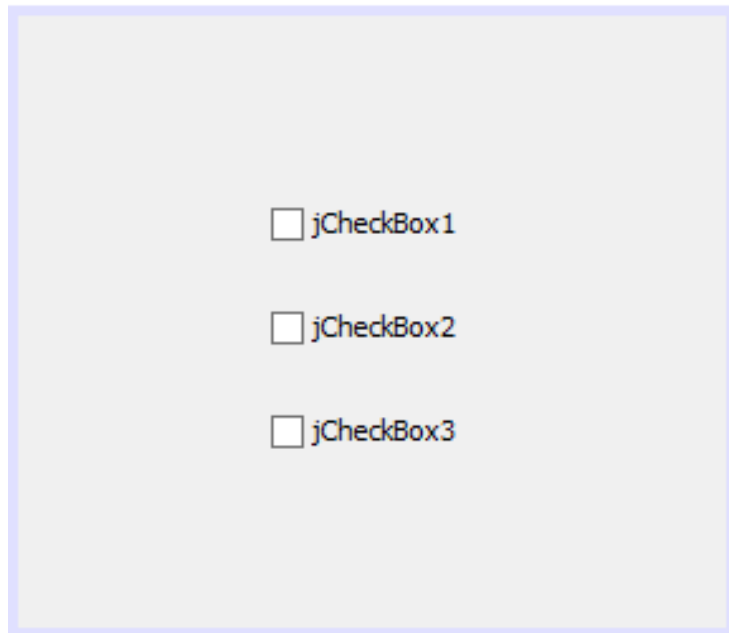
5

- ❑ Funciona de forma semelhante ao JRadioButton que vimos na aula anterior.
- ❑ A diferença é que este componente é utilizado para seleção de uma ou várias opções.



# JCheckBox (caixa de seleção)

6



# JCheckBox (Caixa de seleção)

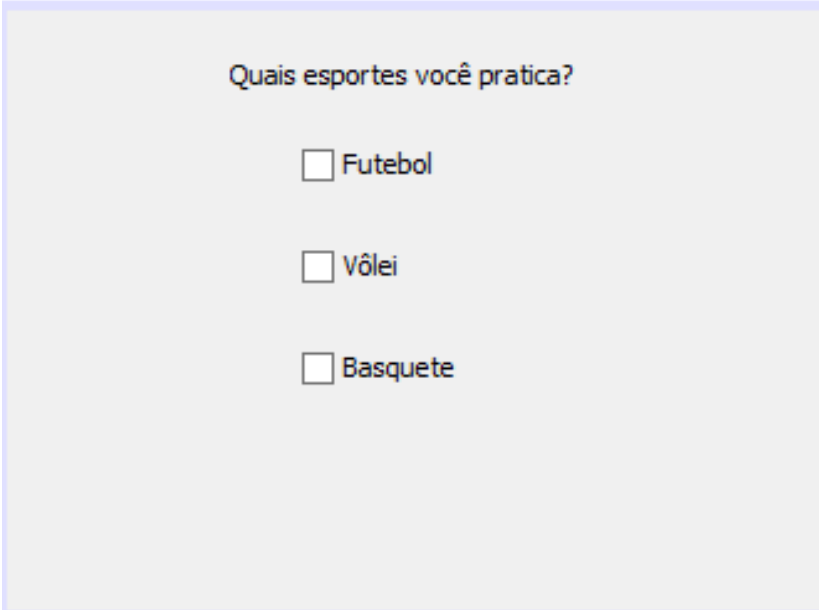
7

- Principais propriedades:
  - `enabled`: esta propriedade indica se a caixa de seleção está ativa ou não.
  - `selected`: indica se, ao abrir a página, a opção estará marcada.
  - `text`: texto apresentado para o usuário.

# Utilizando eventos no JCheckBox

8

- Podemos incluir eventos que serão executados quando o usuário alterar o estado do componente.



Quais esportes você pratica?

Futebol

Vôlei

Basquete



# Utilizando eventos no JCheckBox

9

The screenshot shows an IDE interface with a Swing window titled "Quais esportes você pratica?". Inside the window, there are three checkboxes: "Futebol", "Vôlei", and "Basquete". A context menu is open over the "Futebol" checkbox, with the "Events" option selected. This opens a sub-menu where "Item" is selected, and "itemStateChanged [op1ItemStateChanged]" is chosen. On the right side, the "Properties" palette is visible, showing the "op1 [JCheckBox]" component with its "itemStateChanged" property set to "op1ItemStateChanged". At the bottom left, the "Output" window shows the message "BUILD SUCCESSFUL".

# Utilizando eventos no JCheckBox

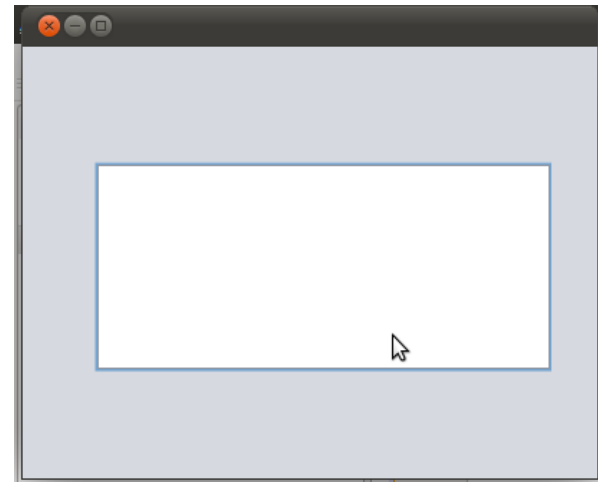
10

```
private void op1ItemStateChanged(java.awt.event.ItemEvent evt) {  
    // TODO add your handling code here:  
    String conteudo = lblLista.getText();  
  
    if (op1.isSelected()){  
        String texto = op1.getText() + ",";  
        lblLista.setText(conteudo + texto);  
    } else {  
        String texto = op1.getText() + ",";  
        conteudo = conteudo.replace(texto, "");  
        lblLista.setText(conteudo);  
    }  
}
```

# JTextArea

11

- ❑ Componente semelhante ao JTextField.
- ❑ A diferença é que podemos escrever diversas linhas de texto, ou seja, a área para digitação de texto é superior.
- ❑ Podemos definir o tamanho da área de texto (altura e largura).



# Propriedades JTextArea

12

- Columns: indica a quantidade de colunas que o componente apresentará (largura do componente).
- Editable: informa se o componente pode ser editável.
- lineWrap: esta opção indica se as linhas serão quebradas automaticamente.
- Rows: quantidade de linhas que o componente possui (altura do componente).
- Text: inclui um texto padrão no componente.

# Exercício

13

