



Pesquisa – Biblioteca Swing

Disciplina: Programação Orientada a Objeto (POO)

Professor: Diógenes Carvalho Matias

Turma: CCO-3

Unidade: Boa Vista

Turno: Manhã

Alunos: Cristian Matheus Galindo de Brito (202002728)

Paulo Emanuel Madeira de Freitas (202003566)

Paulo Montenegro Campos (202000095)

JButton

O JButton é um componente muito utilizado em Java para a criação de formulários, ou outros programas, onde é necessário o uso de botões para acionar ações dentro deles ou externas, como salvar informações em um banco de dados, ou em um arquivo, entre outros.

```
ExemploBotao.java X
1 package estudosSwing;
2
3 import javax.swing.*;
4
5 public class ExemploBotao {
6
7     public static void main(String[] args) {
8         JFrame f = new JFrame("Exemplo Botão");
9         JButton b = new JButton ("Clique aqui");
10        b.setBounds(50, 100, 125, 30);
11        f.add(b);
12        f.setSize(400,400);
13        f.setLayout(null);
14        f.setVisible(true);
15    }
16 }
17 }
```

JLabel

Os JLabels são rótulos que podem ser exibidos nos frames. São elementos estáticos, não sendo usados para interagir com o usuário. Um rótulo em programação Java é o mesmo que um rótulo no mundo real, um texto e/ou imagem.

```
ExemploLabel.java X
1 package estudosSwing;
2
3 import javax.swing.*;
4
5 public class ExemploLabel {
6
7     public static void main(String[] args) {
8         JFrame f = new JFrame("Exemplo Label");
9         JLabel l1,l2;
10        l1 = new JLabel("Primeiro Label.");
11        l1.setBounds(50,50,100,30);
12        l2 = new JLabel("Segundo Label.");
13        l2.setBounds(50,50,100,30);
14        f.add(l1);
15        f.add(l2);
16        f.setSize(300,300);
17        f.setLayout(null);
18        f.setVisible(true);
19    }
20 }
21 }
```

JPanel

O JPanel serve para receber diversos componentes, inclusive outros JPanels, que terão seu posicionamento gerenciado pelo layoutManager. O JPanel pode ser utilizado para ajudar na formação do layout dos seus componentes em um frame por exemplo.

```
ExemploPanel.java X
1 package estudosSwing;
2
3 import java.awt.*;
4
5
6 public class ExemploPanel {
7
8     ExemploPanel(){
9         JFrame f = new JFrame ("Exemplo Panel");
10        JPanel panel = new JPanel();
11        panel.setBounds(40, 80, 200, 200);
12        panel.setBackground(Color.black);
13        JButton b1 = new JButton("Botão 1");
14        b1.setBounds(50,100,80,30);
15        b1.setBackground(Color.yellow);
16        JButton b2 = new JButton ("Botão 2");
17        b2.setBounds(100, 100, 80, 30);
18        b2.setBackground(Color.cyan);
19        panel.add(b1);
20        panel.add(b2);
21        f.add(panel);
22        f.setSize(400,400);
23        f.setLayout(null);
24        f.setVisible(true);
25    }
26
27    public static void main(String[] args) {
28        new ExemploPanel();
29    }
30 }
31
```

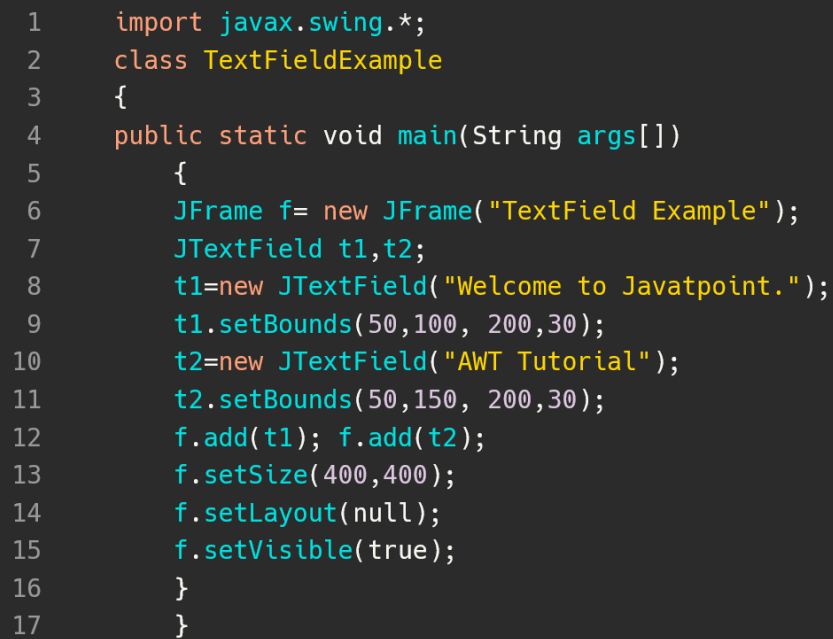
JTextArea

O JTextArea é um componente utilizado para a entrada/saída de textos ou frases maiores. Ele permite que seja adicionada mais de uma linha.

```
1  import javax.swing.*;
2  import java.awt.event.*;
3  public class TextAreaExample implements ActionListener{
4
5      JLabel l1,l2;
6      JTextArea area;
7      JButton b;
8      TextAreaExample() {
9          JFrame f= new JFrame();
10         l1=new JLabel();
11         l1.setBounds(50,25,100,30);
12         l2=new JLabel();
13         l2.setBounds(160,25,100,30);
14         area=new JTextArea();
15         area.setBounds(20,75,250,200);
16         b=new JButton("Count Words");
17         b.setBounds(100,300,120,30);
18         b.addActionListener(this);
19         f.add(l1);f.add(l2);f.add(area);f.add(b);
20         f.setSize(450,450);
21         f.setLayout(null);
22         f.setVisible(true);
23     }
24     public void actionPerformed(ActionEvent e){
25         String text=area.getText();
26         String words[]=text.split("\\s");
27         l1.setText("Words: "+words.length);
28         l2.setText("Characters: "+text.length());
29     }
30     public static void main(String[] args) {
31         new TextAreaExample();
32     }
33 }
```

JTextField

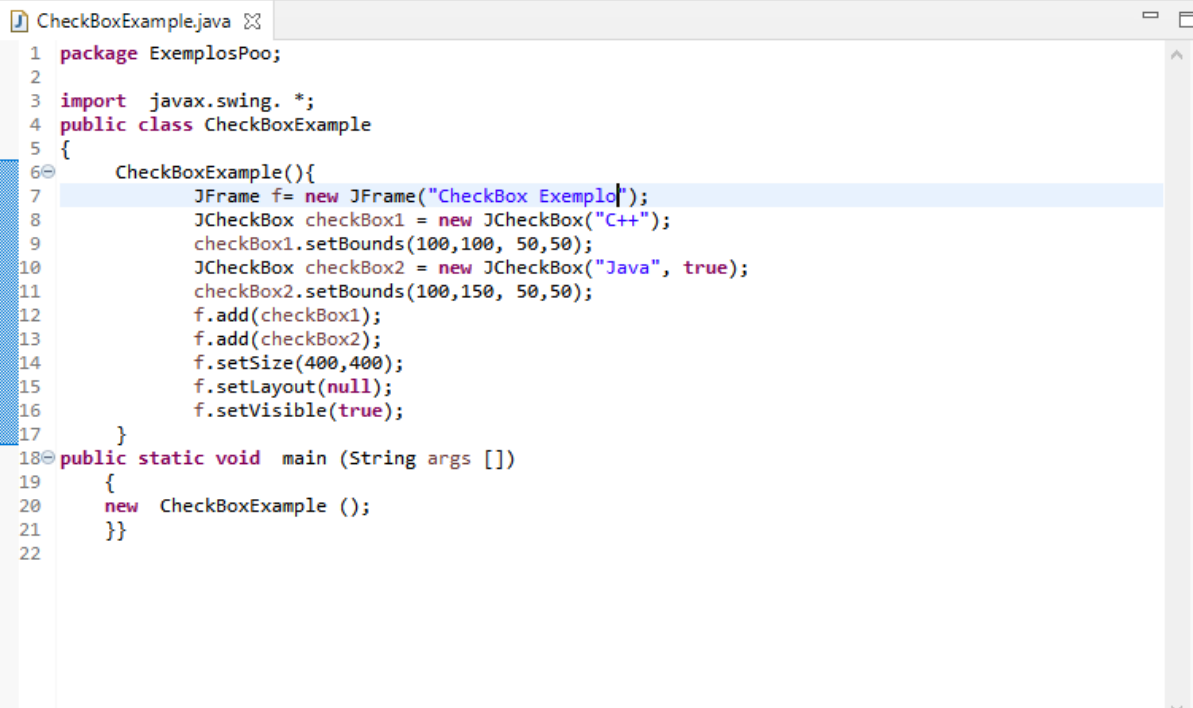
O JTextField é um campo utilizado para a entrada de informações. Normalmente é uma caixa branca onde o usuário pode entrar com algum valor para que seja utilizado pelo algoritmo



```
1  import javax.swing.*;
2  class TextFieldExample
3  {
4      public static void main(String args[])
5      {
6          JFrame f= new JFrame("TextField Example");
7          JTextField t1,t2;
8          t1=new JTextField("Welcome to Javatpoint.");
9          t1.setBounds(50,100, 200,30);
10         t2=new JTextField("AWT Tutorial");
11         t2.setBounds(50,150, 200,30);
12         f.add(t1); f.add(t2);
13         f.setSize(400,400);
14         f.setLayout(null);
15         f.setVisible(true);
16     }
17 }
```

JCheckBox

A classe JCheckBox é usada para criar uma caixa de seleção. É usado para ligar uma opção (verdadeiro) ou desligar (falso). Clicar em um CheckBox muda seu estado de "ligado" para "desligado" ou de "desligado" para "ligado".



```
1 package ExemplosPoo;
2
3 import javax.swing.*;
4 public class CheckBoxExample
5 {
6     CheckBoxExample(){
7         JFrame f= new JFrame("CheckBox Exemplo");
8         JCheckBox checkBox1 = new JCheckBox("C++");
9         checkBox1.setBounds(100,100, 50,50);
10        JCheckBox checkBox2 = new JCheckBox("Java", true);
11        checkBox2.setBounds(100,150, 50,50);
12        f.add(checkBox1);
13        f.add(checkBox2);
14        f.setSize(400,400);
15        f.setLayout(null);
16        f.setVisible(true);
17    }
18    public static void main (String args [])
19    {
20        new CheckBoxExample ();
21    }
22 }
```

JComboBox

O objeto da classe Choice é usado para mostrar o menu pop-up de opções. A escolha selecionada pelo usuário é mostrada no topo de um menu . Ele herda a classe JComponent.



```
1 package ExemplosPoo;
2
3 import javax.swing.*;
4 public class JComboBoxexemplo {
5     JFrame f;
6     JComboBoxexemplo(){
7         f=new JFrame("ComboBox Exemplo");
8         String country[]={"India","Aus","U.S.A","England","Newzealand"};
9         JComboBox cb=new JComboBox(country);
10        cb.setBounds(50, 50,90,20);
11        f.add(cb);
12        f.setLayout(null);
13        f.setSize(400,500);
14        f.setVisible(true);
15    }
16    public static void main(String[] args) {
17        new JComboBoxexemplo();
18    }
19 }
```