

# **JAVA SWING** *com* **Eventos**

Aplicações de Linguagem de Programação Orientada a Objetos

Carlos Arruda Baltazar

UNIP

**JLabel** representa um rótulo/etiqueta de texto. Entre os principais métodos temos:

- ***setText(String)*** que insere um texto à etiqueta (o texto também pode ser inserido diretamente no método construtor do objeto);
- ***getText()*** que retorna o texto contido na etiqueta.
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;

## Eventos:

- ***MouseListener()*** é uma interface para receber eventos relacionado ao *mouse* como: pressão; clique e etc.
- ***mouseClicked(MouseEvent)*** é invocado quando o botão do mouse (direito ou esquerdo) é clicado.

```
package Pck_Frame;

import javax.swing.*;
import javax.swing.border.Border;

import java.awt.*;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;

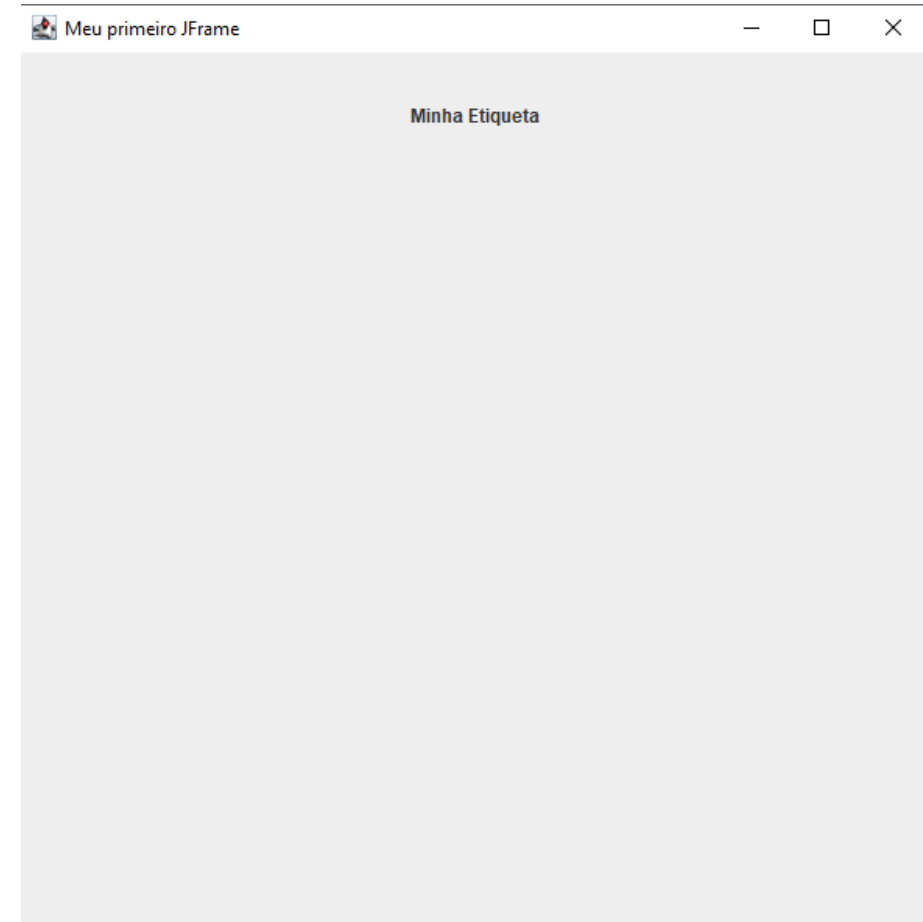
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;

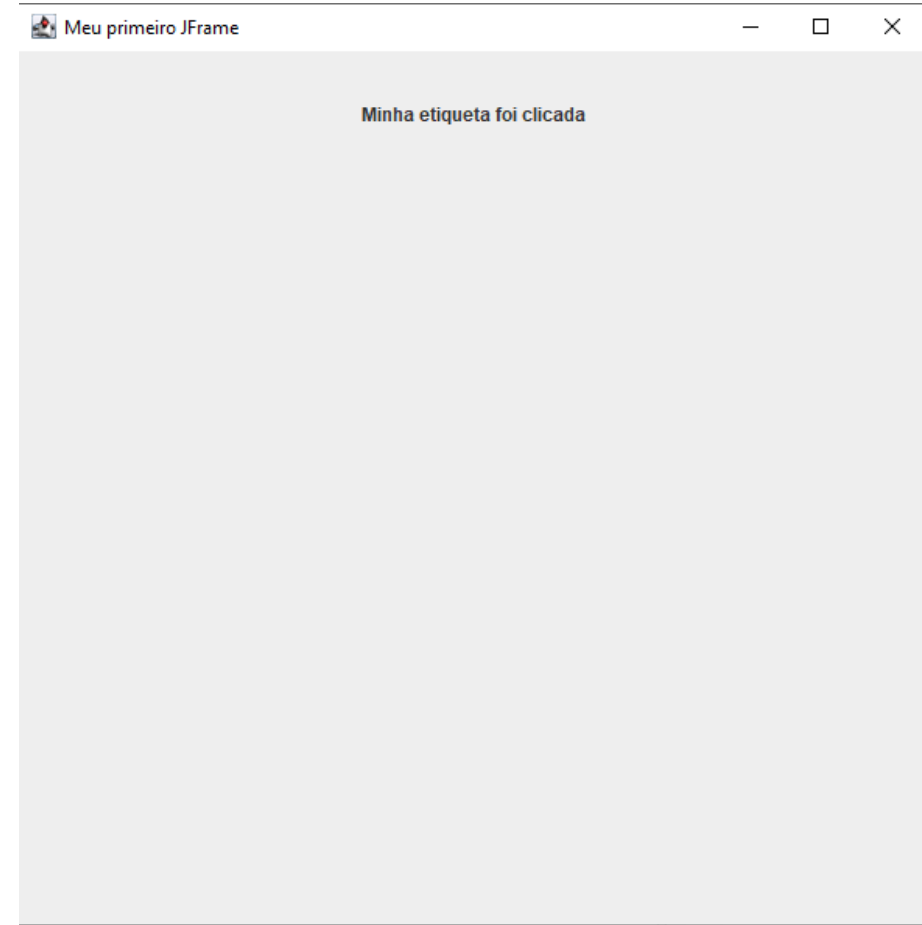
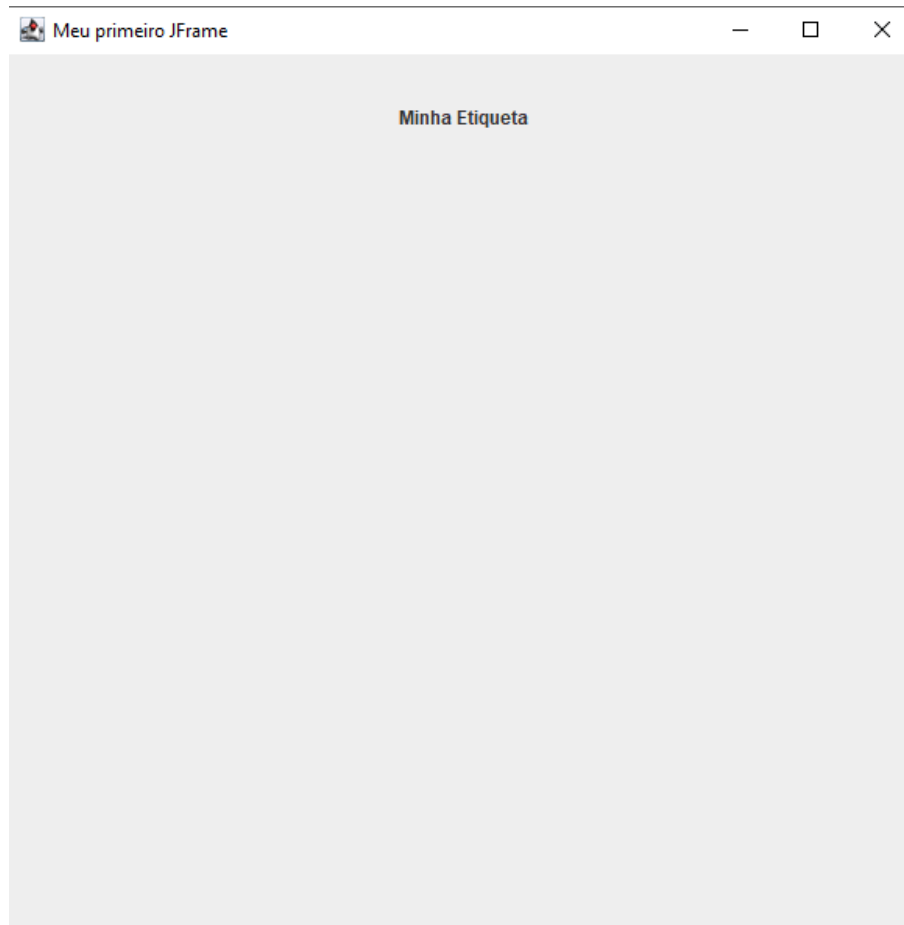
    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);
    }

    public void CreateEvent()
    {
        this.jl_etiqueta.addMouseListener(new MouseAdapter()
        {
            public void mouseClicked(MouseEvent e)
            {
                jl_etiqueta.setText("Minha etiqueta foi clicada");
            }
        });
    }
}
```





**JTextField** representa um campo para preenchimento de textos curtos de única linha. Entre os principais métodos temos:

- ***setText(String)*** que insere um texto padrão ao campo (o texto também pode ser inserido diretamente no método construtor do objeto);
- ***getText()*** que retorna o texto inserido no campo.
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;

## Eventos:

- ***ActionListener()*** é uma interface para receber qualquer evento de ação sobre um objeto.
- ***actionPerformed(ActionEvent)*** é invocado quando uma ação ocorre.

```
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JTextField jtf_texto;

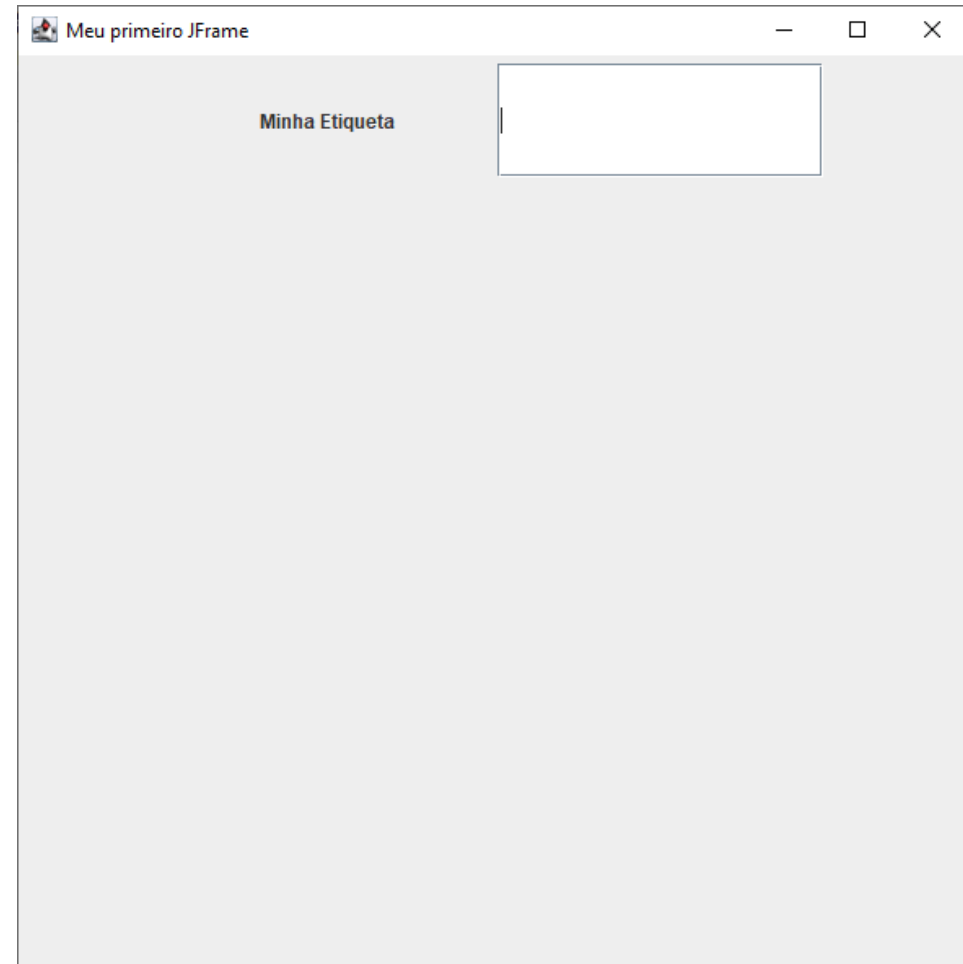
    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

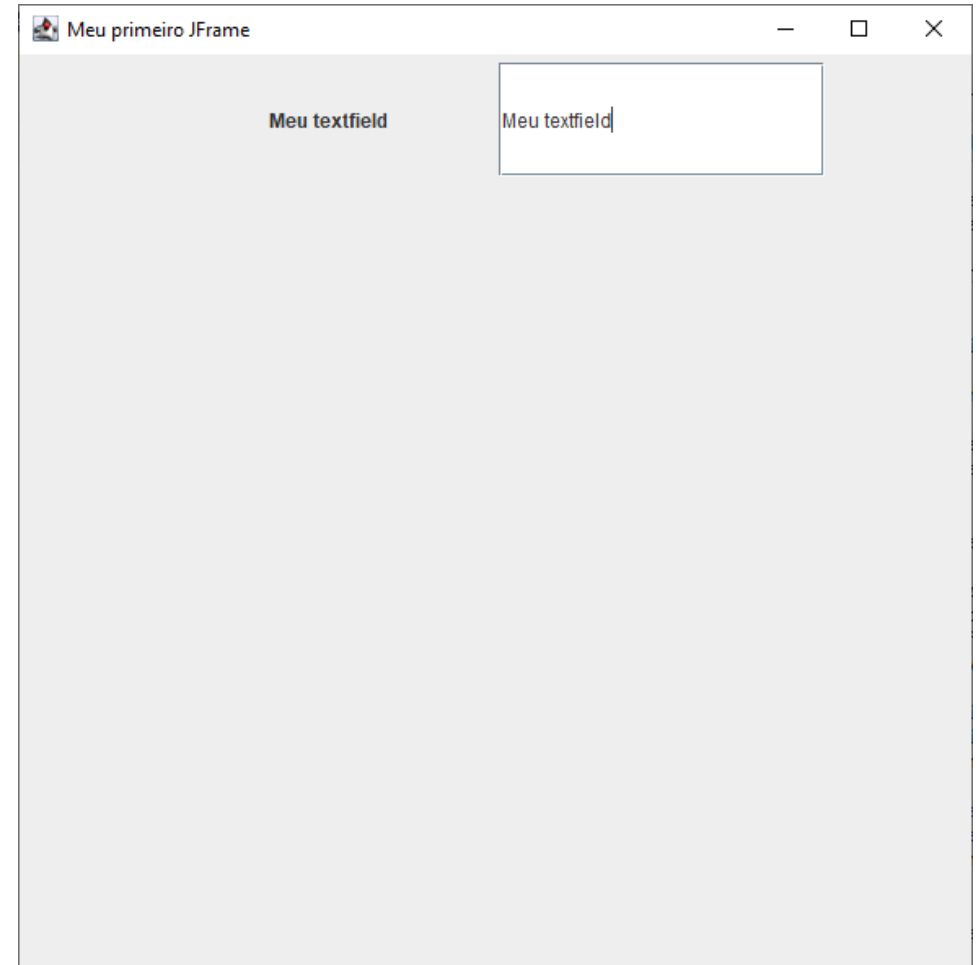
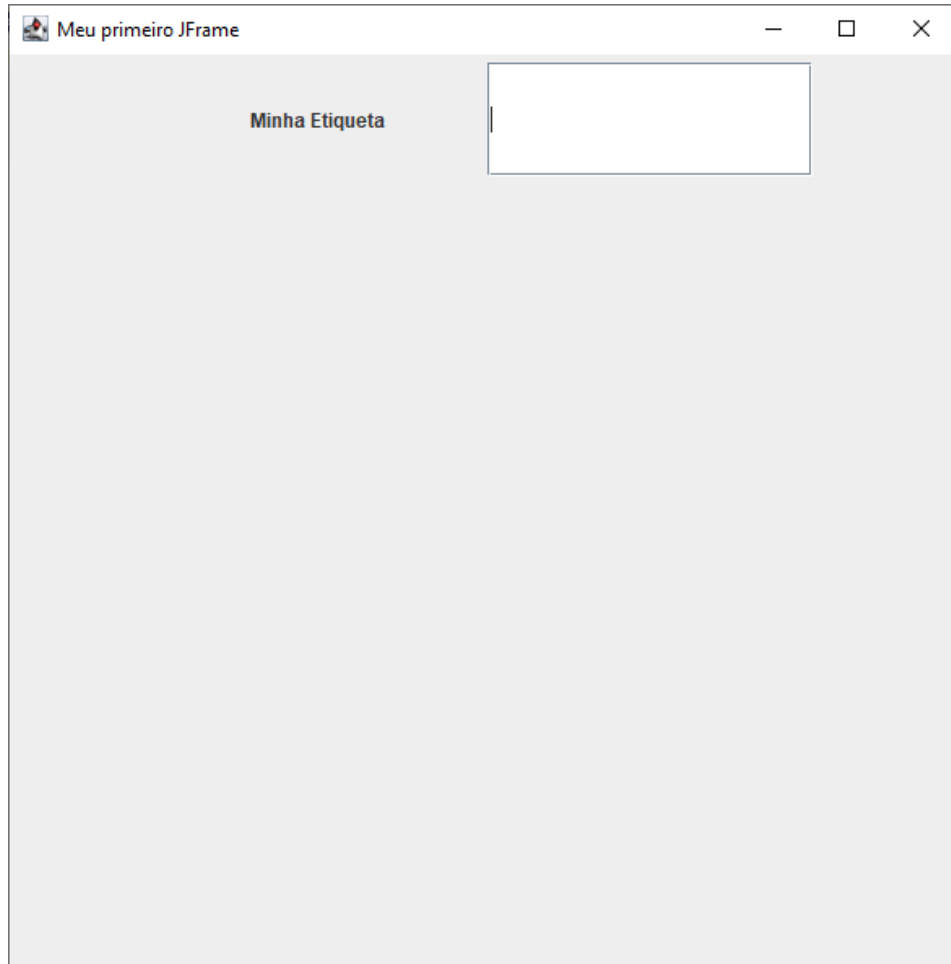
    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta",SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200,70));
        this.getContentPane().add("Center",this.jl_etiqueta);

        this.jtf_texto = new JTextField();
        this.jtf_texto.setPreferredSize(new Dimension(200,70));
        this.getContentPane().add("Center",this.jtf_texto);
    }

    public void CreateEvent()
    {
        this.jtf_texto.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                jl_etiqueta.setText(jtf_texto.getText());
            }
        });
    }
}
```





**JPasswordField** representa um campo de texto protegido, subclasse de `JTextField`. Entre os principais métodos temos:

- **`setText(String)`** que insere um texto padrão ao campo (o texto também pode ser inserido diretamente no método construtor do objeto);
- **`getText()`** que retorna o texto inserido no campo.
- **`setSize(int, int)`** que define as medidas largura e altura da etiqueta em pixels;
- **`setEchoChar(char)`** que define o caractere que aparece ao digitar um texto.

## Eventos:

- **`ActionListener()`** é uma interface para receber qualquer evento de ação sobre um objeto.
- **`actionPerformed(ActionEvent)`** é invocado quando uma ação ocorre.



```
import java.awt.*;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;

public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JPasswordField jpf_senha;

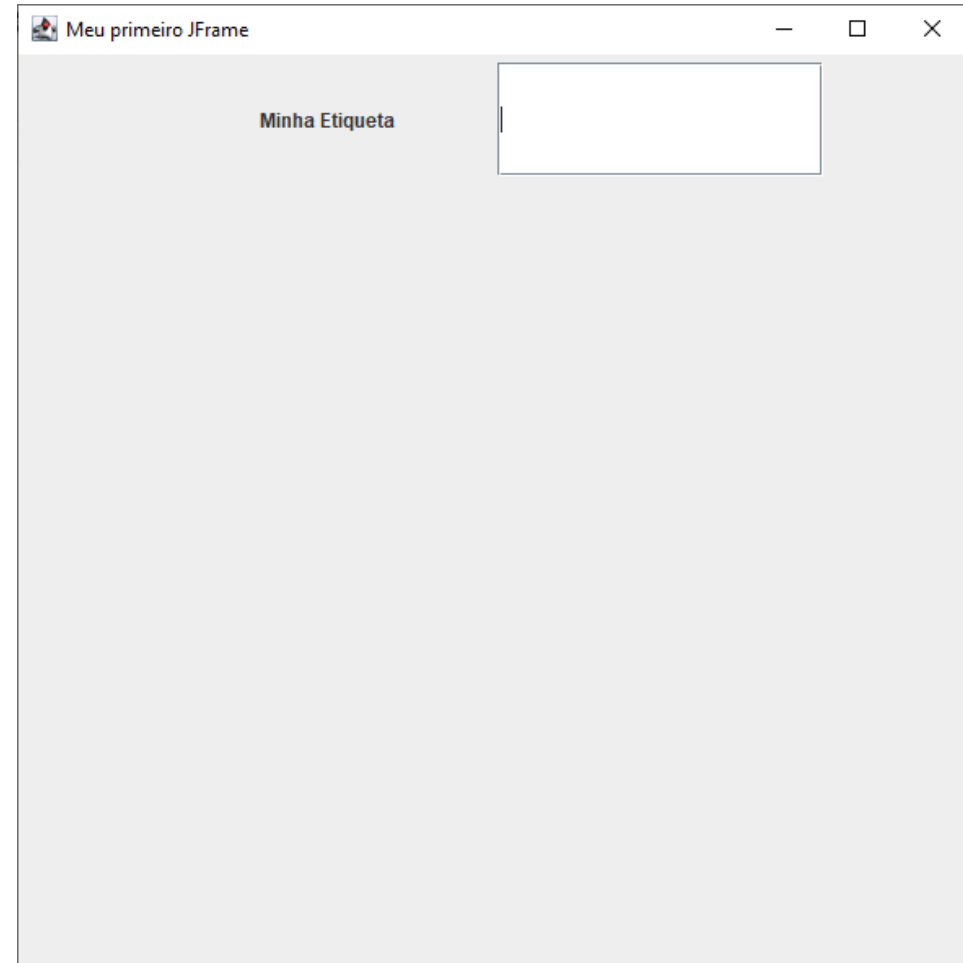
    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

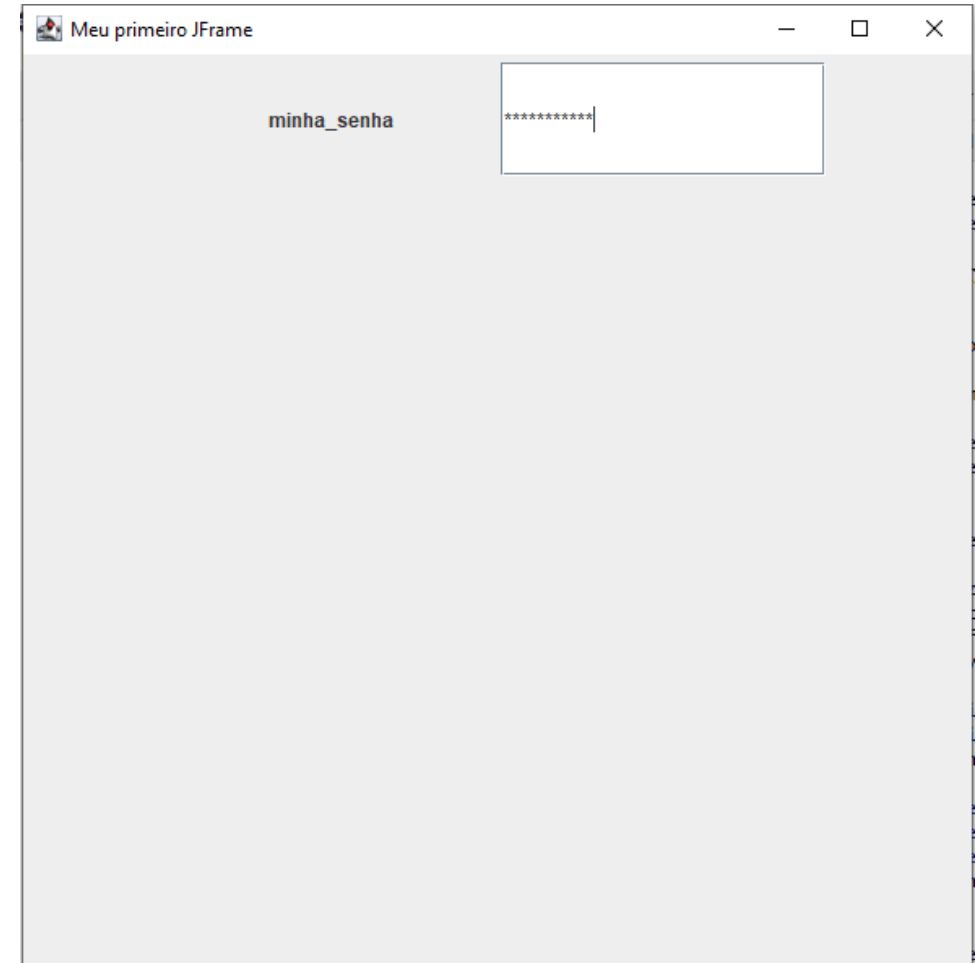
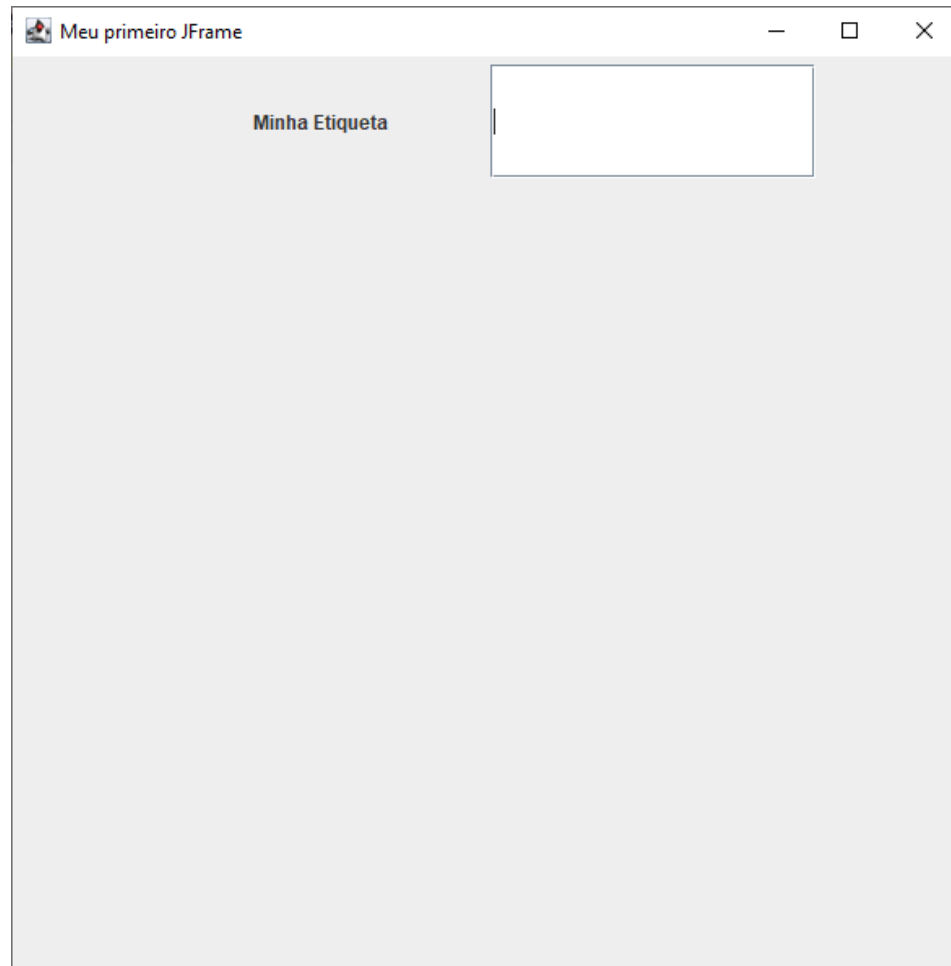
    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);

        this.jpf_senha = new JPasswordField();
        this.jpf_senha.setPreferredSize(new Dimension(200, 70));
        this.jpf_senha.setEchoChar('*');
        this.getContentPane().add("Center", this.jpf_senha);
    }

    public void CreateEvent()
    {
        this.jpf_senha.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                jl_etiqueta.setText(jpf_senha.getText());
            }
        });
    }
}
```





**JTextArea** representa um campo que pode ser preenchido com várias linhas de texto. Entre os principais métodos temos:

- ***setText(String)*** que insere um texto padrão ao campo (o texto também pode ser inserido diretamente no método construtor do objeto);
- ***getText()*** que retorna o texto inserido no campo.
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;
- ***getSelectedText()*** que retorna o texto selecionado pelo usuário;
- ***insert(String, int)*** que insere um texto na posição especificada.

**JTextArea** representa um campo que pode ser preenchido com várias linhas de texto. Entre os principais métodos temos:

## Eventos:

- ***DocumentListener()*** é uma interface para receber qualquer evento de ação sobre um objeto documento, que ser obtido de um *JTextArea*.
- ***insertUpdate(ActionEvent)*** é invocado quando qualquer caractere é inserido.
- ***changedUpdate(ActionEvent)*** é invocado quando qualquer caractere é alterado.
- ***removeUpdate(ActionEvent)*** é invocado quando qualquer caractere é removido.

```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JTextArea jta_texto;

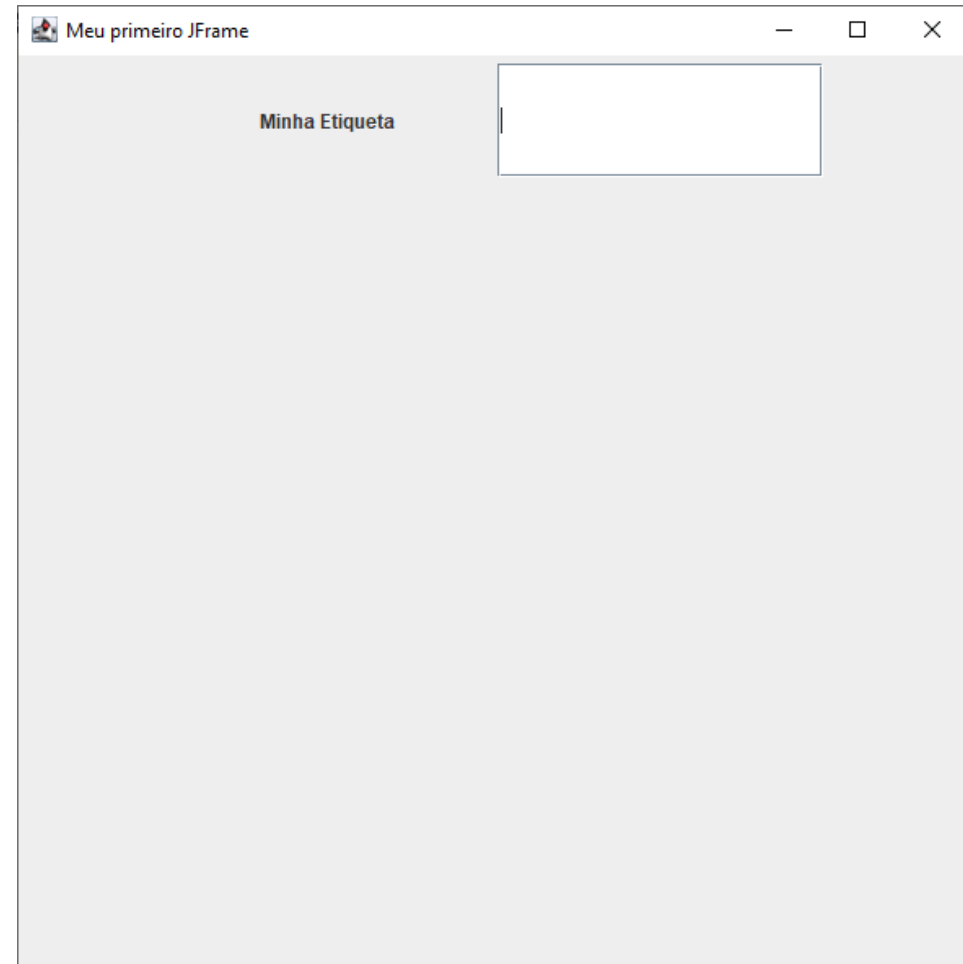
    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

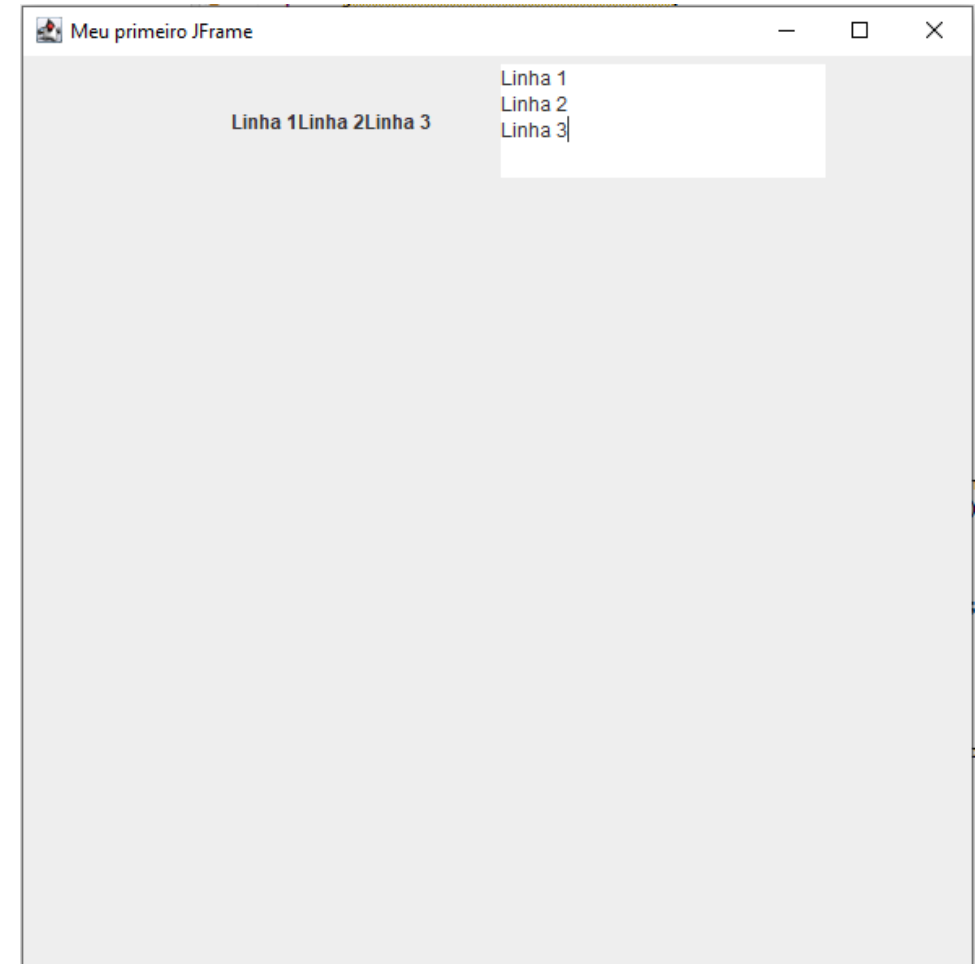
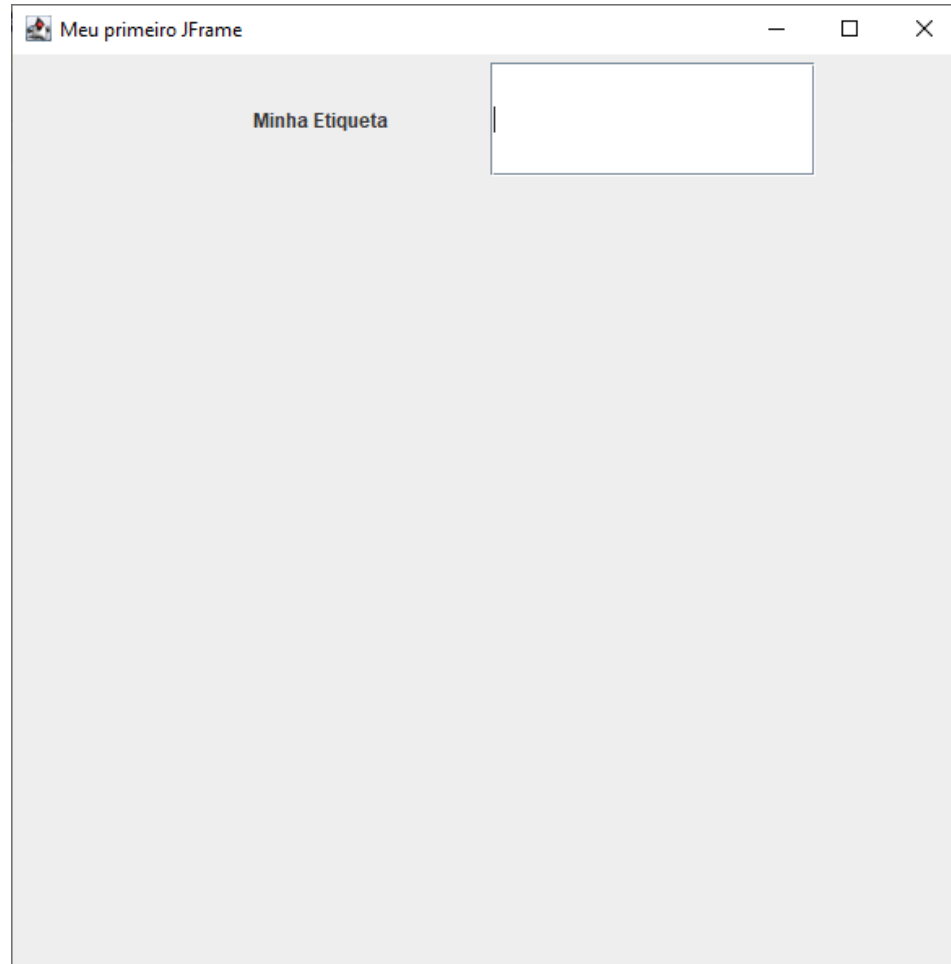
    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);

        this.jta_texto = new JTextArea();
        this.jta_texto.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jta_texto);
    }

    public void CreateEvent()
    {
        this.jta_texto.getDocument().addDocumentListener(new DocumentListener()
        {
            public void changedUpdate(DocumentEvent arg0)
            {
                jl_etiqueta.setText(jta_texto.getText());
            }
            public void insertUpdate(DocumentEvent e)
            {
                jl_etiqueta.setText(jta_texto.getText());
            }
            public void removeUpdate(DocumentEvent e)
            {
                jl_etiqueta.setText(jta_texto.getText());
            }
        });
    }
}
```





**JCheckBox** representa uma caixa de seleção e permite selecionar mais de uma opção. Entre os principais métodos temos:

- ***setText(String)*** que insere um texto padrão ao campo (o texto também pode ser inserido diretamente no método construtor do objeto);
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;
- ***setSelected(boolean)*** que altera o estado da caixa de seleção;
- ***isSelected()*** que retorna ***true*** se a caixa estiver marcada e ***false*** se não estiver marcada.

## Eventos:

- ***ItemListener()*** é uma interface para receber um evento de seleção de item.
- ***itemStateChanged(ItemEvent)*** é invocado um item é selecionado ou desselecionado pelo usuário.

```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JCheckBox jcb_opcao1;
    private JCheckBox jcb_opcao2;
    private JCheckBox jcb_opcao3;

    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

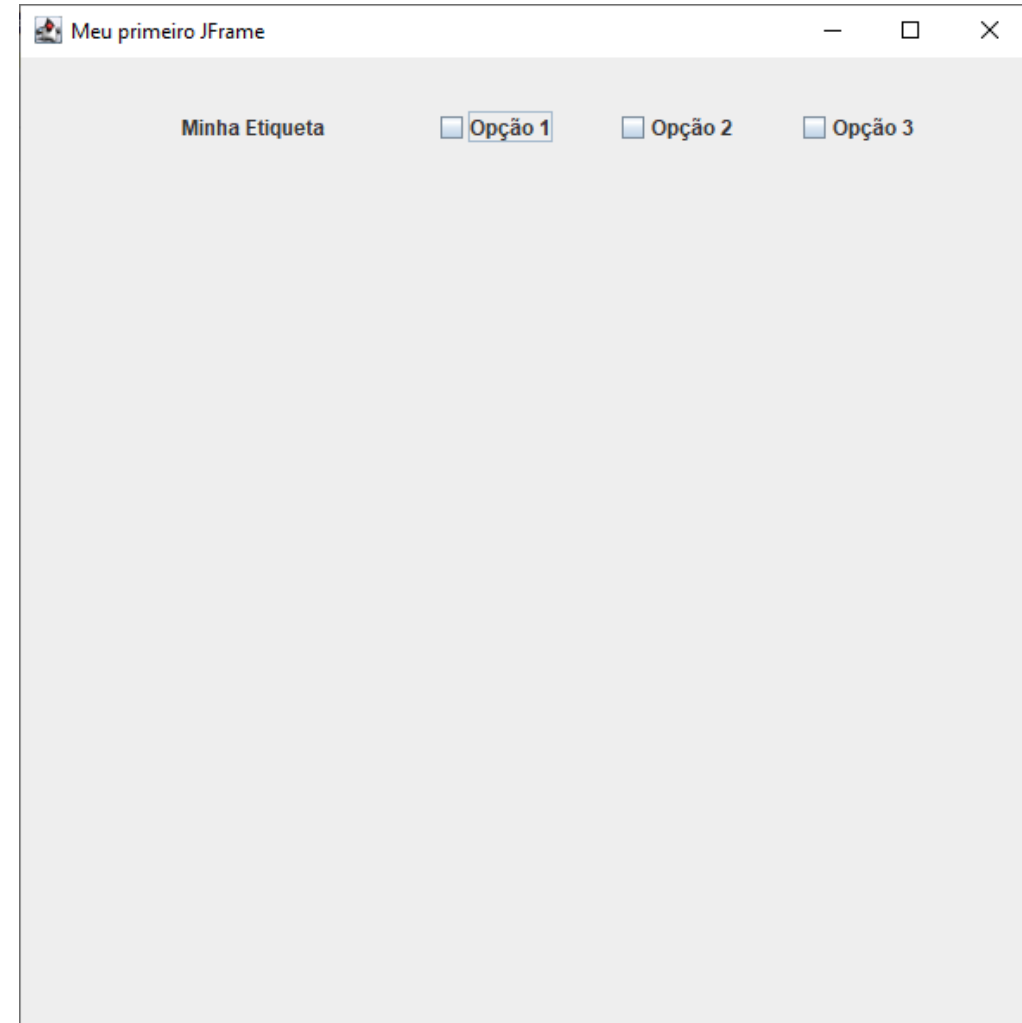
        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);

        this.jcb_opcao1 = new JCheckBox("Opção 1");
        this.jcb_opcao1.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jcb_opcao1);

        this.jcb_opcao2 = new JCheckBox("Opção 2");
        this.jcb_opcao2.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jcb_opcao2);

        this.jcb_opcao3 = new JCheckBox("Opção 3");
        this.jcb_opcao3.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jcb_opcao3);
    }

    public void CreateEvent()
    {
        this.jcb_opcao1.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jcb_opcao1.getText());
            }
        });
        this.jcb_opcao2.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jcb_opcao2.getText());
            }
        });
        this.jcb_opcao3.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jcb_opcao3.getText());
            }
        });
    }
}
```





```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JCheckBox jcb_opcao1;
    private JCheckBox jcb_opcao2;
    private JCheckBox jcb_opcao3;

    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

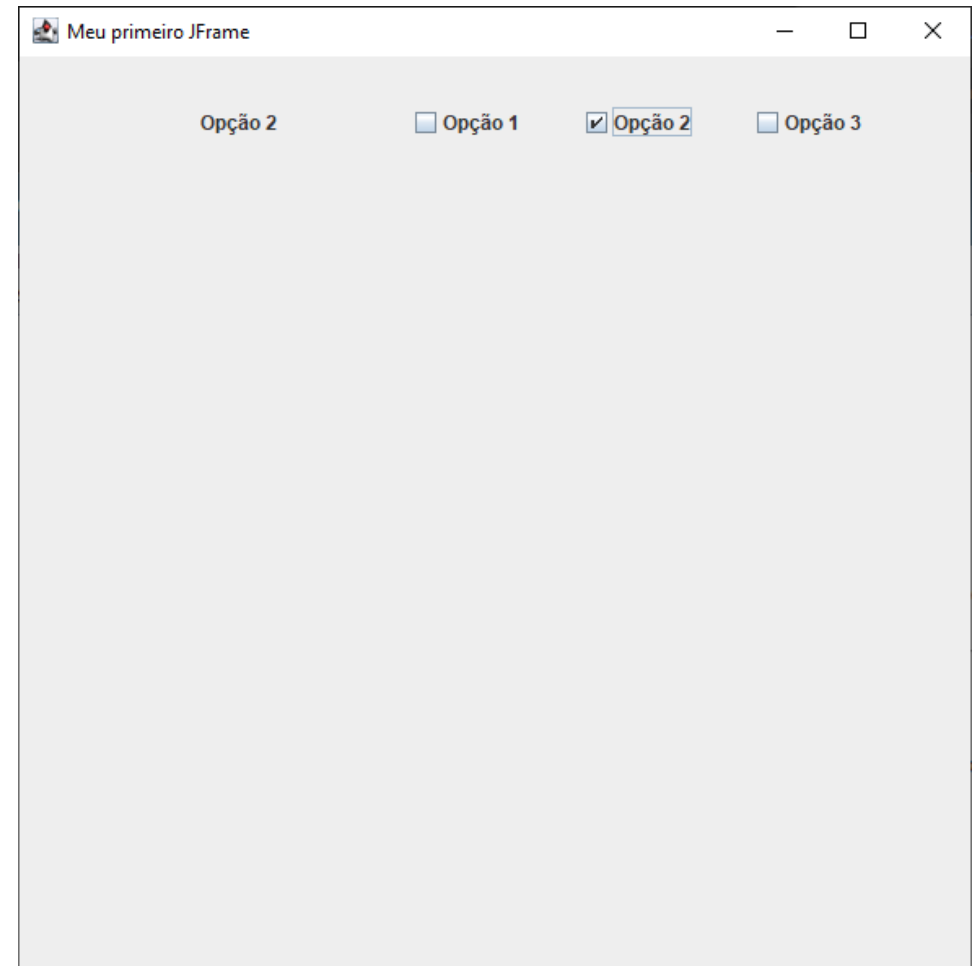
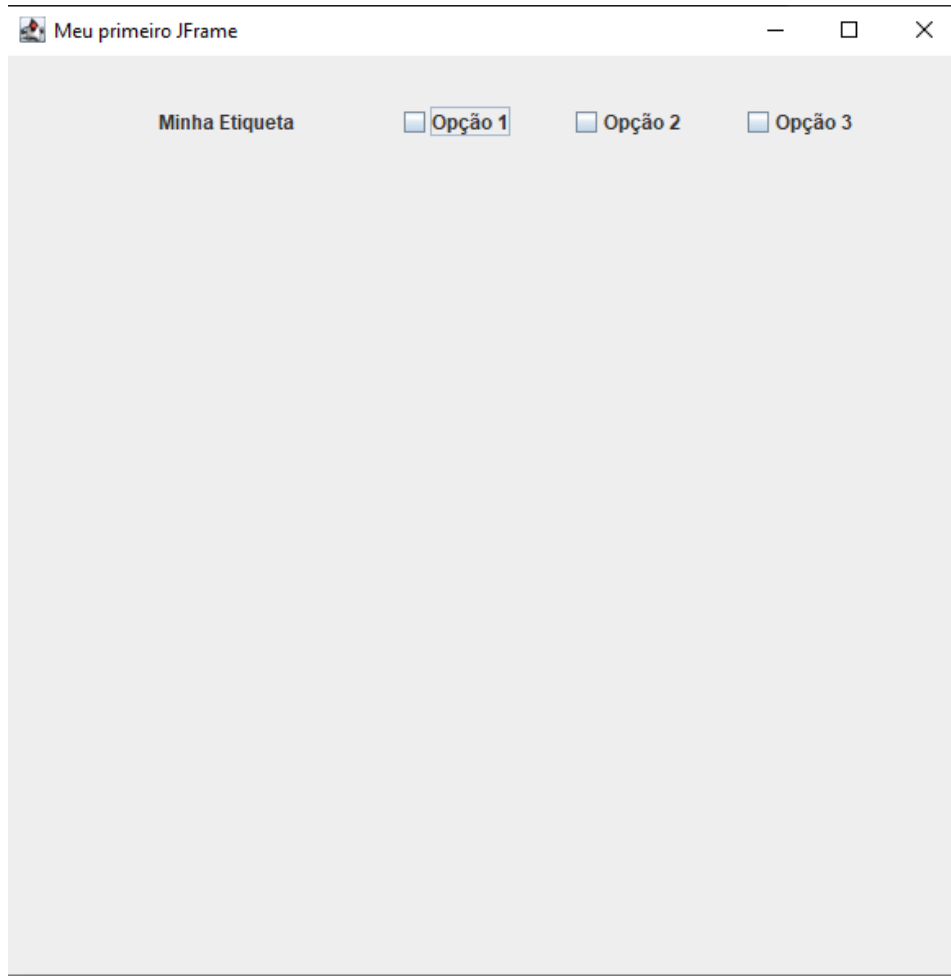
        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);

        this.jcb_opcao1 = new JCheckBox("Opção 1");
        this.jcb_opcao1.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jcb_opcao1);

        this.jcb_opcao2 = new JCheckBox("Opção 2");
        this.jcb_opcao2.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jcb_opcao2);

        this.jcb_opcao3 = new JCheckBox("Opção 3");
        this.jcb_opcao3.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jcb_opcao3);
    }
}
```

```
public void CreateEvent()
{
    this.jcb_opcao1.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            jl_etiqueta.setText(jcb_opcao1.getText());
        }
    });
    this.jcb_opcao2.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            jl_etiqueta.setText(jcb_opcao2.getText());
        }
    });
    this.jcb_opcao3.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            jl_etiqueta.setText(jcb_opcao3.getText());
        }
    });
}
```



**JRadioButton** representa um componente que permite selecionar uma única opção. Entre os principais métodos temos:

- ***setText(String)*** que insere um texto padrão ao campo (o texto também pode ser inserido diretamente no método construtor do objeto);
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;
- ***setSelected(boolean)*** que altera o estado da caixa de seleção;
- ***isSelected()*** que retorna ***true*** se a caixa estiver marcada e ***false*** se não estiver marcada.

## Eventos:

- ***ItemListener()*** é uma interface para receber um evento de seleção de item.
- ***itemStateChanged(ItemEvent)*** é invocado quando um item é selecionado ou desselecionado pelo usuário.

```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private ButtonGroup bg_grupo;
    private JRadioButton jrb_opcao1;
    private JRadioButton jrb_opcao2;
    private JRadioButton jrb_opcao3;

    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);

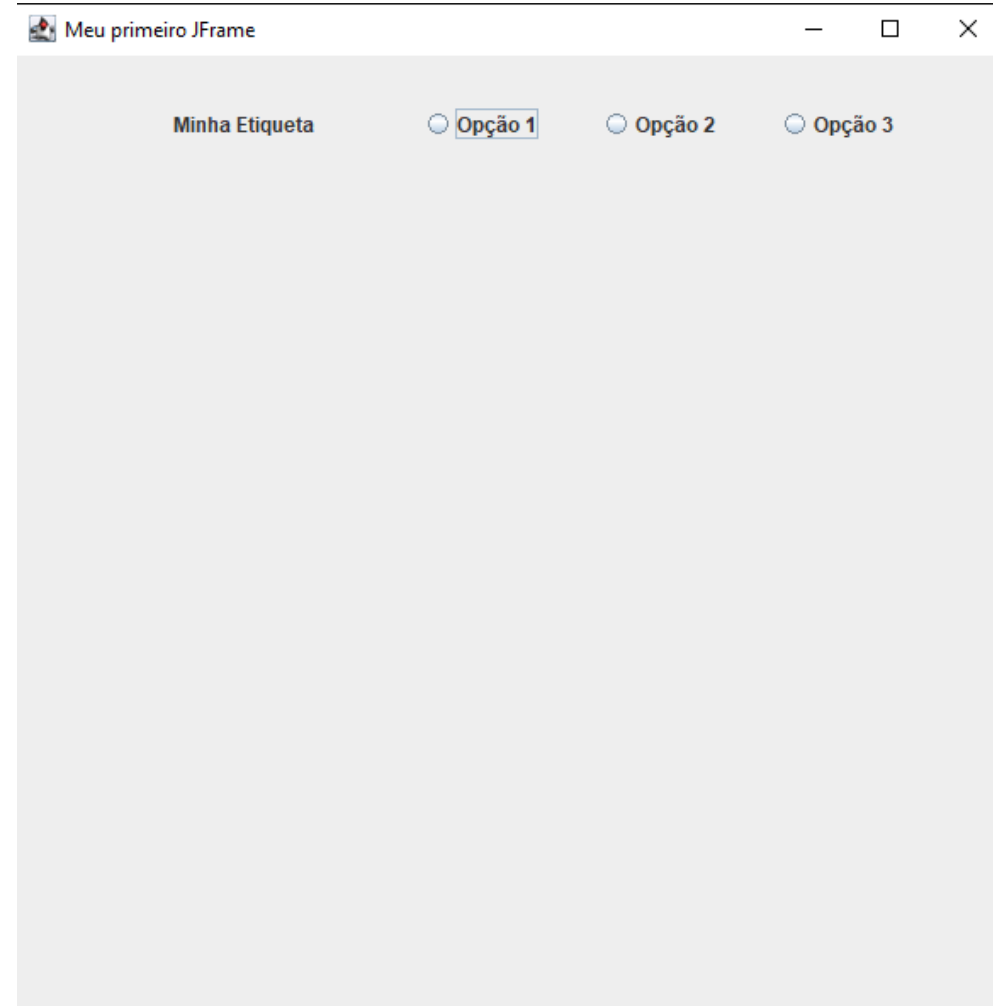
        this.jrb_opcao1 = new JRadioButton("Opção 1");
        this.jrb_opcao1.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jrb_opcao1);

        this.jrb_opcao2 = new JRadioButton("Opção 2");
        this.jrb_opcao2.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jrb_opcao2);

        this.jrb_opcao3 = new JRadioButton("Opção 3");
        this.jrb_opcao3.setPreferredSize(new Dimension(100, 70));
        this.getContentPane().add("Center", this.jrb_opcao3);

        this.bg_grupo = new ButtonGroup();
        this.bg_grupo.add(jrb_opcao1);
        this.bg_grupo.add(jrb_opcao2);
        this.bg_grupo.add(jrb_opcao3);
    }

    public void CreateEvent()
    {
        this.jrb_opcao1.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jrb_opcao1.getText());
            }
        });
        this.jrb_opcao2.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jrb_opcao2.getText());
            }
        });
        this.jrb_opcao3.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jrb_opcao3.getText());
            }
        });
    }
}
```



```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private ButtonGroup bg_grupo;
    private JRadioButton jrb_opcao1;
    private JRadioButton jrb_opcao2;
    private JRadioButton jrb_opcao3;

    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }
    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta",SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200,70));
        this.getContentPane().add("Center",this.jl_etiqueta);

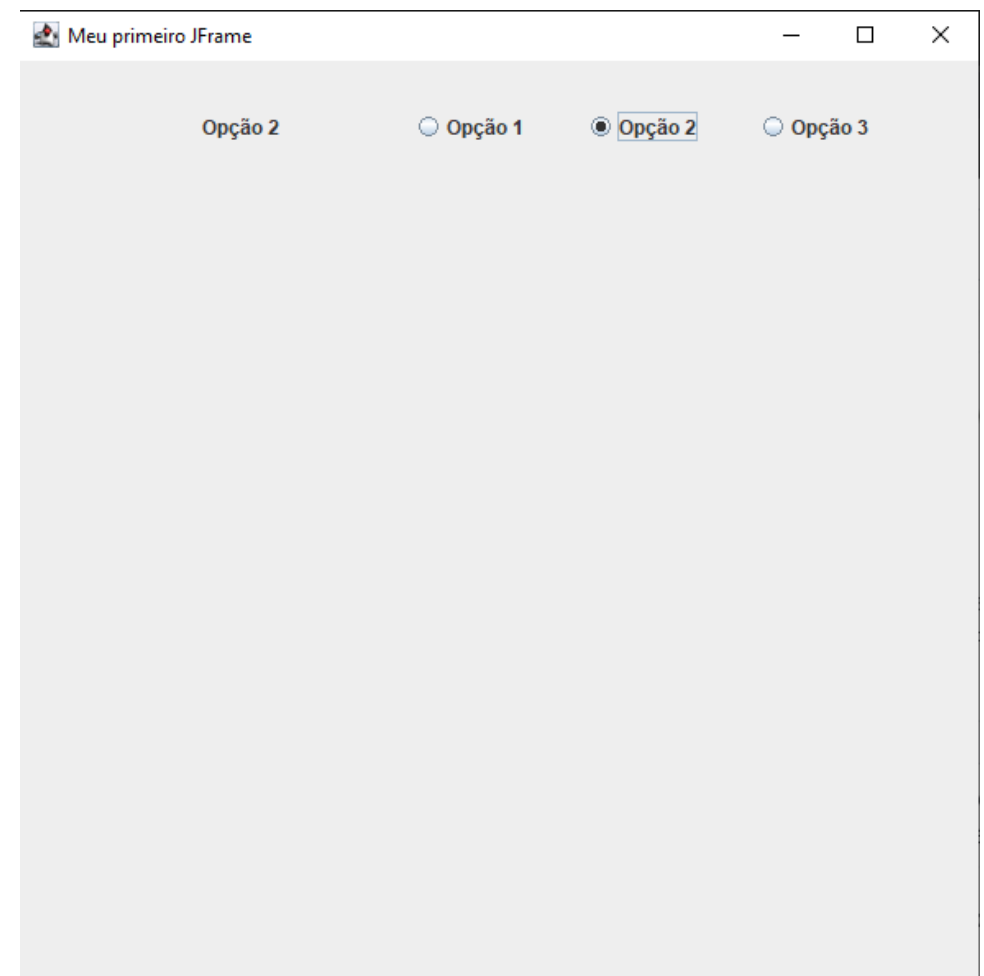
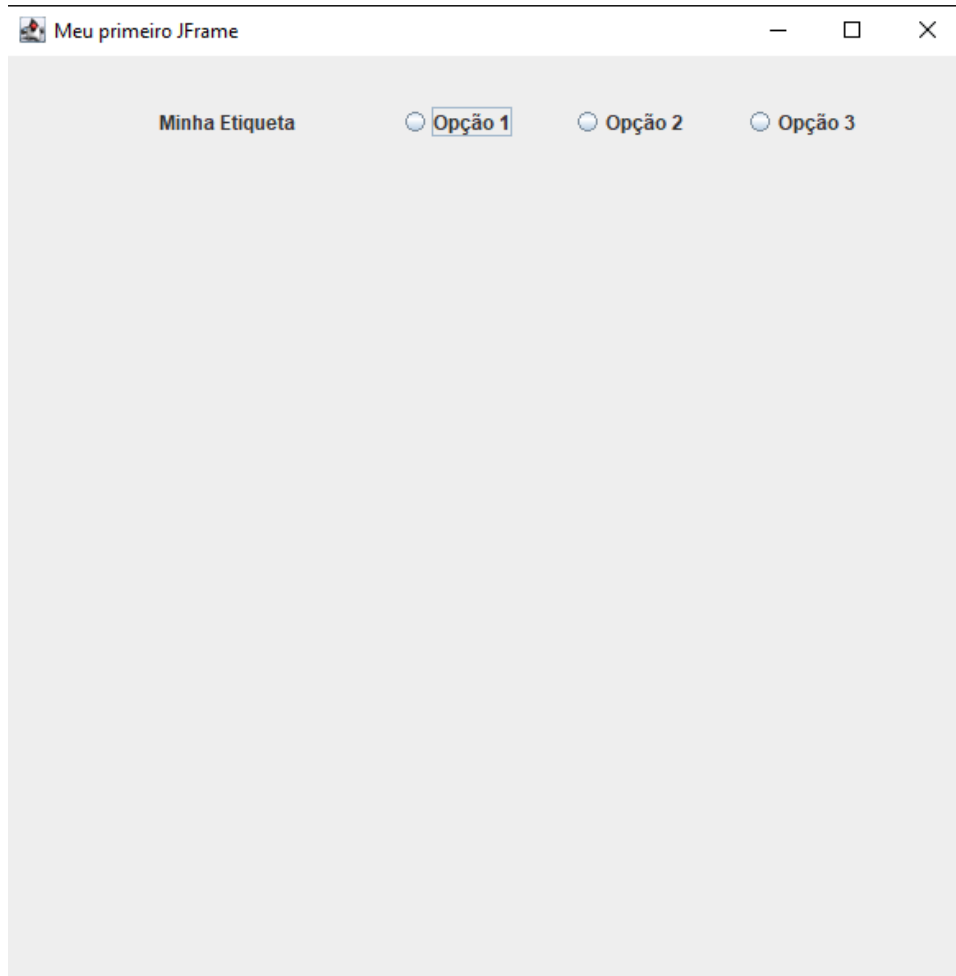
        this.jrb_opcao1 = new JRadioButton("Opção 1");
        this.jrb_opcao1.setPreferredSize(new Dimension(100,70));
        this.getContentPane().add("Center",this.jrb_opcao1);

        this.jrb_opcao2 = new JRadioButton("Opção 2");
        this.jrb_opcao2.setPreferredSize(new Dimension(100,70));
        this.getContentPane().add("Center",this.jrb_opcao2);

        this.jrb_opcao3 = new JRadioButton("Opção 3");
        this.jrb_opcao3.setPreferredSize(new Dimension(100,70));
        this.getContentPane().add("Center",this.jrb_opcao3);

        this.bg_grupo = new ButtonGroup();
        this.bg_grupo.add(jrb_opcao1);
        this.bg_grupo.add(jrb_opcao2);
        this.bg_grupo.add(jrb_opcao3);
    }
}
```

```
public void CreateEvent()
{
    this.jrb_opcao1.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            jl_etiqueta.setText(jrb_opcao1.getText());
        }
    });
    this.jrb_opcao2.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            jl_etiqueta.setText(jrb_opcao2.getText());
        }
    });
    this.jrb_opcao3.addItemListener(new ItemListener()
    {
        public void itemStateChanged(ItemEvent e)
        {
            jl_etiqueta.setText(jrb_opcao3.getText());
        }
    });
}
```



**JComboBox** representa uma caixa de seleção, da qual pode ser selecionada uma única opção.

Entre os principais métodos temos:

- ***addItem(Object)*** que adiciona um item à lista de opções;
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;
- ***setEditable(boolean)*** que permite ao usuário digitar uma opção;
- ***getSelectedIndex()*** que retorna a posição do item atualmente selecionado;
- ***getSelectedItem()*** que retorna o texto do item atualmente selecionado;
- ***setSelectedIndex(int)*** que seleciona o item da posição especificada;
- ***setSelectedIndex(Object)*** que seleciona o objeto especificado na lista.

**JComboBox** representa uma caixa de seleção, da qual pode ser selecionada uma única opção.

Entre os principais métodos temos:

## Eventos:

- ***ItemListener()*** é uma interface para receber um evento de seleção de item.
- ***itemStateChanged(ItemEvent)*** é invocado um item é selecionado ou deselecionado pelo usuário.

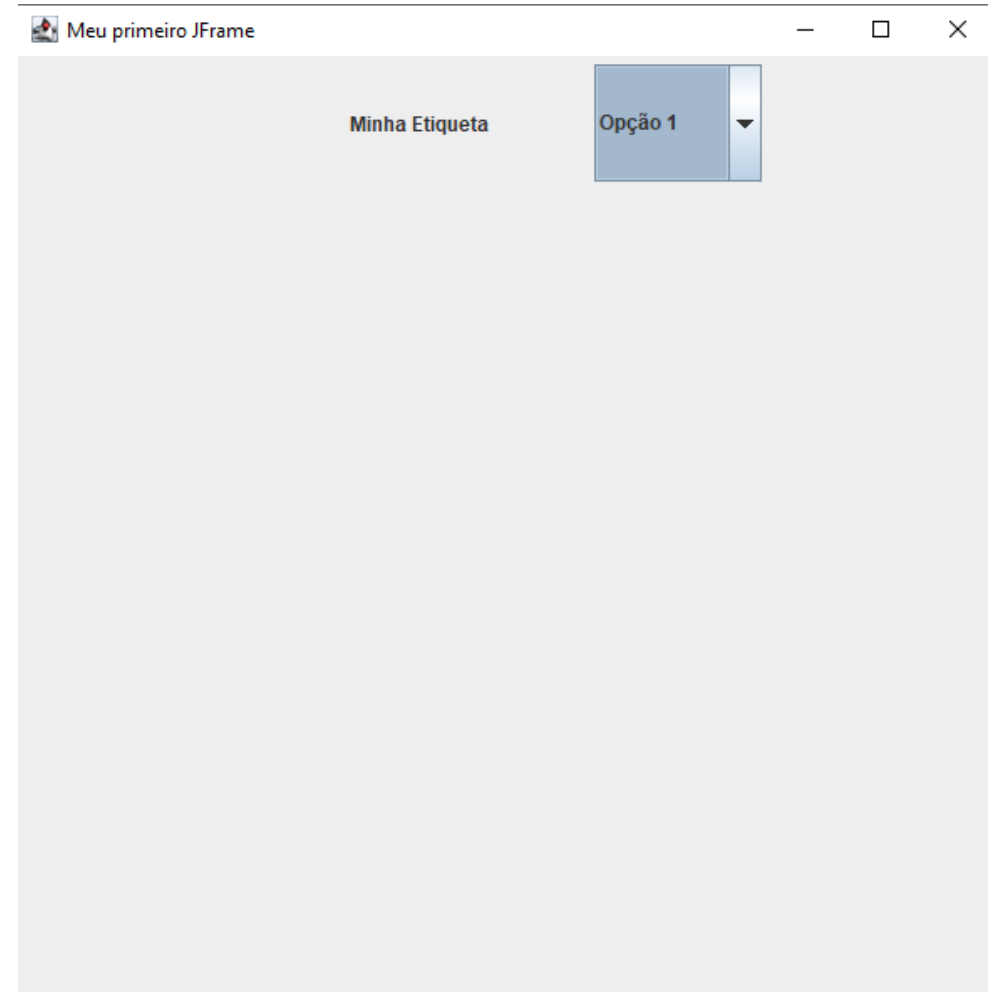


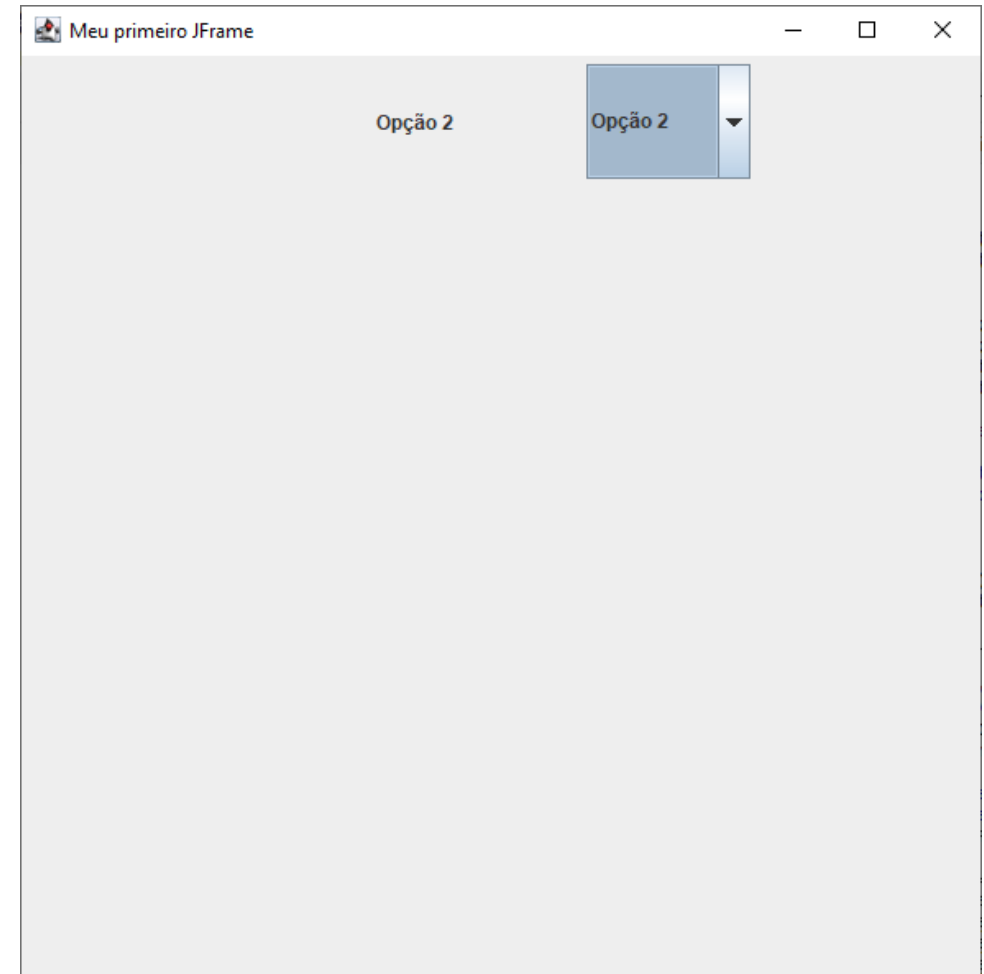
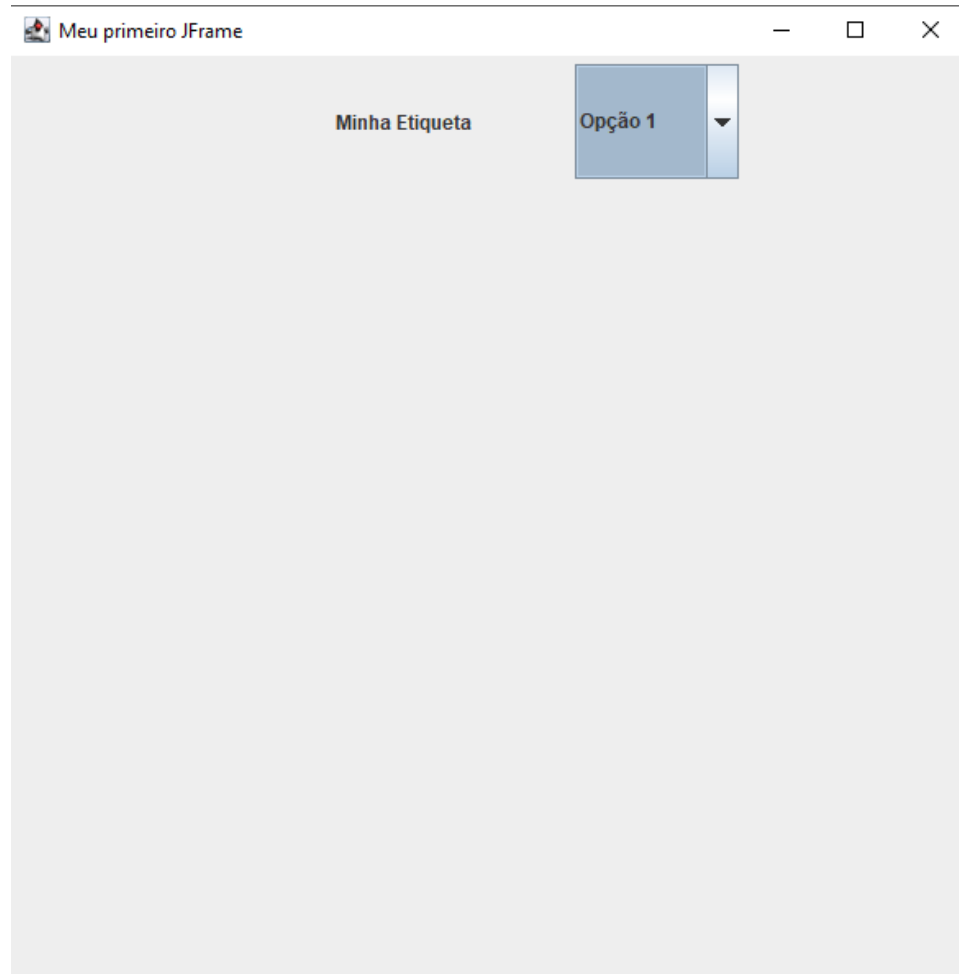
```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JComboBox jcb_combo;

    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }
    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta",SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200,70));
        this.getContentPane().add("Center",this.jl_etiqueta);

        this.jcb_combo = new JComboBox();
        this.jcb_combo.setPreferredSize(new Dimension(100,70));
        this.jcb_combo.addItem("Opção 1");
        this.jcb_combo.addItem("Opção 2");
        this.jcb_combo.addItem("Opção 3");
        this.getContentPane().add("Center",this.jcb_combo);
    }
    public void CreateEvent()
    {
        this.jcb_combo.addItemListener(new ItemListener()
        {
            public void itemStateChanged(ItemEvent e)
            {
                jl_etiqueta.setText(jcb_combo.getSelectedItem().toString());
            }
        });
    }
}
```





**JList** representa uma lista de opções que permite a seleção de mais de um item simultaneamente.

Entre os principais métodos temos:

- ***setListData(Object[])*** que preenche ou altera os itens de uma lista;
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels
- ***getSelectedValues()*** que retorna um array de objetos contendo itens selecionados na lista;
- ***getSelectedIndex()*** que retorna a posição do item atualmente selecionado;
- ***getSelectedItem()*** que retorna o texto do item atualmente selecionado;
- ***setSelectedIndex(int)*** que seleciona o item da posição especificada;
- ***setSelectedIndex(Object)*** que seleciona o objeto especificado na lista.

**JList** representa uma lista de opções que permite a seleção de mais de um item simultaneamente.

Entre os principais métodos temos:

### Eventos:

- **ListSelectionListener()** é uma interface para receber um evento de seleção de item contido em uma lista.
- **valueChanged(ListSelectionEvent)** é invocado um item de uma lista é selecionado ou deselecionado pelo usuário.

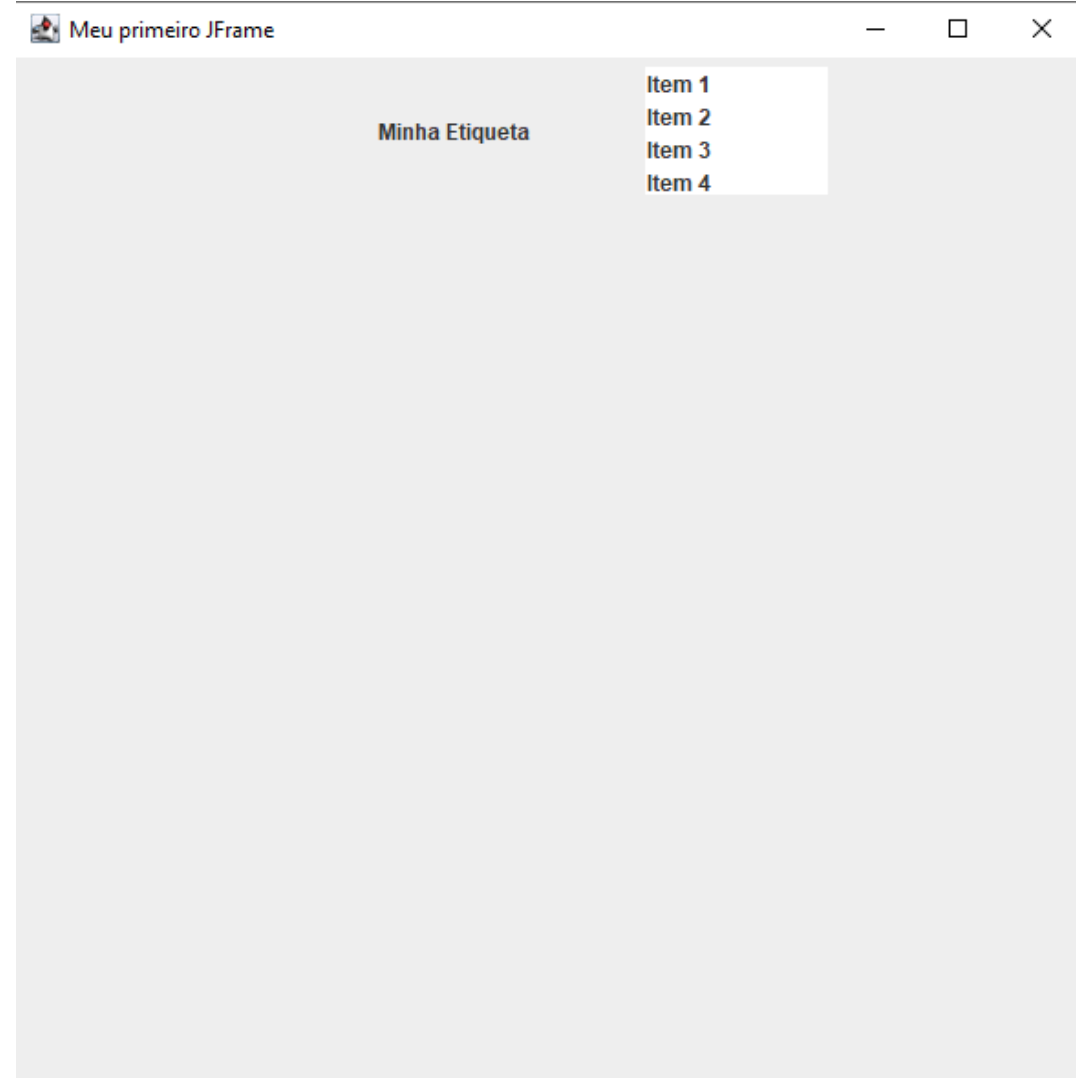
```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JList jlist_lista;
    private DefaultListModel dlm_modelo;

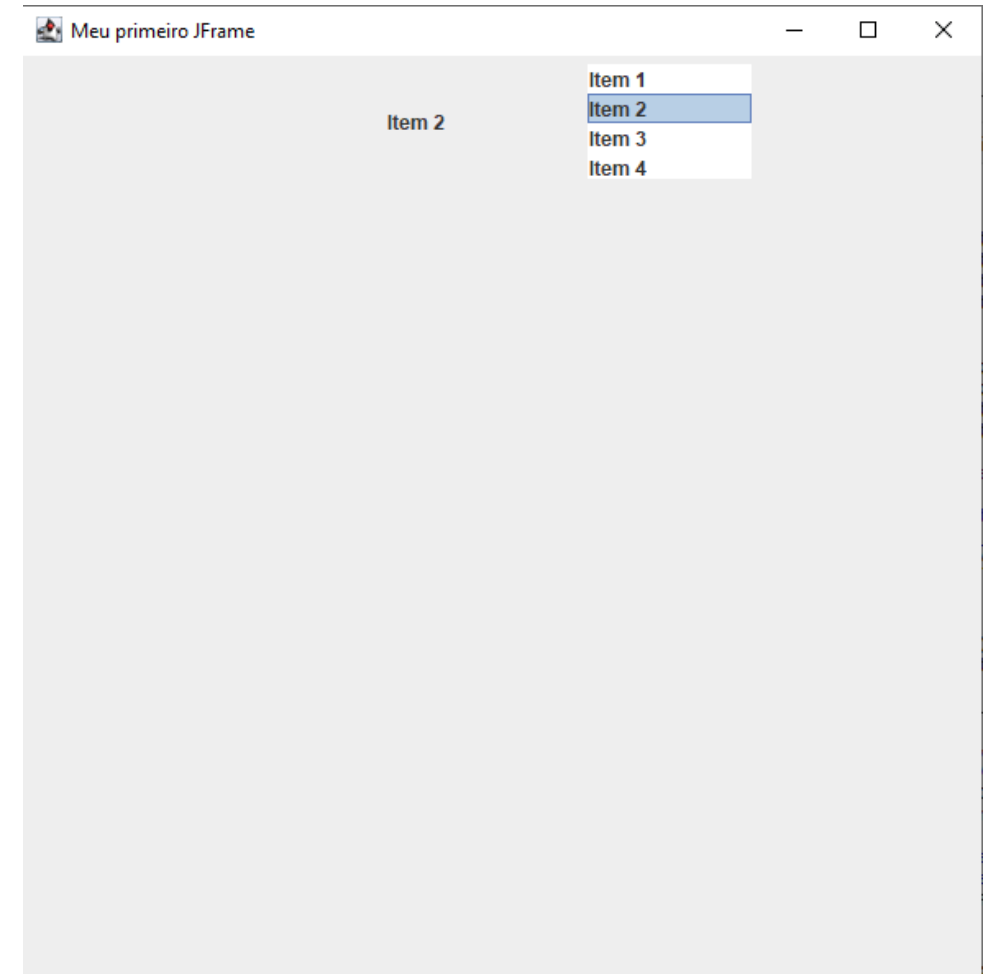
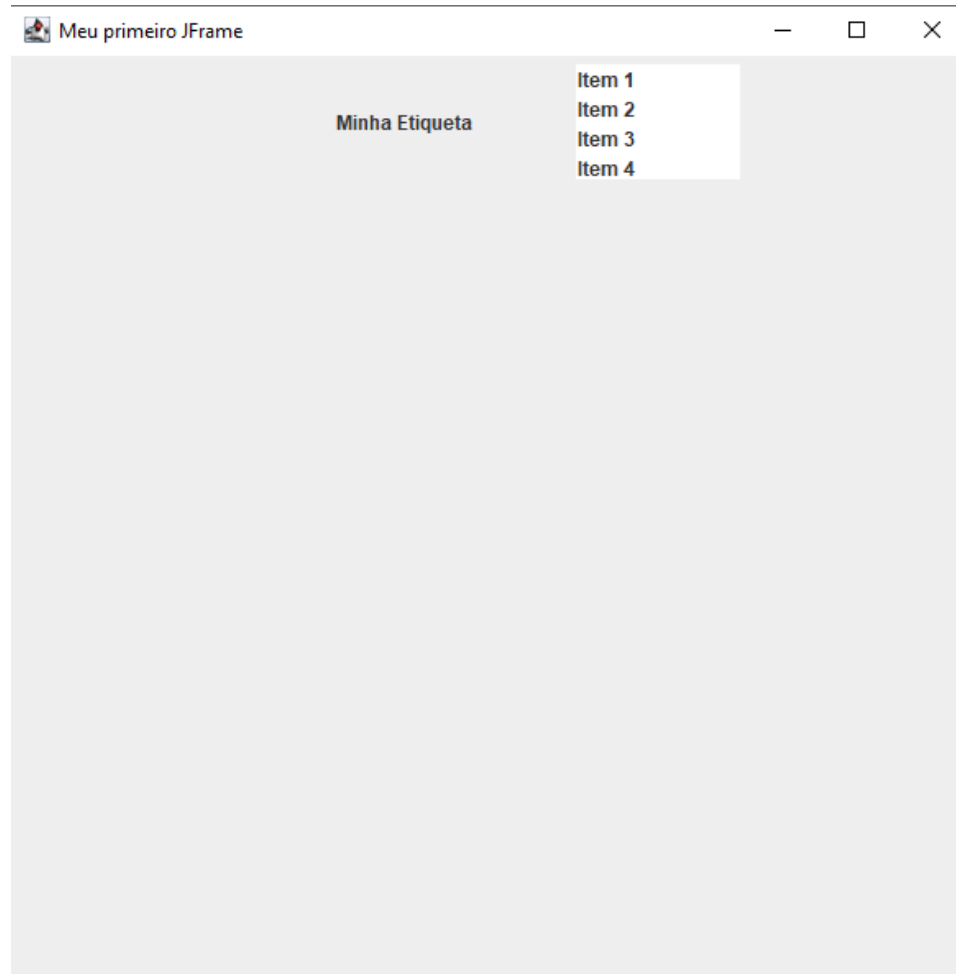
    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }
    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

        this.jl_etiqueta = new JLabel("Minha Etiqueta", SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200, 70));
        this.getContentPane().add("Center", this.jl_etiqueta);

        this.dlm_modelo = new DefaultListModel<String>();
        this.dlm_modelo.add(0, "Item 1");
        this.dlm_modelo.add(1, "Item 2");
        this.dlm_modelo.add(2, "Item 3");
        this.dlm_modelo.add(3, "Item 4");

        this.jlist_lista = new JList();
        this.jlist_lista.setPreferredSize(new Dimension(100, 70));
        this.jlist_lista.setModel(this.dlm_modelo);
        this.getContentPane().add("Center", this.jlist_lista);
    }
    public void CreateEvent()
    {
        this.jlist_lista.addListSelectionListener(new ListSelectionListener()
        {
            public void valueChanged(ListSelectionEvent listSelectionEvent)
            {
                jl_etiqueta.setText(jlist_lista.getSelectedValue().toString());
            }
        });
    }
}
```





**JButton** representa um botão destinado a executar uma ação. Entre os principais métodos temos:

- ***setText(String)*** que insere um texto padrão ao campo (o texto também pode ser inserido diretamente no método construtor do objeto);
- ***setSize(int, int)*** que define as medidas largura e altura da etiqueta em pixels;
- ***setIcon(Icon)*** que altera o ícone do botão.

## Eventos:

- ***ActionListener()*** é uma interface para receber qualquer evento de ação sobre um objeto.
- ***actionPerformed(ActionEvent)*** é invocado quando uma ação ocorre.

```
public class MeuJFrame extends JFrame
{
    private JLabel jl_etiqueta;
    private JButton jb_botao;

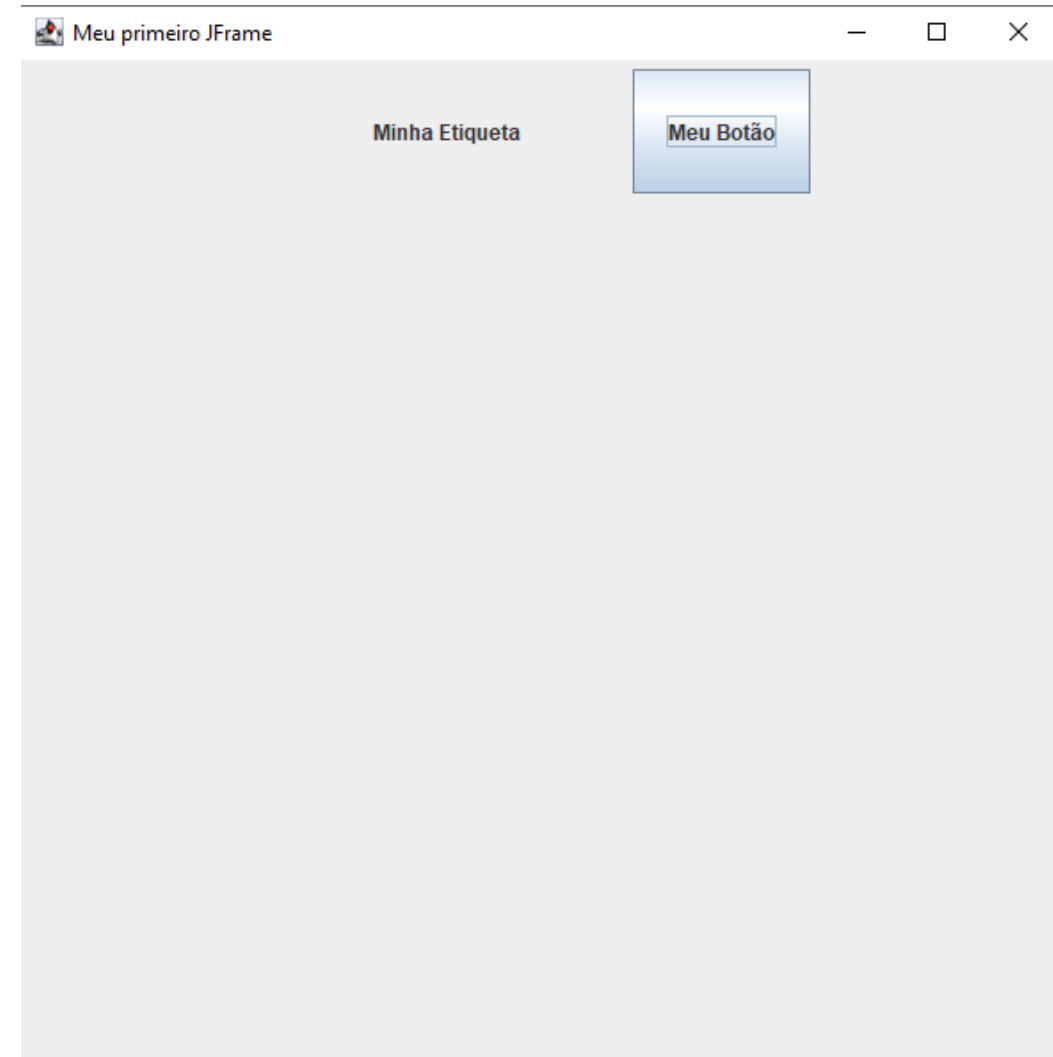
    public MeuJFrame()
    {
        this.CreateGUI();
        this.CreateEvent();
    }

    public void CreateGUI ()
    {
        this.setSize(new Dimension(600, 600));
        this.setTitle("Meu primeiro JFrame");
        this.setDefaultCloseOperation(EXIT_ON_CLOSE);
        this.setLayout(new FlowLayout());

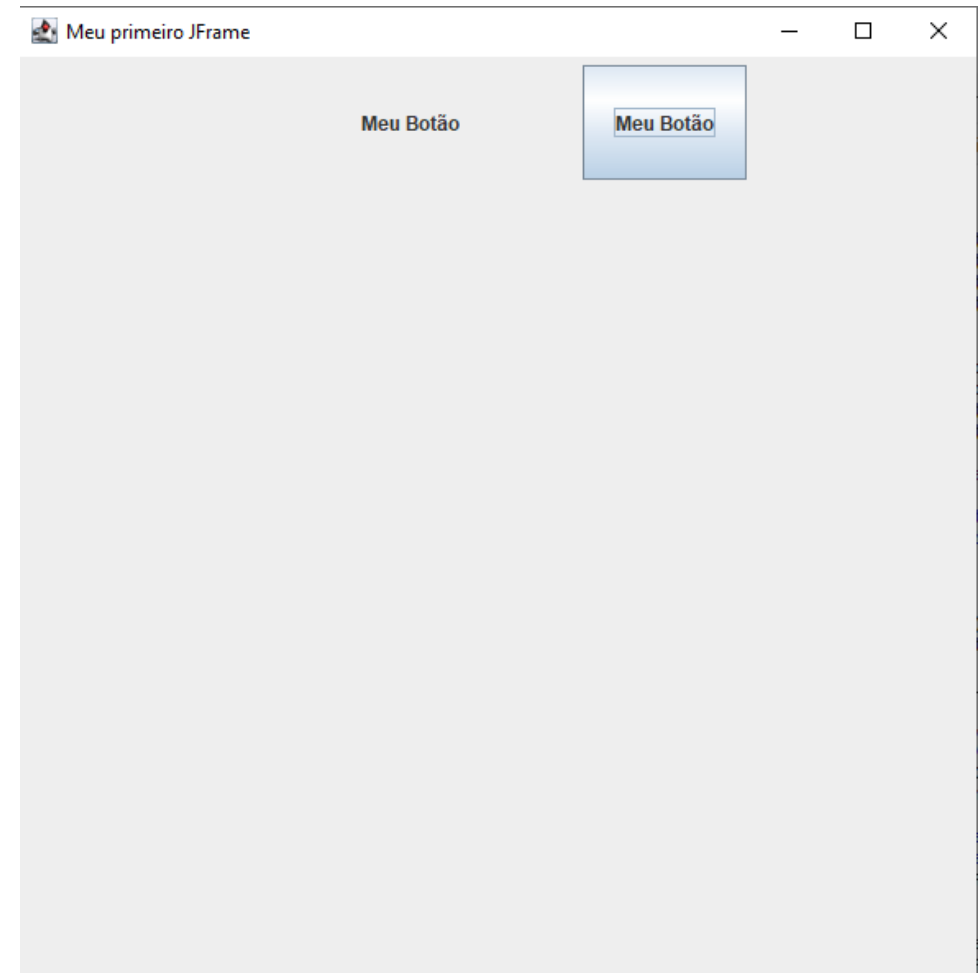
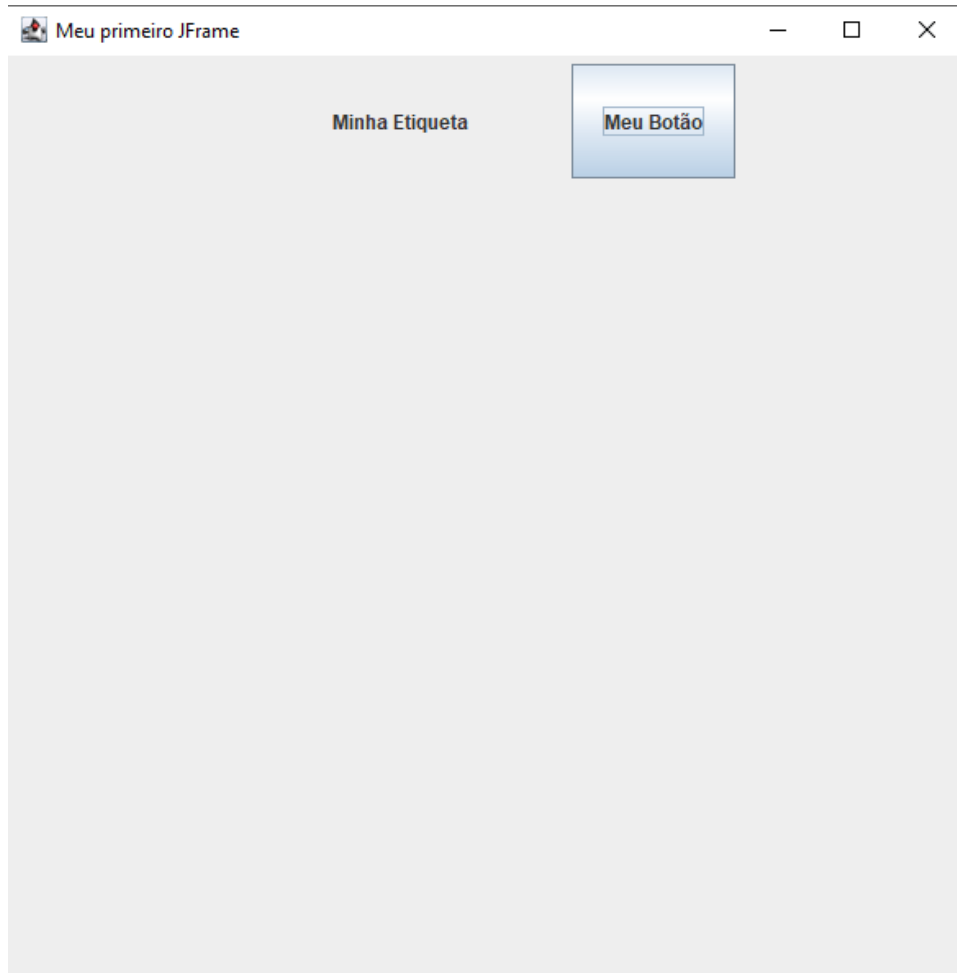
        this.jl_etiqueta = new JLabel("Minha Etiqueta",SwingConstants.CENTER);
        this.jl_etiqueta.setPreferredSize(new Dimension(200,70));
        this.getContentPane().add("Center",this.jl_etiqueta);

        this.jb_botao = new JButton("Meu Botão");
        this.jb_botao.setPreferredSize(new Dimension(100,70));
        this.getContentPane().add("Center",this.jb_botao);
    }

    public void CreateEvent()
    {
        this.jb_botao.addActionListener(new ActionListener()
        {
            public void actionPerformed(ActionEvent e)
            {
                jl_etiqueta.setText(jb_botao.getText());
            }
        });
    }
}
```







# OBRIGADO