

Lista de Exercícios UIA 3 e 4

Prof. Rubens Laurini

Exercícios 1. Crie a interface gráfica a seguir utilizando os componentes do SWING – JFrame, JLabel, JPanel, JTextField, JTextArea, para o desenvolvimento siga as instruções para os nomes dos componentes e as propriedades pedidas.

- a. Crie uma classe com o nome Matricula;
- b. Utilize o componente JFrame para a criação da Janela, nesse exercício o componente JFrame deve ser declarado como uma variável global e ser instanciado como um componente qualquer;
- c. Declare 1 objeto global do tipo JFrame como o nome frame (ele será a janela principal) 2 objetos globais do tipo JPanel (painelBranco, painelAmarelo) e 5 objetos globais do tipo JLabel (lblTitulo, lblNome, lblCurso, lblDisciplina, lblComentario); 3 objetos globais do tipo JTextField (tfNome, tfcurso, tfDisciplina). 1 objeto global JTextArea como o nome taComentario.
- d. Agora ajuste as seguintes propriedades para cada componente:
 - **frame**: setar o título (setTitle("Utilizando o JtextComponent")); setar o tamanho (pack()); setar a visibilidade (setVisible(true); lembre-se de delcarar um container intermediário e associá-lo ao container frame, utilizando o método getContentPane(), setar o Layout para nulo;
 - **painelBranco**: setLayout(null); ajustar o localização e o tamanho utilizando o método setBounds(10, 10, 510, 340); setBackground(Color.White); e adicionar ao container intermediário.
 - **lblTitulo**: ajustar o que deve ser escrito nesse JLabel, utilizando o étodo setText("Matrícula em Disciplina"); para ajustar a cor da fonte, utilize o método setForeground(Color.blue); para ajustar a fonte utilizamos o método .setFont(new Font("Arial",Font.BOLD|Font.ITALIC,18), para ajustar localização e o tamanho utilize o método setBounds(170, 10, 210, 30); Adicionar ao painelBranco.
 - **lblNome**: ajustar o que deve ser escrito nesse JLabel, utilizando o método setText("Nome do aluno: "); para ajustar a localização e o tamanho utilize o método setBounds(20, 60, 115, 15); Adicionar ao painelBranco.
 - **lblCurso**: ajustar o que deve ser escrito nesse JLabel, utilizando o método setText("Curso: "); para ajustar localização e o tamanho utilize o método setBounds(20, 90, 50, 15); Adicionar ao painelBranco.
 - **lblDisciplina**: ajustar o que deve ser escrito nesse JLabel, utilizando o método setText("Disciplina: "); para ajustar localização e o tamanho utilize o método setBounds(20, 130, 110, 15); Adicionar ao painelBranco.

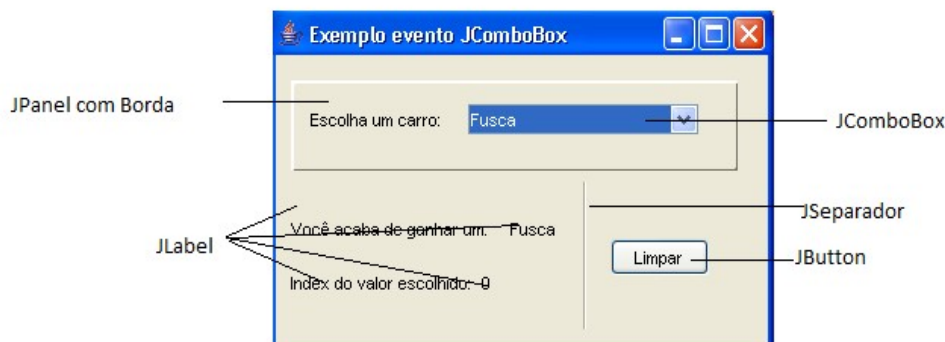
- **tfNome:** ajustar o que deve ser escrito nesse JTextField, utilizando o método `setText("Digite o seu nome!")`; ajustar a cor da letra, através do método `setForeground(Color.blue)`; para ajustar localização e o tamanho utilize o método `setBounds(100, 60, 390, 20)`; Adicionar ao `painelBranco`.
- **tfCurso:** ajustar o que deve ser escrito nesse JTextField, utilizando o método `setText("Sistemas de Informação")`; ajustar a cor da letra, através do método `setForeground(Color.blue)`; para ajustar localização e o tamanho utilize o método `setBounds(60, 90, 430, 20)`; Adicionar ao `painelBranco`.
- **tfDisciplina:** ajustar o que deve ser escrito nesse JTextField, utilizando o método `setText("Linguagens e Técnicas de Programação")`; ajustar a cor da letra, através do método `setForeground(Color.blue)`; para ajustar localização e o tamanho utilize o método `setBounds(70, 130, 420, 20)`; Adicionar ao `painelBranco`.
- **painelAmarelo:** `setLayout(null)`; ajustar o localização e o tamanho utilizando o método `setBounds(20, 180, 470, 130)`; `setBackground(Color.yellow)`; note que esse painel possui uma borda, para ajustar bordas em painéis utilizamos o método `setBorder` e escolhemos o tipo de borda, esse tipo é passado como parâmetro `setBorder(BorderFactory.createEtchedBorder())`; adicionar ao `painelBranco`.
- **lblComentario:** ajustar o que deve ser escrito nesse JLabel, utilizando o método `setText("Comentários: ")`; para ajustar localização e o tamanho utilize o método `setBounds(20, 10, 70, 15)`; Adicionar ao `painelAmarelo`.
- **taComentario:** ajustar o que deve ser escrito nesse JLabel, utilizando o método `setText("O Intuito da disciplina é ensinar ao aluno o paradigma orientado a objetos, setText ("O Intuito da disciplina é ensinar ao aluno o paradigma orientado a objetos, utilizando a linguagem de programação orientada a objetos JAVA, também apresentar os componentes para construção de interface gráfica, o SWING.")`; ajustar a cor da letra, através do método `setForeground(Color.blue)`; para fazer com que o componente JTextArea possua quebra de linha é necessário ajustar o método - `setLineWrap(true)`; também para evitar que a palavra fique dividida entre duas linhas, utilizamos o método - `setWrapStyleWord(true)`; para ajustar localização e o tamanho utilize o método `setBounds(20, 30, 430, 87)`; Adicionar ao `painelAmarelo`.

Exercícios 2. Crie outra classe de interface gráfica a seguir utilizando os componentes do SWING – JFrame, JLabel, JPanel, JTextField, JPasswordField e JButton, para o desenvolvimento siga as instruções para os nomes dos componentes e as propriedades pedidas.

<p>DICAS: Para utilizar botões em java o componente é denominado JButton</p> <pre>//declara instancia e ajusta o texto do botão JButton botao = new JButton ("ok"); //adiciona o JButton ao Container cnt.add(botao);</pre>

- JFrameLogin: utilize o pack(); Ajuste o Container intermediário para setLayout(null);
- lblTitulo: setBounds(90, 20, 180, 15); Font("Arial", Font.BOLD, 12); setForeground(Color.red); adicionar ao Container.
- lblLogin: setBounds(30, 50, 29, 20); setText("Insira seu Login!"); adicionar ao Container.
- lblPassword: setBounds(20, 90, 53, 15); adicionar ao Container.
- jTFLogin: setBounds(80, 50, 210, 20); adicionar ao Container.
- jPFPassword: setBounds(80, 90, 210, 20); adicionar ao Container, lembre-se de utilizar o componente JPasswordField.
- jPanelBotoes: setBounds(30, 120, 280, 60); setLayout (null); setBackground(Color.blue); adicionar ao Container.
- btnOk: setBounds(70, 20, 47, 23); Adicionar ao jPanelButoes.
- btnCancelar: setBounds(150, 20, 75, 23); setToolTipText("Sair do Programa"); Adicionar ao jPanelButoes.

3. No Netbeans, implementar o programa a seguir. O programa deverá ter uma classe com o nome FrmCombo e deverá apresentar todos os componentes da figura abaixo.



DICA 1: Para utilizar comboBox em Java, o componente é denominado JComboBox utilizaremos um vetor para os dados que queremos dentro do combo, neste exercício os dados serão (fusca, palio, civic)

```
//declaração do vetor de string
String[] dados = {"fusca", "palio", "civic"}
JComboBox combo = new JComboBox(dados);
Combo.setBounds(10,10,200,30);
cnt.add(combo);
```

DICA 2: Inicialmente vc deve utilizar apenas 3 objetos de tipo JLabel, o conteúdo dos objetos JLabel "Fusca e 0", deverão ser utilizados com o tratamento de eventos

DICA 3: Você deve pesquisar na Internet como faz para o JPanel ficar com esta borda conforme figura e também como utilizar o componente JSeparator.