

USJT – 2018/1 - Programação Orientada a Objetos - Laboratório de Exercícios

Professora: Machion

### Aula: 10

Assunto: Interface Gráfica

**Exemplo Resolvido:** Junto com seu professor, abra o código abaixo, fornecido junto com o exercício, no JGrasp; seu professor dar uma explicação geral do código, entenda-o e depois execute-o para ver os resultados.

*Fazer uma tela que para a hierarquia de classes Ponto, Circulo e Cilindro da aula passada. A tela deve decidir qual é a forma pelos parâmetros de entrada, instanciar esta forma e mostrar seus dados por meio do toString e um JOptionPane. Depois deve calcular a área e o perímetro, se for um Circulo, ou a área, o perímetro, o volume e a área de superfície, se for um Cilindro. A tela deve ter um botão limpar, que limpa todos os campos, e um botão sair.*

```
import java.awt.BorderLayout;
import java.awt.Container;
import java.awt.FlowLayout;
import java.awt.GridLayout;
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JButton;
import javax.swing.JFrame;
import javax.swing.JLabel;
import javax.swing.JOptionPane;
import javax.swing.JPanel;
import javax.swing.JTextField;

public class Tela extends JFrame implements ActionListener {
    private JTextField txtCoordenadaX, txtCoordenadaY, txtRaio, txtAltura,
        txtArea, txtPerimetro, txtVolume, txtSuperficie;
    private JLabel lblCoordenadaX, lblCoordenadaY, lblRaio, lblAltura, lblArea,
        lblPerimetro, lblVolume, lblSuperficie;
    private JButton btnCalcula, btnLimpa, btnSai;

    public Tela() {
        super("Formas");

        // montando o painel dos dados de entrada
        JPanel painelDadosEntrada = new JPanel(new GridLayout(4, 2));
        JPanel painelEntrada = new JPanel(new FlowLayout());
        txtCoordenadaX = new JTextField(10);
        txtCoordenadaY = new JTextField(10);
        txtRaio = new JTextField(10);
        txtAltura = new JTextField(10);
        lblCoordenadaX = new JLabel("CoordenadaX:");
        lblCoordenadaY = new JLabel("CoordenadaY:");
```

```

lblAltura = new JLabel("Altura:");
lblRaio = new JLabel("Raio:");
painelDadosEntrada.add(lblCoordenadaX);
painelDadosEntrada.add(txtCoordenadaX);
painelDadosEntrada.add(lblCoordenadaY);
painelDadosEntrada.add(txtCoordenadaY);
painelDadosEntrada.add(lblRaio);
painelDadosEntrada.add(txtRaio);
painelDadosEntrada.add(lblAltura);
painelDadosEntrada.add(txtAltura);
painelEntrada.add(painelDadosEntrada);
// montando o painel dos dados de saida
JPanel painelDadosSaida = new JPanel(new GridLayout(4, 2));
JPanel painelSaida = new JPanel(new FlowLayout());
lblArea = new JLabel("Area do Circulo: ");
txtArea = new JTextField(10);
lblPerimetro = new JLabel("Perimetro do Circulo: ");
txtPerimetro = new JTextField(10);
lblVolume = new JLabel("Volume do Cilindro: ");
txtVolume = new JTextField(10);
lblSuperficie = new JLabel("Area de Superficie do Cilindro: ");
txtSuperficie = new JTextField(10);
painelDadosSaida.add(lblArea);
painelDadosSaida.add(txtArea);
painelDadosSaida.add(lblPerimetro);
painelDadosSaida.add(txtPerimetro);
painelDadosSaida.add(lblVolume);
painelDadosSaida.add(txtVolume);
painelDadosSaida.add(lblSuperficie);
painelDadosSaida.add(txtSuperficie);
painelSaida.add(painelDadosSaida);
// montando o painel dos botoes
JPanel painelBotoes = new JPanel(new FlowLayout());
btnCalcula = new JButton("Calcular");
btnLimpa = new JButton("Limpar");
btnSai = new JButton("Sair");
painelBotoes.add(btnCalcula);
painelBotoes.add(btnLimpa);
painelBotoes.add(btnSai);

// montando a tela toda
Container caixa = getContentPane();
caixa.setLayout(new BorderLayout());
caixa.add(painelEntrada, BorderLayout.WEST);
caixa.add(painelSaida, BorderLayout.EAST);
caixa.add(painelBotoes, BorderLayout.SOUTH);

// atribuir listener aos botoes
btnCalcula.addActionListener(this);
btnLimpa.addActionListener(this);
btnSai.addActionListener(this);

// arremates finais
setSize(710, 190);
setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
setLocationRelativeTo(null);
setVisible(true);

```

```

}

```

```

public void actionPerformed(ActionEvent e) {
    // depois iremos aprender a modularizar melhor
    if (e.getSource() == btnCalcula) {

```

```

String coordenadaX = txtCoordenadaX.getText();
String coordenadaY = txtCoordenadaY.getText();
String raio = txtRaio.getText();
String altura = txtAltura.getText();
if (coordenadaX.length() > 0 && coordenadaY.length() > 0
    && raio.length() > 0 && altura.length() > 0) {
    // e um cilindro
    Cilindro cilindro = new Cilindro();
    cilindro.setCoordenadaX(Double.parseDouble(coordenadaX));
    cilindro.setCoordenadaY(Double.parseDouble(coordenadaY));
    cilindro.setRaio(Double.parseDouble(raio));
    cilindro.setAltura(Double.parseDouble(altura));
    // calcular usando a heranca
    txtArea.setText("" + cilindro.area());
    txtPerimetro.setText("" + cilindro.perimetro());
    txtVolume.setText("" + cilindro.volume());
    txtSuperficie.setText("" + cilindro.areaDaSuperficieExterna());
    // estou usando o this em vez de null para travar a tela de
    // baixo enquanto nao fechar o dialogo
    JOptionPane.showMessageDialog(this, cilindro, "Formas",
        JOptionPane.INFORMATION_MESSAGE);
} else if (coordenadaX.length() > 0 && coordenadaY.length() > 0
    && raio.length() > 0) {
    // e um circulo
    Circulo circulo = new Circulo();
    circulo.setCoordenadaX(Double.parseDouble(coordenadaX));
    circulo.setCoordenadaY(Double.parseDouble(coordenadaY));
    circulo.setRaio(Double.parseDouble(raio));
    // calcular
    txtArea.setText("" + circulo.area());
    txtPerimetro.setText("" + circulo.perimetro());
    txtVolume.setText("");
    txtSuperficie.setText("");
    JOptionPane.showMessageDialog(this, circulo, "Formas",
        JOptionPane.INFORMATION_MESSAGE);
} else if (coordenadaX.length() > 0 && coordenadaY.length() > 0) {
    // e um ponto
    Ponto ponto = new Ponto();
    ponto.setCoordenadaX(Double.parseDouble(coordenadaX));
    ponto.setCoordenadaY(Double.parseDouble(coordenadaY));
    txtArea.setText("");
    txtPerimetro.setText("");
    txtVolume.setText("");
    txtSuperficie.setText("");
    JOptionPane.showMessageDialog(this, ponto, "Formas",
        JOptionPane.INFORMATION_MESSAGE);
} else {
    JOptionPane.showMessageDialog(this, "Dados invalidos.",
        "Formas", JOptionPane.ERROR_MESSAGE);
}
} else if (e.getSource() == btnLimpa) {
    txtCoordenadaX.setText("");
    txtCoordenadaY.setText("");
    txtRaio.setText("");
    txtAltura.setText("");
    txtArea.setText("");
    txtPerimetro.setText("");
    txtVolume.setText("");
    txtSuperficie.setText("");
} else {
    System.exit(0);
}
}

```

```
}
```

```
public class Teste{  
    public static void main(String[] args){  
        new Tela();  
    }  
}
```

### Problemas Propostos:

#### Exercícios iniciais: valor 0,5 ponto

Resolva os exercícios desta seção para conquistar 0,5 ponto

1) Crie uma tela com um campo texto com a etiqueta *Texto*, um botão *Mostrar*, um botão *Limpar* e um botão *Sair*. O botão *Mostrar*, ao ser clicado, mostra o conteúdo do campo *Texto* em um JOptionPane. O botão *Limpar* limpa o campo *Texto* e o botão *Sair* sai do programa.

#### Exercícios intermediários: valor 0,5 ponto

Resolva os exercícios desta seção para conquistar mais 0,5 ponto

2) Use as classes Relógio e Mostrador da aula 06. Crie uma tela que apresenta os mostrador do relógio (hora e minuto) em um JLabel e que tenha três botões: TicTac, que aumenta um minuto a cada clique, Hora, que acerta a Hora (0 a 23) e Minuto, que acerta o minuto (0 a 59). Use sempre o método mostra() da classe relógio para atualizar o mostrador da tela. Consulte a documentação do Java com seu professor para aprender a aumentar o tamanho e a cor da fonte.

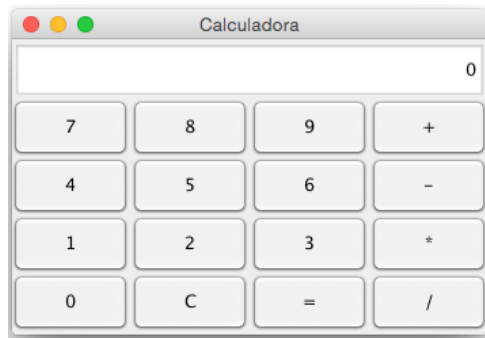
#### Exercícios complementares (para praticar)

Resolva os exercícios desta seção para aprimorar seus conhecimentos

3 ) De modo análogo ao Exemplo Resolvido, crie uma tela para a hierarquia de classes Empregado, Mensalista, Comissionado, Horista e Tarefeiro.

4 ) Crie uma tela para a hierarquia de classes Pessoa Física, Desempregado, Empregado, Mensalista, Comissionado, Horista e Tarefeiro.

5) Faça uma calculadora com as 4 operações básicas, com botões de 0 a 9 para a entrada dos dígitos e um único mostrador que mostra os números digitados e o resultado das operações. Aceite apenas inteiros e faça as operações com inteiros. Use a imagem abaixo como exemplo de tela:



### **Bibliografia**

BARNES, David J.; KÖLLING, Michael. **Programação orientada a objetos com java**: uma introdução prática usando o BlueJ . 4. ed. São Paulo: Pearson Prentice Hall, 2009.

DEITEL, P. DEITEL, H. **Java: como programar**. 8 Ed. São Paulo: Prentice – Hall (Pearson), 2010.

LOPES, ANITA. GARCIA, GUTO. **Introdução à Programação: 500 algoritmos resolvidos**. Rio de Janeiro: Elsevier, 2002.