

# Programação Orientada a Objetos

## Semana 09 Interface Gráfica

## Reflexão

“Um dos maiores danos que se pode causar a uma criança é levá-la a perder a confiança na sua própria capacidade de pensar.”

Emilia Ferreiro

## Interface Gráfica

Conhecida pela sigla GUI (Graphical User Interface).

Permite que o usuário possa utilizar todos os recursos do sistema interagindo com telas ou ainda outros dispositivos, como mouse, teclado, microfone etc.

Constitui-se em um combinado de hardware e software necessário para permitir a comunicação entre o usuário e a aplicação

## Interface Gráfica

Uma interface gráfica em Java é baseada em dois elementos:

- ✓ **Containers** - servem para agrupar e exibir outros componentes
- ✓ **Componentes** - botões, labels, scrollbars, etc.

Dessa forma, todo programa que ofereça uma interface vai possuir pelo menos um container, que pode ser:

- ✓ JFrame: janela principal do programa
- ✓ JPanel: um painel que abriga outros componentes
- ✓ JDialog: janela para diálogos
- ✓ JApplet: janela para Applets

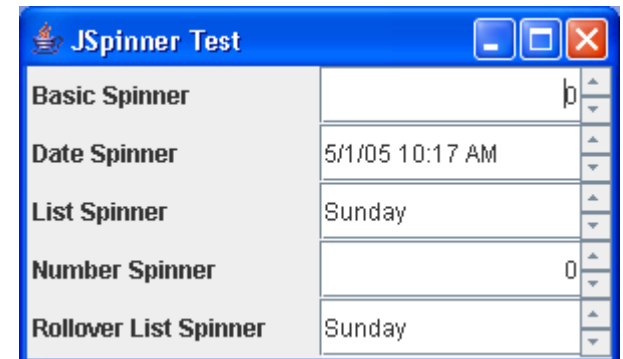
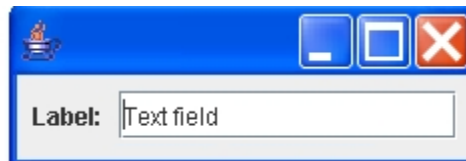
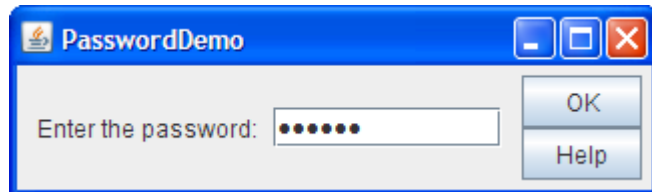
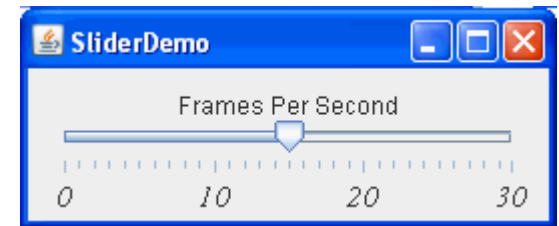
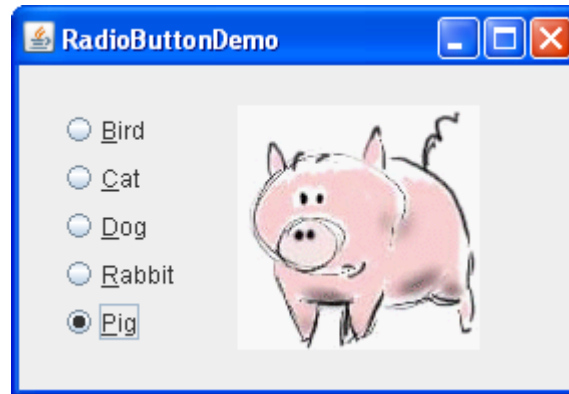
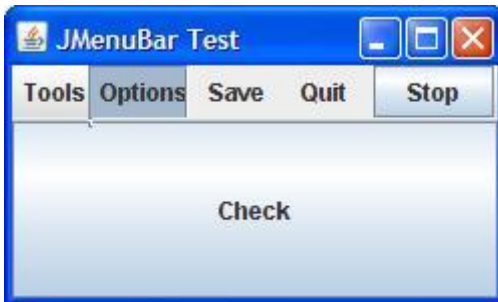
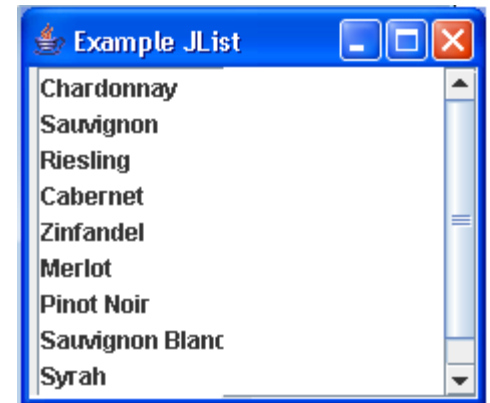
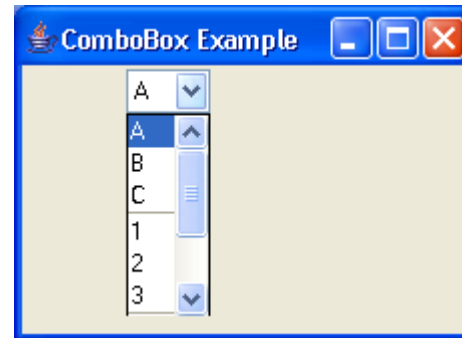
## Como implementar?

**AWT** – Abstract Window Toolkit (java.awt.\*): API básica para o desenvolvimento de GUIs e applets em Java

**Swing** (javax.swing.\*) – Extensão da API básica com inclusão de componentes visuais de mais alto nível.

**SWT** - Standard Widget Toolkit (org.eclipse.swt.\*): Biblioteca desenvolvida pela IBM como parte da plataforma Eclipse; entretanto, seu uso não está amarrado ao uso da IDE

# Controles Básicos

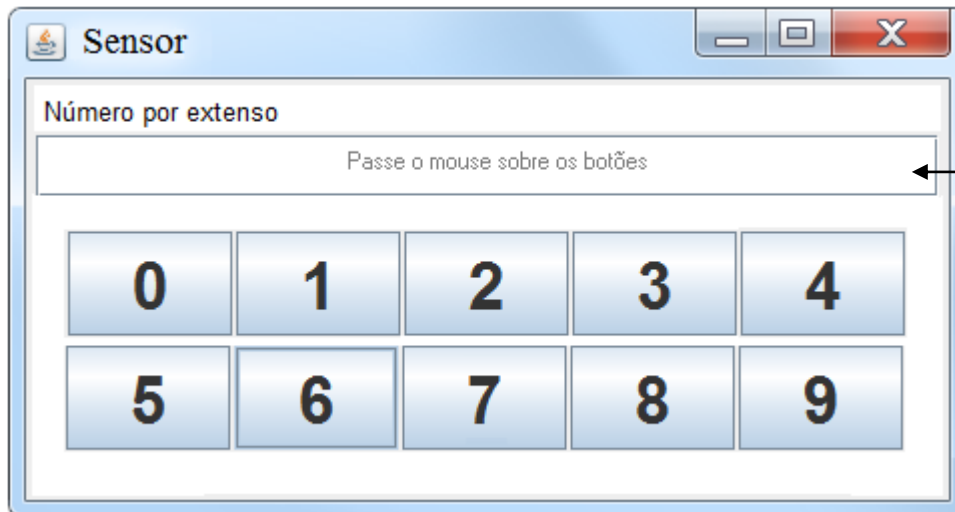


## Padrão para criar aplicativos com telas

1. Definição de todos os componentes
2. O construtor invoca a definição das propriedades dos componentes e as interações do usuário.
3. O método inicializarComponentes()  
Define a “cara” de todos os componentes da tela
4. O método definirEventos()  
Controla a interação com os eventos do usuário

## Exercícios

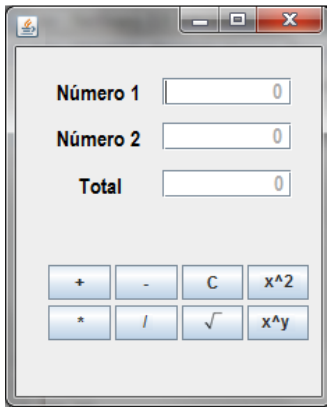
1. Crie uma aplicação que implemente uma janela contendo Buttons com números de 0 a 10 (conforme a figura abaixo), ao passar o mouse sobre os botões numéricos, seu extenso correspondente deve aparecer na caixa de texto





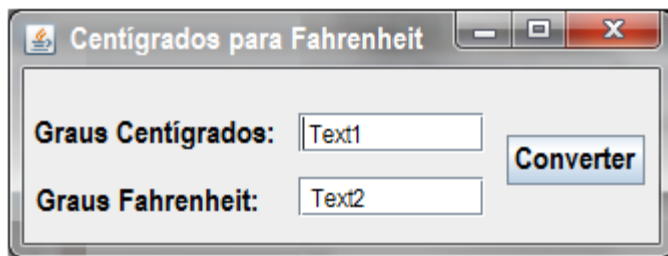
## Exercícios

2. Crie uma aplicação que implemente uma janela conforme figura abaixo:



Obs. Os botões de Raiz e quadrado atuam apenas sobre o número 1. O botão C (clear) limpa o conteúdo das caixas de texto. O botão X elevado a Y eleva o número 1

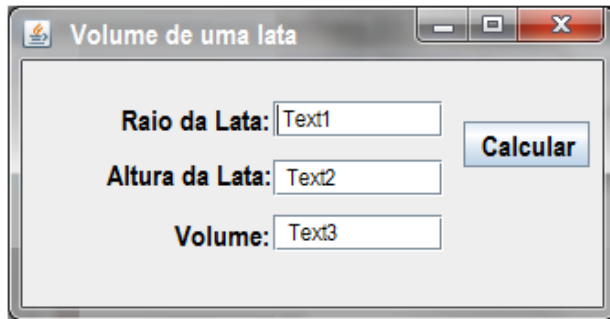
3. Crie uma aplicação que implemente uma janela para conversão de temperatura, a aplicação deve ler uma temperatura em graus Centígrados e apresentá-la convertida em graus Fahrenheit. A fórmula de conversão é:  $F = (9 * C + 160) / 5$ . Onde F é a temperatura em Fahrenheit e C é a temperatura em Centígrados.



## Exercícios

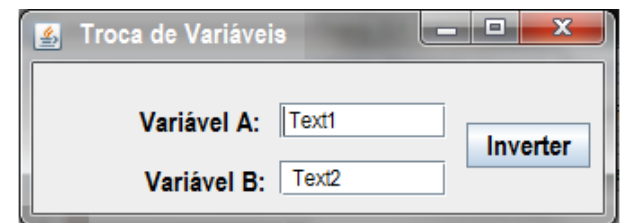
4. Crie uma aplicação para calcular e apresentar o valor do volume de uma lata de óleo, utilizando a fórmula :  $VOLUME = 3.14159 * R^2 * ALTURA$ .

Construa a interface gráfica conforme abaixo:



5. Crie uma aplicação para ler dois valores para as variáveis A e B, efetuar a troca dos valores de forma que a variável A passe a possuir o valor da variável B e que a variável B passe a possuir o valor da variável A. Apresentar os valores trocados.

Construa a interface gráfica conforme abaixo:



## Exercícios

6. Crie uma aplicação que implemente uma janela conforme a figura abaixo.

Ao ser pressionado o botão Calcular, deverá se multiplicar a quantidade de horas pelo salário hora e apresentar no campo Salário Bruto. Se a caixa Abono 5% estiver checada, haverá um acréscimo de 5% sobre o salário, se a caixa dependentes estiver checada, para cada dependente acrescente 100 ao salário.

O botão limpar limpa todos os campos e não deixa checado os CheckBox.

The image shows a Java Swing window titled "Salário". It contains the following elements:

- Qtde Horas:
- Salário Hora:
- Salário Bruto:
- ☐ Abono 5%
- ☐ Dependentes
- Calcular
- Limpar

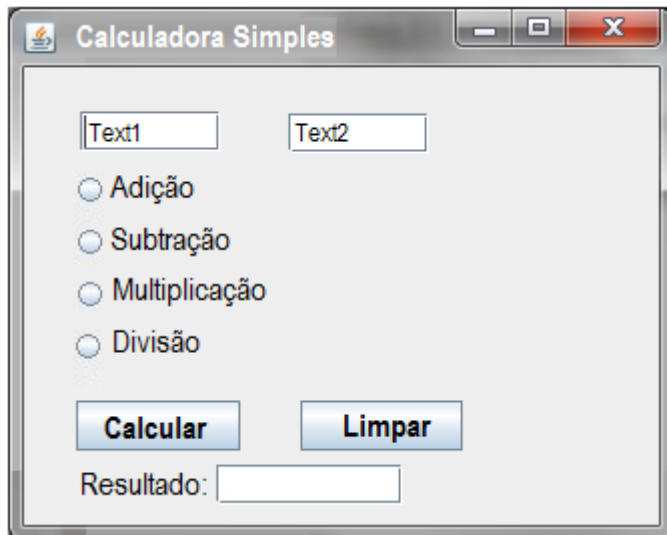
Para cada dependente, acrescente 100 ao salário

Somente leitura

Quantidade de dependentes

## Exercícios

7. Crie uma aplicação que implemente uma janela conforme figura abaixo. Ao pressionar o botão calcular, deverá ser informado no campo Resultado, de acordo com a operação selecionada, o cálculo realizado entre os dois números.



## Exercícios

8. Crie uma aplicação que implemente uma janela conforme figura abaixo. A pessoa deverá selecionar ou o número do candidato ou seu nome e, pressionando o botão confirmar, o voto irá para pessoa selecionada. Se pressionar o botão Branco, irá para Branco

Urna Eletrônica

Número do candidato: 1

Nome do candidato: João

Confirmar

Cancelar

Branco

Candidatos	Votos
João	0
Marcos	0
Pedro	0
Ana	0
Lúcia	0
Branco	0

De 1 a 5

Um atualiza o outro

Mesmos nomes

## Exercícios

9. Crie uma aplicação que implemente uma janela conforme figura abaixo:

JComboBox                      JList

**Verificação de Notas de Alunos**

Inserir Nome

Nome do Aluno:

Nota      Nota      Nota      Média

1      1      1      Text1  
2      2      2

Resultado

☐ Aprovado

☐ Reprovado

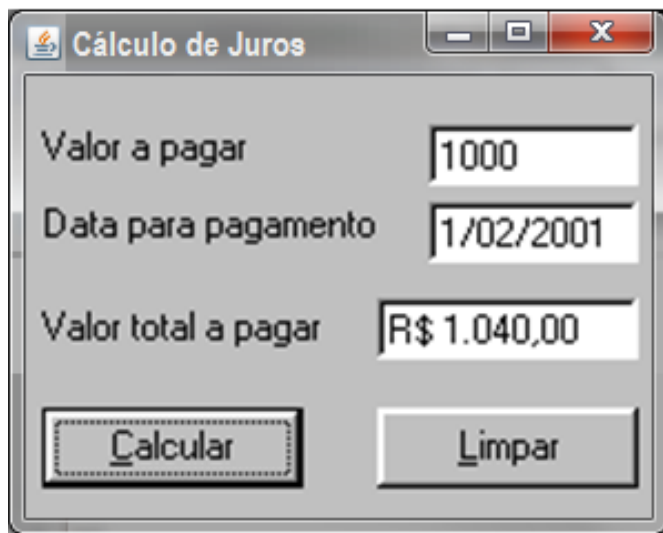
☐ Em Recuperação

Verificar      Limpar

Mensagem: Label6

## Exercícios

10. Crie uma aplicação que implemente uma janela onde deverá ser informado um valor qualquer a ser pago e a data para o pagamento do mesmo. Ao pressionar o botão Calcular, deverá aparecer o total a pagar de acordo com a quantidade de dias que ainda faltam para o pagamento, considerando um juros de 0,1% ao dia (Veja figura abaixo).



Cálculo de Juros

Valor a pagar 1000

Data para pagamento 1/02/2001

Valor total a pagar R\$ 1.040,00

Calcular Limpar

Juros de 0,1% ao dia em  
relação a data do sistema