



# TECNOLOGIA EM DESENVOLVIMENTO DE SOFTWARE MULTIPLATAFORMA

TÉCNICA DE  
PROGRAMAÇÃO I



Técnica de Programação I

Profº Luiz Cláudio

# Exercício 1

## CalculoCustoPiso



### CustoPiso

-comprimentoComodo: double  
-larguraComodo: double  
-precoporArea: double  
-custoTotalComodo : double

+inserirValores()  
+calcularprecoArea():void  
+mostrarValores(): void

### Principal

+main(args[]):String):void()

### Classe: Principal

#### Método main

- Instanciar um objeto do tipo CustoPiso chamado piso
- Apresentar um menu com as opções:
  - 1 – Inserir Valores do comodo
  - 2 – Calcular preço por Area
  - 3 – mostrar Valores
  - 0 – Sair

Obs.: Na opção Calcular preço Area Leia o valor do preço por área

### Classe: CustoPiso

#### Métodos

**inserirValores:** Lê (showInputDialog) e armazena (sets) valores para os atributos (comprimentoComodo e larguraComodo e precoporArea)

**calcularprecoArea:** calcula o custo total para assentar piso  
**setcustoTotalComodo(getComprimento() \* getLargura() \* getPrecoporArea())**

**mostrarValores:** Apresenta (showMessageDialog) os valores (gets) de todos os atributos.

# Exercício 2

## CalculoQuadrado



### Quadrado

-areaQuadrado: double  
-perimetroQuadrado : double  
+calcularArea(double ladoA): double  
+calcularPerimetro(double ladoA): double  
+mostrarValores(): void

### Principal

+main(args[]):void()

### Classe: Principal

#### Método main

- Instanciar um objeto do tipo Quadrado chamado quad
- Apresentar um menu com as opções:
  - 1 – Calcular Area Quadrado
  - 2 – Calcular Perimetro Quadrado
  - 3 – mostrar Valores
  - 0 – Sair

Obs.: Na opção dos Calculos. Leia o valor do lado A

### Classe: Quadrado

#### Métodos

**calcularArea:** recebe por parâmetro os valores do ladoA e calcula a Area **setAreaQuadrado ( Math.pow(ladoa,2) )**

**calcularPerimetro:** recebe por parâmetro os valores do ladoA calcula perimetro **setperimetroQuadrado(4 \* ladoa )**

**mostrarValores:** Apresenta (showMessageDialog) os valores (gets) de todos os atributos.

# Exercício 3

## CalculoEleitores



### Eleitores

-numeroTotalEleitores: int  
 -numeroVotosBrancos : int  
 -numeroVotosNulos: int  
 -numeroVotosValidos: int  
 -percBrancos: double  
 -percNulo: double  
 -percValidos: double

+inserirQtdVotos()  
 +calcularTotalEleitores():void  
 +calcularPercentualVotos():void

### Classe: Principal

#### Método main

- Instanciar um objeto do tipo Eleitores chamado eleit
- Apresentar um menu com as opções:
  - 1 – Inserir quantidade Votos
  - 2 – Calcular Total Eleitores
  - 3 – Calcular Percentual Votos
  - 3 – Mostrar Percentual Votos

### Principal

+main(args[]):String):void()

### Métodos

**InserirQtdVotos:** faça o input dos atributos números votos brancos, nulos e validos

**calcularTotalEleitores:** somar todos os votos

**calcularPercentualVotos:** Faça o percentual para cada voto  
 exemplo:

**setPercBrancos ((numeroVotosBrancos \* 100)/ total eleitores)**

Apresentar (showMessageDialog) o percentual de cada voto

# Exercício 4

## Calculo Consorcio



### Consorcio

```
-numeroTotalPrestacoes: int
-qtddPrestacoesPagas : int
-valorPrestacao: double
-saldodevedor: double
-totalPago: double

+inserirValoresConsorcio()
+calcularTotalpago(): double
+calcularSaldoDevedor():double
+mostrarValores(): void
```

### Classe: Principal

#### Método main

- Instanciar um objeto do tipo Consorcio chamado cons
- Apresentar um menu com as opções:
  - 1 – Inserir Valores Consorcio
  - 2 – Calcular Total Pago
  - 3 – Calcular Saldo Devedor
  - 4- Mostrar Valores
  - 0 – Sair

### Principal

```
+main(args[]):String):void()
```

### Métodos

**InserirValoresConsorcio:** faça o input dos atributos numeroTotalPrestacoes, qtddPrestacoesPagas, valorPrestacao

**calcularTotalPago:**  $\text{totalPago} = \text{valorPrestacao} * \text{qtddPrestacoesPagas}$

**calcularSaldoDevedor:**

$\text{saldodevedor} = (\text{numerototalprest} * \text{valorPrestacao}) - \text{totalpago}$

**mostrarValores:** Apresenta (showMessageDialog) os valores (gets) dos atributos totalpago e saldodevedor

# Exercício 5- ProjetoFuncionario



## Funcionario

-nomeFuncionario: string  
-telefoneFuncionario: string  
-enderecoFuncionario: string  
-salariohora: double  
-salariofinal: double

+cadastrarFuncionario():void  
+calcularSalarioFinal(Int hrTrab): double  
+listarFuncionario():void

## Principal

+main(args[]):String):void()

## Classe: Principal

### Método main

- Instanciar um objeto do tipo Funcionario chamado func
- Apresentar um menu com as opções:
  - 1 – Cadastrar Funcionario
  - 2 – Calcular Salário
  - 3 – Listar FUnctionario
  - 0 – Sair

Obs.: Na opção Calcular salário  
Leia o valor das horastrab e por fim  
mostre a Mensagem com valor do  
salario Final

## Classe: Funcionario

### Métodos

**cadastrarFuncionario:** Lê (showInputDialog) e armazena (sets) valores para os atributos (nomeFuncionario, telefone, endereço, salariohora). Exemplo:

**setNomeFuncionario(JOptionPane.showInputDialog("Digite o nome Funcionário: "));**

**calcularSalarioFinal:** recebe por parâmetro o valor da horas trabalhadas e multiplica pelo valor do salariohora

**setsalariofinal(horastrab\*salariohora)**

**listarFuncionario:** Apresenta (showMessageDialog) os valores (gets) de todos os atributos.