

 <div> <p>Instituto Federal do Triângulo Mineiro - Campus Patrocínio</p> <p>Análise e Desenvolvimento de Sistemas</p> </div>	
Disciplina: Programação Orientada a Objetos	Turma: 3º período
Professora: Ligia Pestili	
Entrega: 28/03/2025	Valor: 10 pontos

1) Implemente uma classe chamada **Pessoa** com os seguintes atributos:

- int **código**
- String **nome**
- String **dataCadastro**

Como subclasse da classe **Pessoa**, implemente duas classes:

- **Cliente**
 - Atributos: email, endereço e telefone (String - todos)
- **Usuario**
 - Atributos: login e senha (String - todos)

Implemente os métodos **getters e setters** de todos os atributos de todas as classes.

Faça uma classe **Principal** para testar o programa, com **dois objetos de cada classe. Teste todos os métodos.**

2) Crie uma classe (superclasse) chamada **ContaBancaria**. Adicione os seguintes atributos privados:

titular (String)
saldo (double)
rendaMensal(double)

Valide se a variável **titular** não está vazia. Se estiver vazia atribua o valor “**não identificado**”.
 Para validar se uma variável do tipo String contém informação diferente de espaço vazio, utilize o método **equals()** – pesquisem!

Valide se a **rendaMensal** é maior ou igual à dois salários mínimos, se **não** for maior atribua o valor de dois salários mínimos.

Crie o método **depositar**(double montante), adicione o valor da variável montante na variável **saldo**.

Crie o método **sacar**(double montante), verifique se o valor na variável **saldo** é maior que o valor passado por parâmetro, se for, subtraia o valor solicitado da variável **saldo** e envie uma mensagem “**saque efetuado com sucesso**”, senão envie a mensagem “**saldo insuficiente**”.

Crie o método **verificarSaldo()** que imprime para o usuário uma mensagem : “**O saldo na conta de <<titular>> é: R\$<<saldo>>**”. Substitua dos valores entre <<>> pelas variáveis correspondentes.
 Chame sempre esse método depois dos métodos **depositar()** e **sacar()**, seja na classe de teste ou dentro dos métodos depois das operações realizadas.

Crie uma classe (subclasse) chamada **ContaPouPanca**. Essa classe herdará da classe **ContaBancaria**. Adicione o atributo privado **taxaJuros** (double)

A variável **taxaJuros** só poderá aceitar valores entre **0.6 a 2.8**.

Crie os métodos **getter e setter** para o atributo.

Crie um método **aplicarJuros()**. Esse método deverá acrescentar a porcentagem da variável **taxaJuros** na variável **saldo**.

Crie uma classe (subclasse) chamada **ContaCorrente**. Essa classe herdará da classe **ContaBancaria**. Adicione o atributo privado **limiteEspecial** (double)

Valide a variável **limiteEspecial** conforme a variável **rendaMensal**. O limite especial na sua conta não pode ultrapassar **40% do seu salário**.

Implemente **Getters e Setters em todos os atributos de todas as classes**.

Crie uma classe **Principal** para testar o programa com 3 objetos de cada classe. Teste **todos** os métodos, inclusive enviando valores para que as validações sejam testadas corretamente.

3) Crie uma (superclasse) chamada **Funcionario**. Adicione os seguintes atributos privados:

nome(String)
salario(double)

Implemente um método chamado **calcularBonus**(String mesPagamento), nele todos os funcionarios receberão um bonus no mes de dezembro de 10% do salário. Além disso ele imprime o salário atualizado do funcionário: "**Funcionário <<nome>>. Salário para o mês <<mesPagamento>> : R\$<<valor calculado no método>>**"

Crie uma (subclasse) chamada **Gerente**. Essa classe herdará da classe **Funcionario**. Adicione o atributo privado **gratificacao** (double)

Crie uma (subclasse) chamada **Programador**. Essa classe herdará da classe **Funcionario**. Adicione o atributo privado **valorPontoFuncao** (double)

Crie um método chamado **calcularValorSoftware** (int linhasCodigo). Esse método faz um orçamento do software que o programador irá desenvolver. Para calcular **multiplique o valor de pontos por função pelas linhas de código**.

Implemente **Getters e Setters em todos os atributos de todas as classes**.

Crie uma classe **Principal** para testar o programa com 3 objetos de cada classe. **Teste conforme achar necessário para a verificação correta do sistema**.