

Nome do Projeto: Sistema Bancário em Java

Enunciado do Projeto:

Olá, você está prestes a embarcar em um projeto de desenvolvimento de um sistema bancário em Java. Este projeto tem como objetivo aplicar os conceitos de Herança, Polimorfismo, Downcasting, Upcasting, Métodos e Classes Abstratas, além de introduzir o uso de Enums para representar os tipos de conta disponíveis. Siga os requisitos abaixo para criar uma aplicação simples e robusta.

Classes Abstratas:

Crie uma classe abstrata chamada ContaBancaria que contenha os seguintes atributos:

int numeroConta

String titular

double saldo

Adicione um método abstrato double calcularSaldo() na classe ContaBancaria que deve ser implementado nas subclasses.

Classes Concretas:

Crie duas classes concretas que herdam de ContaBancaria: ContaCorrente e ContaPoupanca.

- **Métodos e Funcionalidades:**
 - Implemente métodos para depositar e sacar dinheiro em ambas as classes de conta. Certifique-se de validar se há saldo suficiente para o saque.
 - Crie um método para exibir as informações da conta, incluindo número da conta, titular e saldo.

Polimorfismo:

Crie uma lista de contas bancárias em sua aplicação e use polimorfismo para realizar operações em contas de diferentes tipos (ContaCorrente e ContaPoupanca) usando um único método.

Downcasting e Upcasting:

Realize downcasting e upcasting para acessar atributos e métodos específicos de cada tipo de conta quando necessário.

Enums para Tipo de Conta:

Introduza um Enum chamado TipoConta com os seguintes valores: CORRENTE e POUPANCA, que representarão os tipos de conta disponíveis.

Menu de Opções:

Crie um menu de opções para que o usuário possa escolher entre as seguintes ações:

- Criar nova conta (Corrente ou Poupança)
- Depositar em uma conta
- Sacar de uma conta
- Exibir informações da conta
- Sair do sistema

Regras Adicionais:

Implemente validações para garantir que o número da conta seja único para cada conta criada.

Dica: Para manter o código organizado, você pode criar uma classe separada chamada Banco para gerenciar todas as contas e operações.

Observações Finais:

Certifique-se de documentar seu código e implementar tratamento de exceções adequado para lidar com erros durante as operações. Este projeto abrange os conceitos de herança, polimorfismo, downcasting, upcasting, métodos, classes abstratas e o uso de Enums. Espero que você aproveite o desenvolvimento deste sistema bancário como uma forma prática de aplicar seus conhecimentos. Boa sorte!