

Fundamentos de Java

Exercícios Propostos

Classes Abstratas e Interfaces

www.softblue.com.br

# Exercício

Crie uma interface AreaCalculavel com um método calcularArea() e crie classes de figuras geométricas que implementam este método (como quadrado, circunferência e retângulo). Depois crie uma classe com um método main()para exercitar as chamadas aos métodos que calculam a área.

# Exercício

Crie uma classe ContaBancaria, que representa uma conta bancária genérica e não pode ser instanciada. Esta classe deve ter um atributo saldo (visível apenas para ela e para as suas subclasses) e os métodos depositar(double), sacar(double) e transferir(double, ContaBancaria). Estes métodos devem depositar um valor na conta, sacar um valor da conta e transferir um valor da conta de origem para uma conta de destino, respectivamente.

Além destes, ContaBancaria deve ter um método calcularSaldo(). Este método possui a regra do cálculo do saldo final (que pode ser diferente do saldo armazenado no atributo saldo) e deve ser obrigatoriamente implementado pelas subclasses de ContaBancaria, pois cada classe possui suas próprias regras de cálculo.

Crie duas subclasses de ContaBancaria: ContaCorrente e ContaInvestimento. Cada uma deverá implementar suas regras para calcular o saldo (método calcularSaldo()). No caso de ContaCorrente, o saldo final é o saldo atual subtraído de 10%, referente a impostos que devem ser pagos. Já para a ContaInvestimento, o saldo final é o saldo atual acrescido de 5%, referente aos rendimentos do dinheiro investido.

Crie uma aplicação que instancia uma conta corrente e uma conta investimento e executa as operações de depósito, saque, transferência e cálculo de saldo. Verifique se os resultados obtidos são consistentes com a proposta do modelo e com as regras de cálculo estabelecidas.

# Exercício

O Java possui uma interface chamada Cloneable, que pode ser implementada por classes que são capazes de gerar cópias de objetos. Esta interface não possui métodos, mas classes que a implementam devem sobrescrever o método clone(), definido na classe Object. Dentro deste método é implementada a lógica para criar um novo objeto com base no objeto original.

Com base nisto, crie uma classe Porta que suporta a criação de novos objetos (cópia). Ela deve ter os atributos altura (double), largura (double) e aberta (boolean). Também deve possuir os métodos abrir(), fechar() e os métodos *getters* correspondentes aos atributos.

Como uma porta pode criar outras cópias dela mesma, você deve sobrescrever o método clone(), o qual deve criar um novo objeto com os valores dos atributos copiados e retorná-lo.

**Dica**: O método clone()lança uma exceção (CloneNotSupportedException). Não é preciso se preocupar com ela neste momento, então declare o método main() da sua aplicação da seguinte forma:

public static void main(String[] args) throws CloneNotSupportedException {

//...

}