

## EXERCÍCIO – TRATAMENTO DE EXCEÇÕES

1. Analise a classe **ContaCorrente**, a seguir. Trabalhando no contexto de **programação defensiva** (*conjunto de técnicas de projeto e programação objetivando a estabilidade e a segurança de um software independentemente de seu imprevisível*):
  - a. Acrescente o tratamento de operações inválidas (saldo inicial, depósito ou saques nulos ou negativos e saque maior que o saldo) lançando exceções.
  - b. Crie exceções verificadas para tratar problemas com o saldo inicial e depósito ou retiradas menores ou iguais a zero.
  - c. Crie uma exceção não verificada para os problemas relativos a retiradas maiores que o saldo.

```
public class ContaCorrente {  
    private double saldo;  
    public ContaCorrente(double saldoInicial){  
        saldo = saldoInicial;  
    }  
    public void deposito(double valor){  
        saldo += valor;  
    }  
    public void saque(double valor){  
        saldo -= valor;  
    }  
    public double getSaldo(){  
        return saldo;  
    }  
}
```

2. Escreva um exemplo de uso que trate as exceções possíveis de serem lançadas pela classe **ContaCorrente** (exercício 1).
3. Acrescente na classe **ContaCorrente** dois contadores **static**. Um deve contar o número de instâncias de **ContaCorrente** criadas e o outro o número de tentativas de criação de instâncias de **ContaCorrente** (incluindo as que não foram bem-sucedidas). Sugestão: a contagem deve ser feita no método construtor (explore o uso de **finally**).