

## Lista de 10 Exercícios – Encapsulamento em Java

### 1. Criando Getters e Setters Simples

1. **Criar uma classe "Pessoa"** com os atributos `private nome` e `private idade`.
  - Criar métodos `getNome()` e `setNome(String nome)`.
  - Criar métodos `getIdade()` e `setIdade(int idade)`.
  - No main, criar um objeto, definir os valores e exibir com os getters.

### 2. Encapsulamento com Validação

2. **Modificar a classe "Pessoa"** para que `setIdade(int idade)` só aceite valores **maiores que zero**.
  - Se um valor inválido for passado, exibir uma mensagem de erro.

### 3. Criando uma Classe com Saldo Protegido

3. **Criar uma classe "ContaBancaria"** com os atributos `titular`, `saldo` e `numeroConta`, todos privados.
  - Criar um método `depositar(double valor)`, que só aceita valores positivos.
  - Criar um método `sacar(double valor)`, que só permite saques se houver saldo suficiente.

### 4. Protegendo o Salário de um Funcionário

4. **Criar uma classe "Funcionario"** com os atributos privados `nome`, `cargo` e `salario`.
  - Criar métodos `getSalario()` e `setSalario(double salario)`.
  - Impedir que `setSalario(double salario)` aceite valores negativos.

### 5. Validando Notas de um Aluno

5. **Criar uma classe "Aluno"** com os atributos `nome`, `nota1` e `nota2`, todos privados.
  - Criar métodos `getNota1()` e `setNota1(double nota)`.
  - Criar métodos `getNota2()` e `setNota2(double nota)`.
  - O método `setNotaX()` deve garantir que as notas sejam **de 0 a 10**.

### 6. Criando uma Classe "Produto" com Preço Protegido

6. **Criar uma classe "Produto"** com os atributos `nome` e `preco`, ambos privados.
  - Criar getters e setters, garantindo que `preco` **não pode ser negativo**.

### 7. Criando um Sistema de Biblioteca

7. **Criar uma classe "Livro"** com os atributos `titulo`, `autor` e `disponivel` (booleano, privado).
  - Criar um método `emprestar()`, que define `disponivel = false`.
  - Criar um método `devolver()`, que define `disponivel = true`.
  - Criar um método `getDisponivel()`, que informa se o livro está disponível ou não.

### 8. Controlando um Termostato

8. **Criar uma classe "Termostato"** com os atributos `temperaturaAtual` e `temperaturaDesejada`, ambos privados.
  - Criar getters e setters, garantindo que `temperaturaDesejada` **não pode ser menor que 15°C nem maior que 30°C**.

## 9. Criando um Controle de Acesso

9. Criar uma classe "**Usuario**" com os atributos nome e senha, ambos privados.
  - Criar um método setSenha(String senha), garantindo que a senha tenha **pelo menos 8 caracteres**.

## 10. Protegendo a Velocidade de um Carro

10. Criar uma classe "**Carro**" com os atributos privados marca, modelo e velocidadeAtual.
  - Criar um método acelerar(int incremento), que aumenta a velocidade, mas **não pode ultrapassar 180 km/h**.
  - Criar um método frear(int decremento), que reduz a velocidade, mas **não pode ser menor que 0 km/h**.