



# Exercício 1 - Java: Classes e Métodos Auxiliares

Disciplina: Programação Orientada a Objeto  
Professor: Dr. Camilo Barreto  
Curso: Grande Área de Computação  
UNIUBE

**Valor:** 2 pts

**Entrega:** 26/03/2025

## Organização das Classes em Pacotes:

Atenção, antes de iniciar a implementação das questões abaixo, utilize a seguinte organização de pacotes para a criação das classes.

- `com.uniube.pessoa`
  - **Classe:** Pessoa.java
- `com.uniube.veiculos`
  - **Classe:** Carro.java
- `com.uniube.financeiro`
  - **Classe:** ContaBancaria.java
- `com.uniube.vendas`
  - **Classe:** Produto.java
  - **Classe:** Pedido.java
- `com.uniube.main`
  - **Classe:** Main.java

## Questão 1: Classe `Carro`

Crie uma classe chamada `Carro` com os seguintes atributos:

- `String marca` (privado)
- `String modelo` (privado)
- `int ano` (privado)
- `int kilometragem` (privado)
- `double preco` (privado)

Adicione os seguintes métodos:

- `acelerar(int km)` : Aumenta a kilometragem do carro em uma certa quantidade de quilômetros.
- `frear()` : Imprime uma mensagem de que o carro está parando.
- `atualizarPreco(double novoPreco)` : Atualiza o preço do carro.
- Métodos `get` e `set` para todos os atributos.



No método principal `main`, crie um objeto da classe `Carro`, simule uma viagem, aumente a quilometragem e atualize o preço. Por fim, mostre os atributos atualizados no console usando `System.out.println`.

### Questão 2: Classe `Pessoa`

Implemente uma classe chamada `Pessoa` com os seguintes atributos:

- `String nome` (privado)
- `int idade` (privado)
- `String cpf` (privado)
- `double altura` (privado)
- `double peso` (privado)

Adicione os seguintes métodos:

- `calcularIMC()` : Calcula e retorna o Índice de Massa Corporal (IMC) usando a fórmula  $IMC = \frac{peso}{altura^2}$
- `fazerAniversario()` : Aumenta a idade em 1.
- Métodos `get` e `set` para todos os atributos.



No método `main`, crie uma instância de `Pessoa`, simule uma pessoa fazendo aniversário e calcule o IMC, mostrando o resultado no console.

### Questão 3: Classe `ContaBancaria`

Crie uma classe chamada `ContaBancaria` com os seguintes atributos:

- `String numeroConta` (privado)
- `double saldo` (privado)
- `String titular` (privado)
- `double limiteCredito` (privado)

Adicione os seguintes métodos:

- `depositar(double valor)` : Adiciona o valor ao saldo.
- `sacar(double valor)` : Subtrai o valor do saldo se houver saldo ou usa o limite de crédito.
- `verificarSaldoDisponivel()` : Retorna o saldo disponível, considerando o limite de crédito.
- `aumentarLimite(double valor)` : Aumenta o limite de crédito.



No método `main`, crie uma instância de `ContaBancaria`, faça um saque que usa o limite de crédito, e mostre o saldo disponível no console.

### Questão 4: Classe `Produto`

Implemente uma classe chamada `Produto` com os seguintes atributos:

- `String nome` (privado)
- `double preco` (privado)
- `int quantidade` (privado)
- `String categoria` (privado)
- `double desconto` (privado)

Adicione os seguintes métodos:

- `calcularValorTotal()` : Calcula o valor total do estoque (preço \* quantidade).
- `aplicarDesconto(double porcentagem)` : Aplica um desconto no preço do produto.
- `aumentarEstoque(int quantidade)` : Aumenta o estoque do produto.
- Métodos `get` e `set` para todos os atributos.



No método `main`, crie um produto, aplique um desconto e adicione mais itens ao estoque. Exiba o valor total atualizado no console.

## Questão 5: Classe `Pedido`

Crie uma classe chamada `Pedido` com os seguintes atributos:

- `int numeroPedido` (privado)
- `String cliente` (privado)
- `String data` (privado)
- `double valorTotal` (privado)
- `String status` (privado)

Adicione os seguintes métodos:

- `adicionarItem(double valor)` : Adiciona o valor de um item ao valor total do pedido.
- `removerItem(double valor)` : Remove o valor de um item do valor total.
- `finalizarPedido()` : Altera o status do pedido para "Finalizado" e exibe uma mensagem de conclusão.
- `cancelarPedido()` : Altera o status para "Cancelado" e imprime uma mensagem.
- Métodos `get` e `set` para todos os atributos.



No `main`, crie um pedido, adicione itens, remova itens, finalize o pedido e exiba o status e o valor total no console.

Bom exercício!