

# Instâncias, Construtores, Relações entre Classes e Sistemas

**2025-2026**

Ficha Prática – Aula 14

**TEC1 – UFCD 0789 – Fundamentos de  
linguagem Java**

Aula 14

# OBJETIVO

Os exercícios estão divididos em três blocos:

- 1. Exercícios obrigatórios (15)** – devem ser resolvidos em aula.
- 2. Exercícios opcionais (5)** – para treino adicional em aula ou em casa.
- 3. Exercícios de TPC (10)** – para consolidar os conteúdos fora do horário de aula.

## Exercícios Obrigatórios (15)

- 1. Explica com as tuas palavras o que é “sobrecarga de métodos”.**
- 2. Indica se as seguintes situações são sobrecarga ou não:**
  - a) métodos com o mesmo nome e diferentes parâmetros
  - b) métodos com o mesmo nome e mesmo número de parâmetros
  - c) métodos com nomes diferentes
  - d) métodos com o mesmo nome mas apenas retorno diferente
- 3. Escreve dois métodos sobrecarregados chamados mostrar():**  
Um sem parâmetros e outro que recebe um String detalhe.
- 4. Cria três versões do método somar:**  
somar(int, int)  
somar(int, int, int)  
somar(double, double)
- 5. Escreve em UML os três métodos do exercício anterior.**
- 6. Analisa o código e indica quantas versões do método existem e porquê:**  
void escrever(String t)  
void escrever(String t, int vezes)  
void escrever(int numero)
- 7. Cria duas versões do método calcularArea:**  
Com 1 parâmetro: raio (círculo)  
Com 2 parâmetros: largura e altura (retângulo)
- 8. Cria 3 construtores sobrecarregados da classe Filme:**  
vazio  
com título  
com título e duração
- 9. Escreve a UML da classe Filme com os 3 construtores.**
- 10. Analisa os construtores abaixo e explica se há sobrecarga válida:**  
Carro(String marca)  
Carro(String marca, String modelo)  
Carro(String marca) // duplicado
- 11. Cria um método descrever() sobrecarregado com:**  
Sem parâmetros  
Com parâmetro String  
Com parâmetro int
- 12. Desenha em UML a classe Calculadora com estas versões:**  
multiplicar(int,int)  
multiplicar(double,double)  
multiplicar(int,int,int)
- 13. Constrói um exemplo prático onde sobrecarga realmente é útil.**

**14. Cria a classe ContaBancaria com 2 construtores sobrecarregados:**

titular

titular + saldoInicial

**15. Escreve o diagrama UML dos dois construtores anteriores.**

## Exercícios Opcionais (5)

**1. Cria 4 versões do método imprimir:**

imprimir(String)

imprimir(int)

imprimir(String, int)

imprimir()

**2. Escreve o UML da classe “Relatorio” com os 4 métodos imprimir.****3. Cria 3 construtores para a classe “Jogo”:**

nome

nome + género

nome + género + ano

**4. Escreve a UML desses construtores.****5. Analisa o seguinte UML e transforma-o em Java:**

Pessoa

-----

+ Pessoa()

+ Pessoa(nome:String)

+ Pessoa(nome:String, idade:int)

## Trabalho para Casa (10)

**1. Cria a classe Animal com 3 métodos sobrecarregados “falar”.****2. Representa em UML as 3 versões do método falar().****3. Cria dois construtores na classe LivroDigital:**

nome

nome + tamanhoMB

**4. Escreve a UML de LivroDigital.****5. Cria métodos sobrecarregados enviarMensagem:**

enviarMensagem(String texto)

enviarMensagem(String texto, int prioridade)

**6. Escreve os métodos sobrecarregados de calcularPreco:**

calcularPreco(double base)

calcularPreco(double base, double desconto)

- 7. Desenha em UML a classe Nota com os dois métodos calcularPreco().**
- 8. Compara método normal vs método sobrecarregado (2 diferenças).**
- 9. Cria 3 versões de um método “registrar”:**
  - registrar()
  - registrar(String usuario)
  - registrar(String usuario, String senha)
- 10. Cria a UML para a classe “SistemaLogin” com essas 3 versões.**