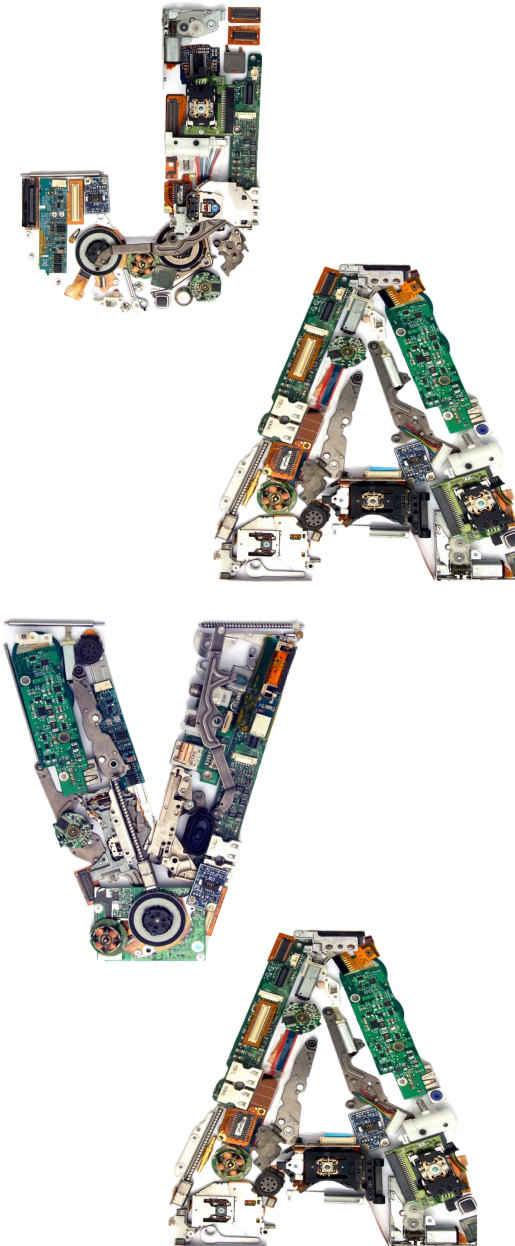


# Programação Orientada a Objetos com Java e WEB

Exercícios com implementação de classes, objetos e  
métodos



# Exercício 1: Sistema de controle pontuação em jogos

Você foi contratado para desenvolver um sistema simples para o jogo **TriScore**. O jogo permite que um jogador participe de **três rodadas**, e sua pontuação será registrada ao final de cada uma.

## Requisitos do Programa

1. O programa deve solicitar que o jogador informe **seu nome** antes de iniciar o jogo.
2. O jogador poderá jogar **três vezes**, e a pontuação de cada rodada deve ser armazenada.
3. Após as três rodadas, o programa deverá exibir:
  - a) O **nome do jogador**
  - b) A pontuação obtida em cada rodada
  - c) A **pontuação total** ao final das três rodadas

## Exercício 2: Sistema de controle de disciplinas e média

Um centro universitário contratou sua equipe de desenvolvimento para criar um sistema orientado a objetos para gerenciar os dados dos alunos da instituição.

Após o levantamento de requisitos realizado pela equipe de engenharia de software, constatou-se que, para cada aluno, devem ser registrados o número de matrícula, o nome e uma lista contendo os nomes das três disciplinas cursadas no semestre. Para cada disciplina, é obrigatório que o professor aplique duas provas, sendo necessário registrar as duas notas de cada disciplina.

O programa deverá exibir o número de matrícula e o nome do aluno. Para cada disciplina cursada, deve-se imprimir o nome da disciplina, a média das notas e o status do aluno (considerando aprovado se a média for maior ou igual a 6).

## Exercício 3: Representação de um ponto geométrico

Codifique uma classe chamada **Ponto** para representar um ponto geométrico no espaço. A classe deverá ter dois atributos para representar as coordenadas  $x$  e  $y$  do ponto no espaço.

Codifique as seguintes funcionalidades:

1. Método para calcular e retornar a distância entre dois pontos.
2. Método para calcular e retornar a distância de um ponto até a origem.
3. Método que receba como parâmetro dois pontos e retorne o ponto que está mais próximo da origem.
4. Método para retornar os dados do objeto no formato:  $(x, y)$ .

Codifique a classe principal para executar as seguintes descritas acima.

## Exercício 4: Sistema de gestão de funcionários

Você foi designado(a) para desenvolver um programa em Java para gerenciar informações sobre os funcionários de uma empresa. Você deverá implementar as classes e os métodos apresentados a seguir:

### ❑ Classe Funcionario

- A classe **Funcionario** deve conter atributos como **nome**, **cargo** e **salario**.

### ❑ Método de Aumento de Salário (aumentar salário de todos os funcionários)

- Crie um método chamado **aumentarSalario** que recebe um valor percentual de aumento como parâmetro e aplica esse aumento ao salário do funcionário.

### ❑ Método de Aumento de Salário (aumentar salário de acordo com um cargo)

- Crie um método chamado **aumentarSalario** que recebe como parâmetro um cargo e o percentual de aumento. Caso o cargo do funcionário seja igual ao cargo recebido como parâmetro, aplique o percentual de aumento no salário.

## Exercício 4: Sistema de gestão de funcionários

### ❑ Método de Promoção

- Implemente um método chamado **promover** que receba como parâmetro o novo cargo e faça a atualização do cargo do funcionário para o novo cargo.

Implemente o programa principal que simule a interação com o sistema. O programa deve instanciar objetos, simular a atualização do salário e a promoção do funcionário.