Exercícios 02 – POO Básico

# Exercício 1: Aluno

Escreva uma classe chamada Aluno cujos objetos representam alunos matriculados em uma disciplina. Cada objeto dessa classe deve guardar os seguintes dados do aluno: matricula, nome, um array com duas notasDeProva e uma notaDeTrabalho. Implemente os seguintes métodos para esta classe:

media: Calcula a média final do aluno (cada prova tem peso 2.5 e o trabalho tem peso 2).  
final: Calcula quanto o aluno precisa para a prova final (retorna zero se ele não for para a final).

Para testar, instancie objetos do tipo Aluno e utilize-os para testar a classe.

# Exercício 2: Ponto

Crie uma classe chamada Ponto. A classe deverá possuir três construtores:

Um construtor sem parâmetros, que inicializa o ponto nas coordenadas (0,0).  
Um construtor que recebe dois parâmetros de coordenadas X e Y.  
Um construtor que inicializa o ponto através das coordenadas de outro ponto passado como parâmetro.

Adicione um método distancia que calcula a distância euclidiana entre o ponto atual e um outro ponto passado como parâmetro.

Para testar, instancie objetos do tipo Ponto e utilize-os para testar a classe.

# Exercício 3: Pessoa

Implemente uma classe chamada Pessoa que possua os seguintes atributos: nome, idade, peso e altura. Inicialize esses dados através do construtor da classe.

Crie um método IMC que calcule o Índice de Massa Corporal usando a fórmula IMC = peso/(altura^2). Implemente também um método categoriaIMC que retorne a categoria do IMC da pessoa de acordo com a tabela fornecida.

Para testar, instancie objetos do tipo Pessoa e utilize-os para testar a classe.

# Exercício 4: Veículo

Crie uma classe chamada Veiculo. A classe deve ter os atributos: marca, modelo, ano e quilometragem.

Implemente os métodos:

andar(distancia: number): Aumenta a quilometragem do veículo pela distância passada como parâmetro.  
info(): Retorna uma string com as informações do veículo.

Para testar, instancie objetos do tipo Veiculo e utilize-os para testar a classe.

# Exercício 5: Pet

Desenvolva uma classe chamada Pet. A classe deve ter os atributos: nome, tipo e idade.

Implemente os métodos:

aniversario(): Aumenta a idade do pet em 1 ano.  
info(): Retorna uma string contendo todas as informações do pet.

Para testar, instancie objetos do tipo Pet e utilize-os para testar a classe.

# Exercício 6: Tarefa

Elabore uma classe "Tarefa" para gerenciar tarefas. Cada tarefa deve ter uma descrição e um nível de prioridade. Inclua um método para marcar a tarefa como concluída. Desenvolva também uma classe "GerenciadorDeTarefas" que deve ser capaz de adicionar tarefas, listar todas as tarefas pendentes e listar tarefas por nível de prioridade.

Realize testes criando tarefas e utilizando as funcionalidades do gerenciador.

# Exercício 8: Livro

Crie uma classe "Livro" com atributos para título, autor e ISBN, além de um estado que indica se está disponível para empréstimo. Implemente métodos para emprestar e devolver o livro. Desenvolva uma classe "Biblioteca" que gerencia uma coleção de livros, permitindo adicionar livros, emprestar e consultar livros por ISBN.

Faça testes para adicionar livros à biblioteca, realizar empréstimos e devoluções.

# Exercício 9: Evento

Implemente uma classe "Evento" que representa um evento, contendo nome, descrição, data e uma lista de participantes. A classe "Pessoa" deve ser usada para representar os participantes. A classe "Evento" deve permitir adicionar e remover participantes, buscar participantes por e-mail e listar todos os participantes.

Teste a classe criando um evento e manipulando a lista de participantes.