

Lista de Exercícios XI

Esta lista de exercício deve:

- Ser realizada em equipes de até 06 alunos.
- Ser entregue no **prazo** proposto.
- Ter os algoritmos pedidos escritos em **linguagem Java ou C#**.
- Ter todos os algoritmos **devidamente indentados**.

Exercícios:

O objetivo deste exercício é aplicar os conceitos de **Abstração (Classes e Métodos Abstratos)** e **Polimorfismo** para modelar diferentes tipos de veículos em um sistema de gerenciamento.

Tarefas

Você deve criar um sistema para modelar veículos utilizando classes abstratas.

1. Criar a Classe Abstrata Veiculo

Crie uma classe abstrata chamada Veiculo que deve conter:

- **Atributos:**

- String marca
- String modelo
- int ano

- **Construtor:** Um construtor que inicialize os três atributos.

- **Método Concreto:** Um método chamado `exibirDetalhes()` que imprime a marca, o modelo e o ano do veículo.

- **Método Abstrato:** Um método chamado `acelerar()` que não deve ter corpo e **obriga** as subclasses a fornecerem sua própria implementação de como o

Centro Universitário UNA

Programação de Soluções Computacionais

Professor Daniel Henrique Matos de Paiva

2. Criar as Subclasses Concretas

Crie as seguintes classes concretas que herdam de Veiculo e implementam o método `acelerar()` de forma específica:

Subclasse	Implementação do <code>acelerar()</code>
Carro	Imprime: <i>"[Modelo] [Marca] está acelerando de 0 a 100 km/h em X segundos."</i>
Moto	Imprime: <i>"[Modelo] [Marca] está acelerando e empinando a roda dianteira!"</i>
Caminhao	Imprime: <i>"[Modelo] [Marca] está engrenando a próxima marcha e acelerando lentamente."</i>

3. Criar a Classe de Teste (SistemaVeiculos)

Crie uma classe `SistemaVeiculos` com o método `main()` para testar seu código. •

Instancie um objeto de cada subclasse: um `Carro`, uma `Moto` e um `Caminhao`.

- Utilize o polimorfismo, declarando as referências como tipo `Veiculo` (ex: `Veiculo carro = new Carro(...)`).
- Para cada objeto, chame os dois métodos: `exibirDetalhes()` e a

Centro Universitário UNA

Programação de Soluções Computacionais

Professor Daniel Henrique Matos de Paiva

Resultado Esperado (Exemplo)

```
una-teach-psc-202502 - SistemaVeiculos.java

1 public class SistemaVeiculos {
2     public static void main(String[] args) {
3
4         // 1. Instanciando e usando a referência do tipo Veiculo (Polimorfismo)
5         Veiculo carro = new Carro("Toyota", "Corolla", 2023);
6         Veiculo moto = new Moto("Honda", "CBR 600", 2024);
7         Veiculo caminhao = new Caminhao("Volvo", "FH 540", 2020);
8
9         // 2. Testando o Carro
10        System.out.println("--- Teste Carro ---");
11        carro.exibirDetalhes(); // Método concreto herdado
12        carro.acelerar(); // Método abstrato implementado (Carro)
13        System.out.println("-----");
14
15        // 3. Testando a Moto
16        System.out.println("--- Teste Moto ---");
17        moto.exibirDetalhes(); // Método concreto herdado
18        moto.acelerar(); // Método abstrato implementado (Moto)
19        System.out.println("-----");
20
21        // 4. Testando o Caminhão
22        System.out.println("--- Teste Caminhão ---");
23        caminhao.exibirDetalhes(); // Método concreto herdado
24        caminhao.acelerar(); // Método abstrato implementado (Caminhao)
25        System.out.println("-----");
26    }
27 }
```

Suba todos os códigos .java para o repositório: **psc-lista-11**

Envie o link do seu repositório para a plataforma solicitada em aula.