

Parte 1: Exercício de Diagrama de Classes

Você foi contratado para projetar a estrutura de classes de um sistema simplificado de gerenciamento de uma **locadora de veículos**. O sistema precisa registrar os veículos disponíveis, os clientes e os aluguéis.

Sua tarefa é desenhar um **Diagrama de Classes UML** para este sistema, incluindo todas as classes, seus atributos, métodos e os relacionamentos entre elas.

Requisitos do Sistema

1. Entidades Principais:

- Veiculo: Classe genérica para qualquer veículo na frota.
- Carro: Um tipo específico de veículo.
- Moto: Outro tipo específico de veículo.
- Cliente: Representa a pessoa que aluga os veículos.
- Aluguel: Representa o registro de uma locação.

2. Atributos e Métodos por Classe:

➤ Veiculo

Atributos:

- marca: String
- modelo: String
- ano: int
- placa: String (Identificador único)
- disponivel: boolean

Métodos:

- + verificarDisponibilidade(): boolean
 - + exibirDetalhes(): String
-

➤ **Carro (É um Veículo)**

Atributos:

- numeroPortas: int
- tipoCombustivel: String

Métodos:

- + ligarArCondicionado(): void
-

➤ **Moto (É um Veículo)**

Atributos:

- cilindradas: int

Métodos:

- + buzinar(): void
-

➤ **Cliente**

Atributos:

- nome: String
- cpf: String (Identificador único)
- telefone: String
- historicoAlugueis: List<Aluguel>

Métodos:

- + realizarAluguel(veiculo: Veiculo): Aluguel
 - + devolverVeiculo(aluguel: Aluguel): void
-



➤ Aluguel

Atributos:

- id: int (Identificador único)
- dataInicio: Date
- dataFim: Date
- valorTotal: double
- cliente: Cliente
- veiculoAlugado: Veiculo

Métodos:

- + calcularValor(dias: int): double
- + encerrarAluguel(): void

3. Relacionamentos:

- Carro **é um tipo de** Veiculo.
- Moto **é um tipo de** Veiculo.
- Um Cliente pode ter **zero ou muitos** Alugueis.
- Um Aluguel **pertence a um** Cliente.
- Um Aluguel **refere-se a um** Veiculo.
- Um Veiculo pode ter **zero ou muitos** Alugueis registrados.