

---

## Lista de Exercícios 07

### Sistema de Clínica Veterinária 🐕 🐈

Você já possui as classes **Tutor, Animal, Veterinário, Atendimento, Procedimento, AtendimentoEmergencia e AtendimentoRetorno e Pessoa.**

Agora, incremente o projeto considerando: **Herança, Classes Abstratas, Polimorfismo e Interfaces**

#### 1. Nova Classe Abstrata: Atendimento

Transforme a classe Atendimento em abstrata, representando um atendimento genérico.

Atributos:

id (int)

animal (Animal)

veterinario (Veterinario)

data (LocalDateTime)

descricao (String)

Métodos:

Método abstrato `calcularValor()`, que deverá ser implementado nas subclasses.

Método `exibirResumo()` que exibe as informações do atendimento e o resultado de `calcularValor()`.

#### 2. Subclasses de Atendimento

As classes `AtendimentoEmergencia` e `AtendimentoRetorno` devem herdar de `Atendimento` e implementar o método `calcularValor()` de forma distinta:

`AtendimentoEmergencia`: valor base + taxa de emergência.

`AtendimentoRetorno`: valor reduzido (por exemplo, 50% do valor base).

#### 3. Interface: Pagavel

Crie uma interface `Pagavel` com os seguintes métodos:

`double getValorTotal();`

`String getDescricaoPagamento();`

As classes AtendimentoEmergencia e AtendimentoRetorno devem implementar essa interface, retornando o valor total e uma descrição textual do pagamento.

#### 4. Interface: Registravel

Crie uma interface Registravel com o método:

```
void registrarNoSistema();
```

Faça com que Tutor, Veterinario e Atendimento (ou suas subclasses) implementem essa interface, exibindo mensagens diferentes:

Tutor: "Tutor registrado no sistema."

Veterinario: "Veterinário registrado com CRMV válido."

Atendimento: "Atendimento nº [id] registrado com sucesso."

#### 5. Polimorfismo

Demonstre o uso de polimorfismo criando uma lista de objetos do tipo Pessoa e uma lista de objetos do tipo Atendimento na classe main.

Percorra cada lista exibindo os resultados de getResumo() e exibirResumo(), respectivamente.

Observe como o comportamento muda conforme o tipo real do objeto.

#### 6. Testes no Main

No método main:

Crie 5 Tutores e 3 Veterinarios.

Crie pelo menos dois animais vinculados a cada Tutor.

Crie um AtendimentoEmergencia e um AtendimentoRetorno para pelo menos 4 dos animais.

Chame os métodos getResumo(), registrarNoSistema() e getDescricaoPagamento() de cada um.

Adicione todos os objetos em listas e percorra-as exibindo os resultados para demonstrar o polimorfismo.