

# Reescrita de métodos

Um conceito importante associado à herança é a **reescrita de métodos** (ou **sobrescrita de métodos**), que permite que uma classe filha redefina um método que foi herdado de sua classe pai.

## O que é Reescrita de Métodos?

Ocorre quando uma classe filha fornece sua própria implementação para um método que já existe na classe pai. Isso é útil quando a classe filha precisa de um comportamento diferente, mesmo que o método compartilhe o mesmo nome e parâmetros que o método original na classe pai.

## Exemplo básico:

python

```
class Animal:
    def fazer_som(self):
        return "Algum som genérico"

class Cachorro(Animal):
    def fazer_som(self):
        return "Latido"

class Gato(Animal):
    def fazer_som(self):
        return "Miado"
```

# Reescrita com Super()

Quando reescrevemos um método, podemos querer manter parte da funcionalidade da classe pai e adicionar ou modificar algumas partes. Para isso, podemos utilizar a função `super()`.

python

```
class Animal:
    def fazer_som(self):
        return "Algum som genérico"

class Cachorro(Animal):
    def fazer_som(self):
        som_base = super().fazer_som() # Chama o método da classe pai
        return som_base + " e Latido"
```

# Quando Utilizar a Reescrita de Métodos?

- A reescrita de métodos é útil em várias situações, como:
- **Polimorfismo:** Quando diferentes classes devem responder ao mesmo método de maneiras distintas.
- **Especialização:** Quando uma classe filha precisa refinar ou expandir o comportamento de um método herdado.
- Exemplo: Imagine um sistema de pagamento em que diferentes tipos de métodos de pagamento (cartão, boleto, etc.) precisam processar transações de maneira diferente.

python

```
class Pagamento:
    def processar(self):
        raise NotImplementedError("Este método deve ser sobrescrito")

class CartaoCredito(Pagamento):
    def processar(self):
        return "Processando pagamento via cartão de crédito"

class Boleto(Pagamento):
    def processar(self):
        return "Gerando boleto para pagamento"
```

# Cuidados ao Reescrever Métodos

- O método reescrito na classe filha deve ter o mesmo nome e parâmetros que o método da classe pai.
- **Chamar `super()` quando necessário:** Se precisar aproveitar funcionalidades da classe pai, sempre utilize o `super()` para garantir que o comportamento herdado seja mantido.

# Exercício Prático

6. Implemente um sistema que simula o comportamento de diferentes tipos de veículos. Crie uma classe `Veiculo` com um método `mover()`. Em seguida, crie subclasses como `Carro`, `Bicicleta` e `Barco`, cada uma com sua própria versão do método `mover()`.