

Interface x herança

DigitalHouse>



**Certified Tech
Developer**

The Ultimate Degree

Índice

1. Interfaces
2. Interfaces X Herança

1 | Interfaces



Vamos dar uma olhada mais de perto nas **interfaces** e, em seguida, discutir sua relação com a **herança**.



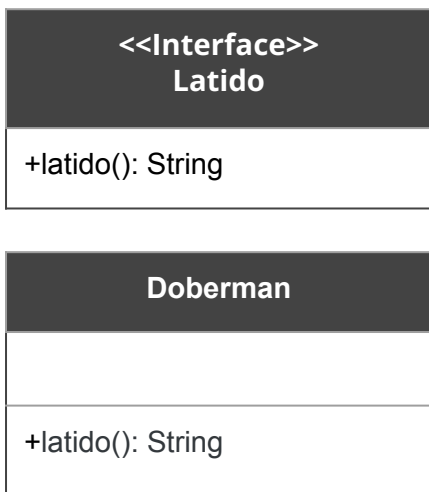
Interfaces

A interface é um recurso muito utilizado em Java, bem como na maioria das linguagens orientadas a objeto, para **“obrigar”** um determinado grupo de classes a ter métodos ou propriedades em comum em um determinado contexto, contudo os métodos podem ser **implementados** em cada classe de uma maneira diferente.



Interfaces

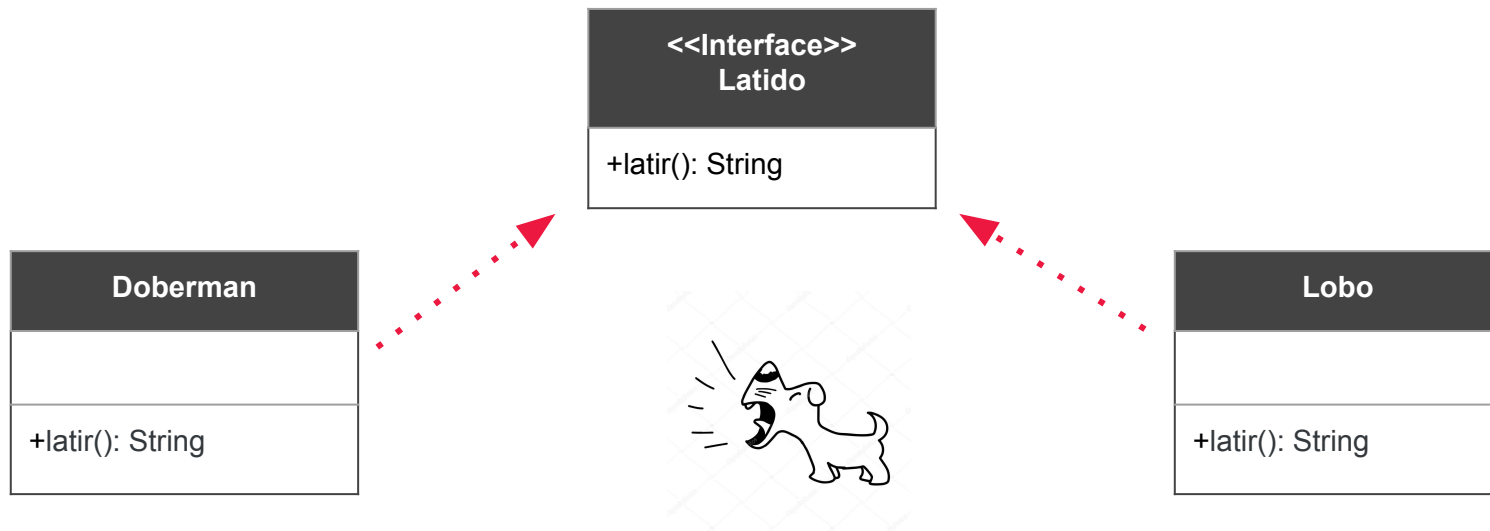
Uma **interface** estabelece um **contrato**. Cada classe que implementa uma interface é obrigada a implementar todos os métodos dessa interface.

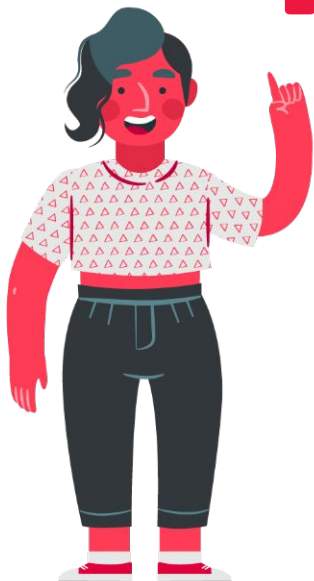


Doberman deve implementar o método latir() para implementar a interface Latido.

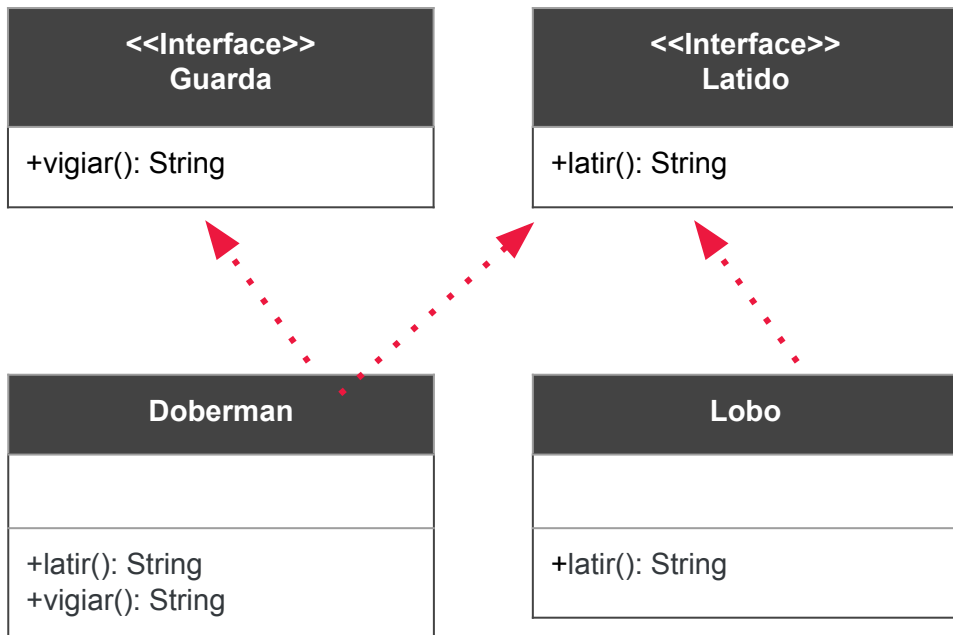
Interfaces

Por exemplo, todas as classes que implementam a interface **Latido** devem implementar o **método latir**.





Uma classe só pode herdar de uma única classe, mas pode implementar várias interfaces.

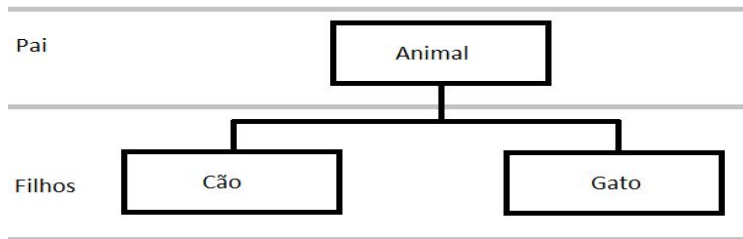


Como podemos ver, o Doberman implementa 2 interfaces, pois além de latir() também terá a responsabilidade de vigiar por ser um Cão Guarda.

2 | Interfaces e Herança

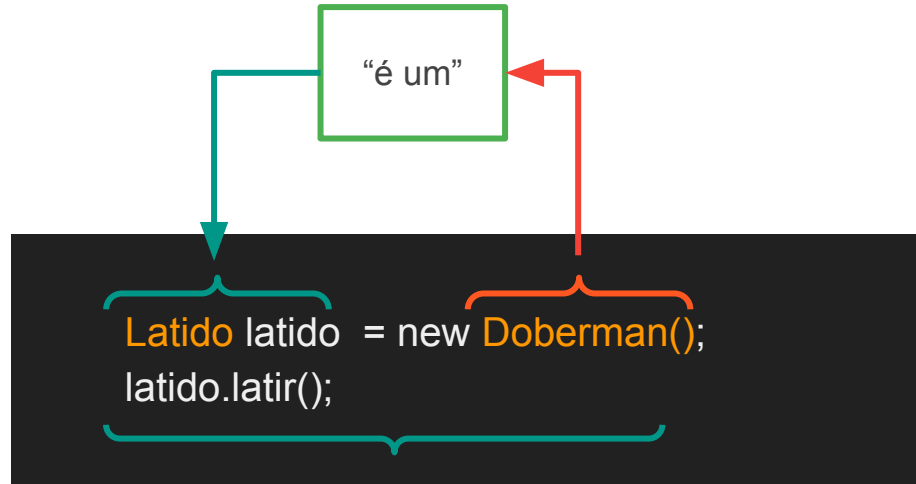
Interfaces e Herança

Quando herdamos de uma classe, adicionamos **atributos** e comportamentos da **classe pai**, e quando implementamos uma **interface**, apenas forçamos a classe que a implementa a **substituir**, ou seja, **implementar**, seus métodos.



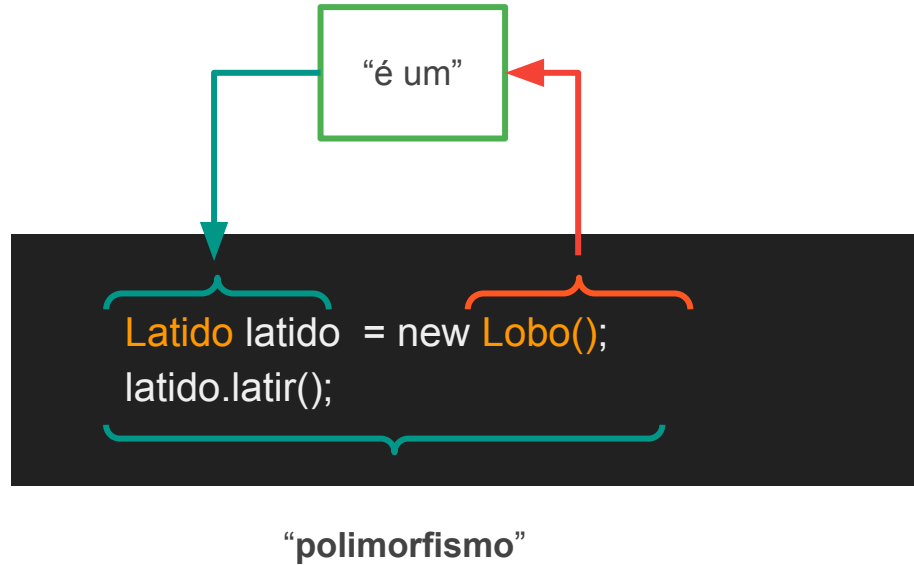
Exemplo de herança

Exemplo 1



"polimorfismo"

Exemplo 2



DigitalHouse>
Coding School