



Sobrecarga e sobreescrita de métodos



**Certified
Developer**
The Ultimate Tech Degree

DigitalHouse >
Coding School



Temas

1

**Sobrecarga de
métodos**

2

**Sobreescrita de
métodos**



1 | Sobrecarga de métodos



A sobrecarga do método está relacionada **à assinatura do método.**





Sobrecarga de métodos

É possível no paradigma orientado a objetos ter na mesma classe **dois ou mais métodos** com o **mesmo nome** e cujo comportamento é diferente. Isso é viável porque, no momento de invocar esse método, você pode saber qual invocar, desde que sua assinatura seja diferente.

O método **latir()** está sobrecarregado, ambos métodos têm assinaturas **diferentes**.

Cachorro

- nome: String
- idade: int

+ brincar()
+ latir()
+ latir(intensidade: int)



O nome e a quantidade, o tipo e a ordem dos parâmetros fazem parte da assinatura de um método e devem ser diferentes para poder sobrecarregá-lo.



O valor retornado por um método e os modificadores de visibilidade não fazem parte da assinatura.





Sobrecarga de métodos

Como o que um método retorna não faz parte da assinatura, os métodos sobrecarregados podem retornar coisas diferentes ou iguais.

Vemos que um dos métodos **latir()** devolve uma String.

Cachorro

- nome: String
- idade: int

+ brincar()
+ latir(): String
+ latir(intensidade: int)

2

Sobreescrita de métodos



Sobreescrita de métodos



Quando lemos a palavra **sobrescrever**, a primeira ideia nos vem é de **reescrever** algo existente.

Esta primeira ideia é muito semelhante ao que podemos encontrar na programação orientada a objetos e que será fundamental para que determinados cenários sejam resolvidos.



Para **sobrescrever** métodos, precisamos de um **relacionamento de herança**, uma vez que o que vamos sobrescrever é um método da superclasse para que ele se comporte de maneira diferente na subclasse.



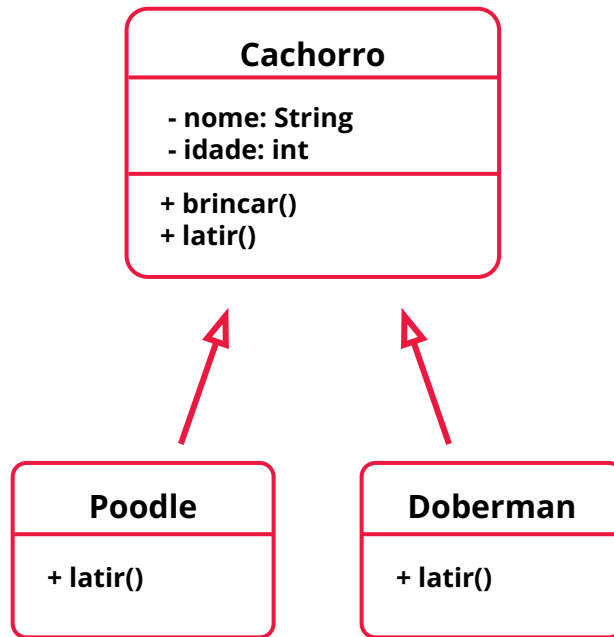
Ao contrário da sobrecarga, em que os métodos devem ter assinaturas diferentes, neste caso, os métodos devem ter a mesma assinatura.





Sobreescrita de métodos

Ao escrever o método `latir()` nas subclasses, estamos dizendo que esse método é sobreescrito, portanto, se comporta de maneira diferente.



DigitalHouse>
Coding School