

Métodos Construtores



Regiane Moreno





CONCEITO


Conceito

O objetivo de um método construtor é inicializar objetos novos quando são criados e também executar outros procedimentos iniciais de configuração.

O OPERADOR NEW

O operador `new` cria uma instância de um determinado objeto:

```
Produto prod1 = new Produto();
```



Foi criada uma referência para o objeto `Produto`, que se chama `prod1`. Por meio desta referência, podemos executar os métodos da classe `Produto` e modificar os atributos dela. Vamos ver a classe `Produto`...

```
public class Produto {  
    int codigo;  
    String descricao;  
    double preco;  
  
    ....  
}
```



O MÉTODO CONSTRUTOR DEFAULT

O método construtor default

Aqui, é feita a chamada do método construtor padrão:



```
Produto prod1 = new Produto();
```

Este método não recebe nenhum parâmetro e inicializa todos os atributos da classe com os valores padrão:

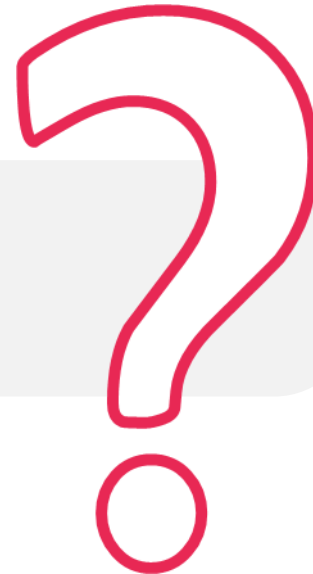
- numéricos inicializados com zero;
- boolean inicializados com `false`; e
- `char/String` inicializados com `null`.



CONSTRUTOR PERSONALIZADO

Construtor personalizado

E se eu quiser instanciar um objeto com valores específicos?

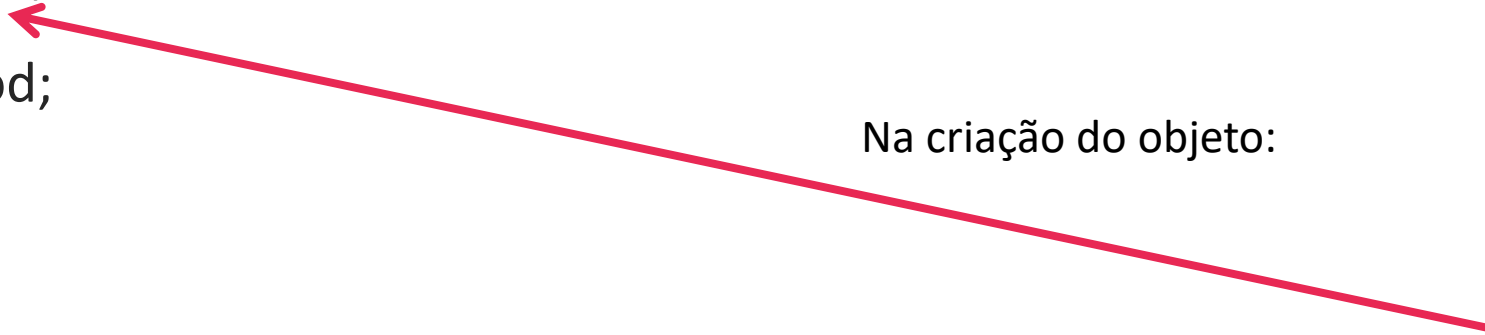


Construtor personalizado

```
public class Produto {  
    int codigo;  
    String descricao;  
    double preco;  
    Produto (int cod){  
        codigo = cod;  
    }  
}
```

Na criação do objeto:

Produto prod1 = new Produto(12345);



Um objeto pode ter mais de um construtor

```
public class Produto {  
    int codigo;  
    String descricao;  
    double preco;  
    Produto (int cod){  
        codigo = cod;  
    }  
    Produto (int cod, String des){  
        codigo = cod;  
        descricao = des;  
    }  
}
```

Quando um objeto tem mais de um método construtor, dizemos que ele está sobrecarregado. Um método sobrecarregado deve ter uma assinatura diferente para cada construtor.

A CLÁUSULA *THIS*

A cláusula this

A variável `this` é uma referência para o próprio objeto.


Pode ser usada para resolver ambiguidade de nome entre um atributo e um parâmetro ou uma variável local de algum método:

```
public class Produto {  
    int codigo;  
    String descricao;  
    double preco;  
    Produto (int codigo){  
        this.codigo = codigo;  
    }  
    Produto (int codigo, String descricao){  
        this.codigo = codigo;  
        this.descricao = descricao;  
    }  
}
```



TESTE DO PROJETO NA IDE

Teste do projeto na IDE



**Vamos testar o
projeto Produto
no IDE!**

