



UNIVERSIDADE  
FEDERAL DO CEARÁ

TÓPICO  
**03**



# PROGRAMAÇÃO ORIENTADA A OBJETOS

Prof. Marcos Vinicius de Andrade Lima

E-mail: [marcos.vinicius@ufc.br](mailto:marcos.vinicius@ufc.br)



## Olá!

### Sou Marcos Vinicius

No tópico passado nós demos os primeiros passos de OO com Java...

Neste tópico começaremos a **definir** as características dos nossos objetos!

“

**Poucos aceitam o fardo da própria vitória. A maioria desiste dos sonhos quando eles se tornam possíveis** (Paulo Coelho)



**Definindo Características dos Objetos**

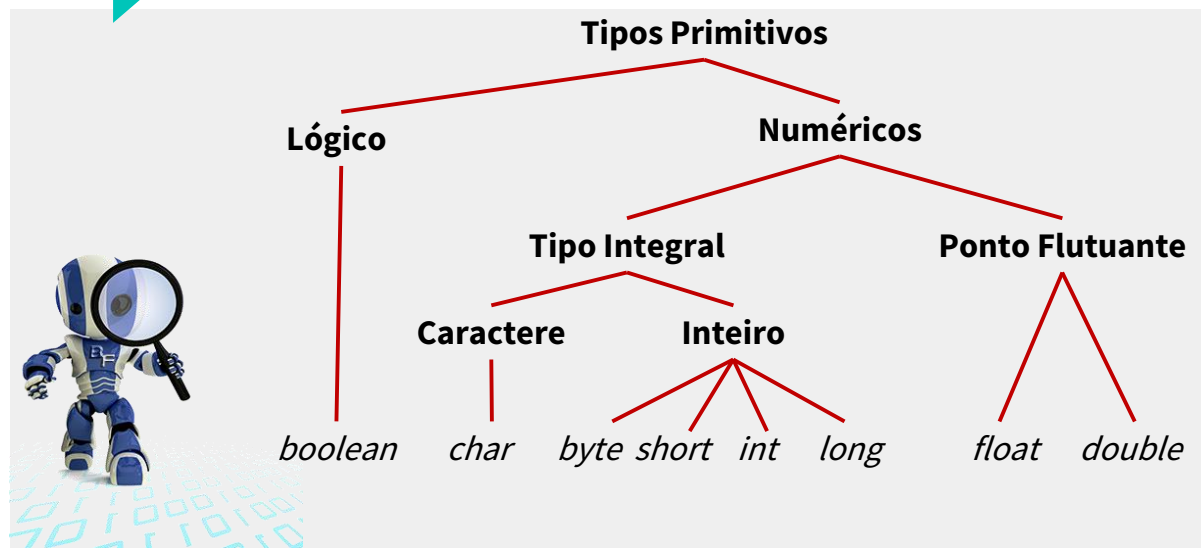
## Os TIPOS PRIMITIVOS

- Java possui 8 (oito) tipos primitivos:
- **Tipos inteiros:**
  - byte
  - short
  - int
  - long
- **Tipos ponto flutuante (decimais)**
  - float
  - double
- **Tipo lógico**
  - boolean
- **Tipo caractere**
  - char

Prof. Marcos Vinicius – UFC/Russas - POO

5/43

## Os TIPOS PRIMITIVOS



Prof. Marcos Vinicius – UFC/Russas - POO

6/43

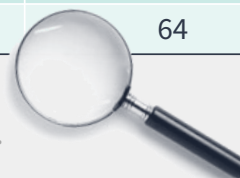
## Os TIPOS PRIMITIVOS

- Cada tipo primitivo tem um tamanho definido, como pode ser visto na tabela abaixo:

Tipo	Tamanho (bits)	Tipo	Tamanho (bits)
boolean	1	char	16
byte	8	short	16
int*	32	long	64
float	32	double**	64

\* padrão para constantes inteiras;

\*\* padrão para constantes de ponto flutuante.



Prof. Marcos Vinicius - UFC/Russas - POO

7 / 43



# CUIDADO!!!

Em Java, **booleanos** só podem receber valores *true* ou *false*.

## DECLARAÇÃO DE VARIÁVEIS/ATRIBUTOS

- Java é uma linguagem de programação **fortemente tipada**;

- Para declarar variáveis e/ou atributos:

**<tipo> <variável>;**

- Exemplos:**

**char genero;**

**float volume, total, salario;**

Isso é bom?



Prof. Marcos Vinicius - UFC/Russas - POO

9/43

## INICIALIZANDO VARIÁVEIS/ATRIBUTOS

- Para inicializar variáveis e/ou atributos:

**<variável> = <expressão/valor>;**

- Exemplos:**

**genero = 'm' ;**

**volume = 112.1;**

**total = 9.94;**

Prof. Marcos Vinicius - UFC/Russas - POO

10/43

## DECLARANDO E INICIALIZANDO VARIÁVEIS/ATRIBUTOS

- Para declarar e inicializar variáveis e/ou atributos:  
`<tipo> <variável> = <expressão/valor>;`
- Exemplos:  
`int x = 1, y = 100;`  
`long big = 12341234L;`

Prof. Marcos Vinicius - UFC/Russas - POO

11/43

## SE LIGA!

- É importante saber que tipos primitivos em Java **alocam** espaço na memória quando são declarados.



Prof. Marcos Vinicius - UFC/Russas - POO

12/43

## VAMOS PENSAR UM POUCO...

```
...  
int a = 2147483647;  
System.out.println(a); // 2147483647  
a = a + 1;  
System.out.println(a); // -2147483648  
...
```

Na maioria das linguagens os tipos possuem um tamanho finito, então um simples  $a+b$  pode ocasionar uma **overflow**!



Prof. Marcos Vinícius - UFC/Russas - POO

13/43

## O QUE É, O QUE É ...

- Que **possui características próprias e comportamento associado**, comportando-se como um ser vivo?



Prof. Marcos Vinícius - UFC/Russas - POO

14/43

## OBJETO

- Um **objeto** é qualquer coisa, real ou abstrata, na qual nós **armazenamos dados e as operações que manipulam os dados** (Martin).
- Um objeto ou uma **instância** é uma abstração de alguma coisa no domínio do problema, refletindo as capacidades do sistema de manter informações sobre ele, interagir com ele, ou ambos, um **encapsulamento de valores de atributos e seus serviços exclusivos** (Yourdon).

Prof. Marcos Vinicius – UFC/Russas - POO

15/43

## OBJETO

- Um objeto é uma entidade **independente**, composta por:
  - **Estado Interno:** uma memória interna em que valores podem ser armazenados e modificados ao longo da vida do objeto (conjunto de atributos ou variáveis de instância).
  - **Comportamento:** conjunto de ações predefinidas (denominadas métodos) através das quais o objeto responderá a demanda de processamento por parte de outros objetos.

Prof. Marcos Vinicius – UFC/Russas - POO

16/43



## QUEM VEIO PRIMEIRO?



**Classe**



**Objeto**

Prof. Marcos Vinicius - UFC/Russas - POO

17/43

## DEFININDO CLASSES EM JAVA

- Suponha que precisamos trabalhar com um tipo de dados diferente dos tipos de dados já existentes em Java, **Robot**, por exemplo.
- Como Java não tem esse nosso tipo, precisamos defini-lo.
- Mas antes de trabalharmos com variáveis do tipo **Robot**, precisamos definir esse novo tipo de dados, conhecido como **tipo agregado de dados**.
- O tipo agregado de dados é um tipo de dados definido pelo programador.



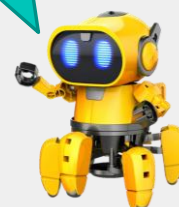
Prof. Marcos Vinicius - UFC/Russas - POO

18/43

## CLASSE ROBOT

```
public class Robot
{
}
```

Tá de  
brincadeira  
comigo?



Prof. Marcos Vinicius - UFC/Russas - POO

19/43

## CRIANDO UM OBJETO DO TIPO ROBOT...

```
public class MatrixWorld
{
    public static void main(String[] args)
    {
        Robot r2d2;           // declarando variável
        r2d2 = new Robot();    // criando um objeto
    }
}
```



Prof. Marcos Vinicius - UFC/Russas - POO

20/43

O operador **new** é quem cria um objeto.



## OLHA SÓ QUE LEGAL!

- No exemplo anterior a variável `r2d2` quando é declarada **NÃO** aloca espaço na memória.
- Variáveis declaradas do tipo agregado **NÃO** contêm o dado e sim uma **referência** (ponteiro) para o dado.
- Após declarada a variável, a mesma **NÃO** referencia nenhum lugar. Nesse caso, diz-se que ela tem referência nula.
- Para representar o nulo, usamos a palavra reservada `null`.

## SE LIGA NOS PARANAUÊ...

- Depois que o **new** é utilizado, a memória está alocada e diz-se que um objeto foi criado.

**Lembre-se:** criar um objeto → instanciar um objeto.

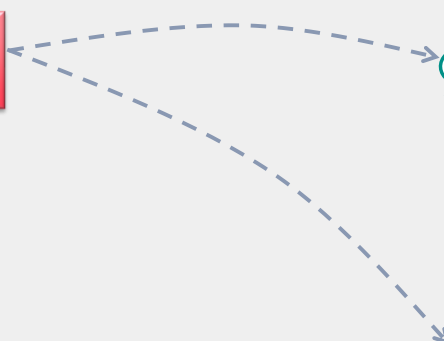
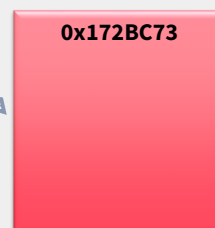
- A variável **r2d2** é chamada variável de referência, pois referencia um objeto.
- É fundamental ter uma distinção clara entre variável e objeto! Deste simples entendimento dependem diversos conceitos e aplicações em POO.

## DECLARAÇÃO Vs. CRIAÇÃO

**Robot r2d2;**



**r2d2 = new Robot();**



## QUAL A UTILIDADE DA CLASSE **Robot**?

Objetos da classe **Robot** não vão conseguir realizar **nenhuma ação** útil e não possuem **nenhuma característica** definida!



Prof. Marcos Vinicius - UFC/Russas - POO



25/43

## QUAL A UTILIDADE DA CLASSE **Robot**?

- Na classe **Robot** só foi definido o corpo exterior!
- Nenhuma **propriedade** ou **ação** foi especificada ainda.



Prof. Marcos Vinicius - UFC/Russas - POO

26/43

## ATRIBUTOS

- No **dicionário** Aurélio **atributo** pode ser resumido como: a quantidade própria de um elemento, **caráter essencial de sua substância**.
- Em **programação orientada a objetos** **atributo** é visto como o conjunto de **características específicas de um objeto**.

Prof. Marcos Vinicius – UFC/Russas - POO

27 / 43

## DEFININDO ATRIBUTOS

- Vamos definir alguns atributos para objetos do tipo **Robot**!
- Cada objeto do tipo **Robot** deve ter:
  - ✓ o número total de sensores;
  - ✓ o tipo de voz ('M' ou 'F');
  - ✓ se está no modo carregando;
  - ✓ uma escala de 0 a 10 para indicar o nível da bateria.
- **Cada atributo deve ser de que tipo?**
- **Quais nomes você daria para cada um dos atributos?**



Prof. Marcos Vinicius – UFC/Russas - POO

28 / 43

## DEFININDO ATRIBUTOS

```
public class Robot
{
    byte totalSensores;
    char voz;
    boolean isCarregando;
    byte nivelCarga;
}
```



Prof. Marcos Vinicius - UFC/Russas - POO

29 / 43

## ACESSANDO PROPRIEDADES DOS OBJETOS

- As propriedades de um objeto são chamadas de variáveis de instância, pois cada instância tem seus próprios valores para suas variáveis de instância.
- **Como acessar as variáveis de instância do objeto?**  
**Resposta:** por meio do operador “.” (ponto), como pode ser visto no trecho de código a seguir:



Prof. Marcos Vinicius - UFC/Russas - POO

30 / 43

## ACESSANDO PROPRIEDADES DOS OBJETOS

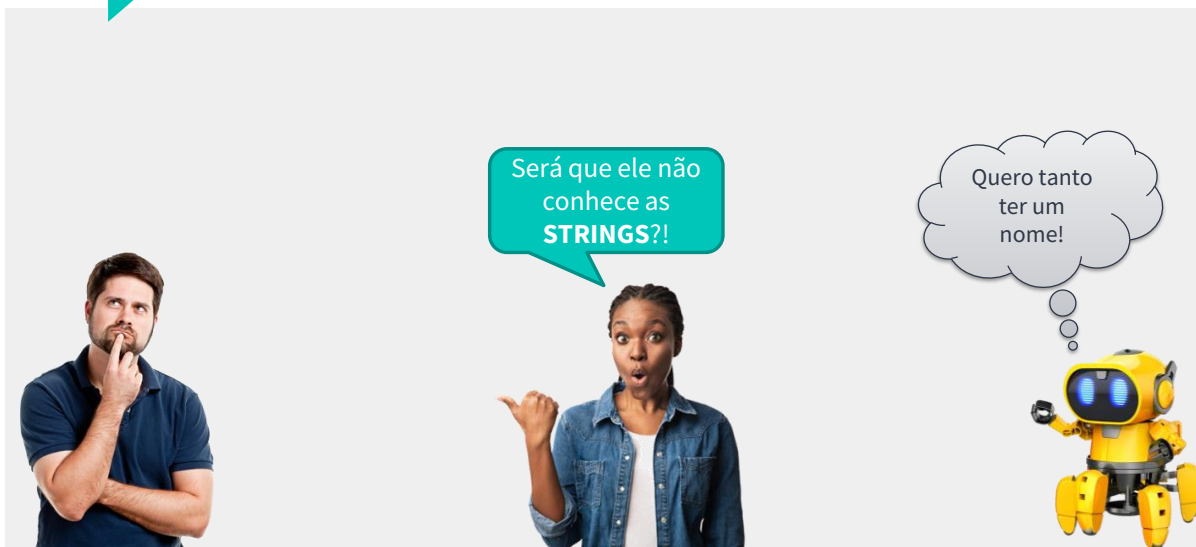
```
public class Robopolis{  
    public static void main(String[] args){  
        Robot r2d2 = new Robot();  
        r2d2.totalSensores = 2;  
        r2d2.nivelCarga = 10;  
        System.out.println(r2d2.totalSensores);  
        System.out.println(r2d2.nivelCarga);  
        System.out.println(r2d2.isCarregando);  
    }  
}
```



Prof. Marcos Vinicius - UFC/Russas - POO

31/43

## E SE EU QUISER ADICIONAR UM NOME EM ROBOT?



Prof. Marcos Vinicius - UFC/Russas - POO

32/43



## CADEIA DE CARACTERES: STRING

- Em Java, as cadeias de caracteres (*strings*) **NÃO** são tipos primitivos!!!
- **Exemplos:** “Eng. de Software”, “Universidade”, “POO-Java”, “”
- **Declaração:** String nome;  
String cor = “azul”;
- **Concatenação:** “Hello” + “World” → “HelloWorld”
- **Conversão implícita:** “Hello” + 2016 → “Hello2016”

Prof. Marcos Vinicius – UFC/Russas - POO

33/43

## CLASSE ROBOT ATUALIZADA

```
public class Robot
{
    String nome;
    byte totalSensores;
    char voz;
    boolean isCarregando;
    byte nivelCarga;
}
```



Prof. Marcos Vinicius – UFC/Russas - POO

34/43

## ACESSANDO PROPRIEDADES DOS OBJETOS

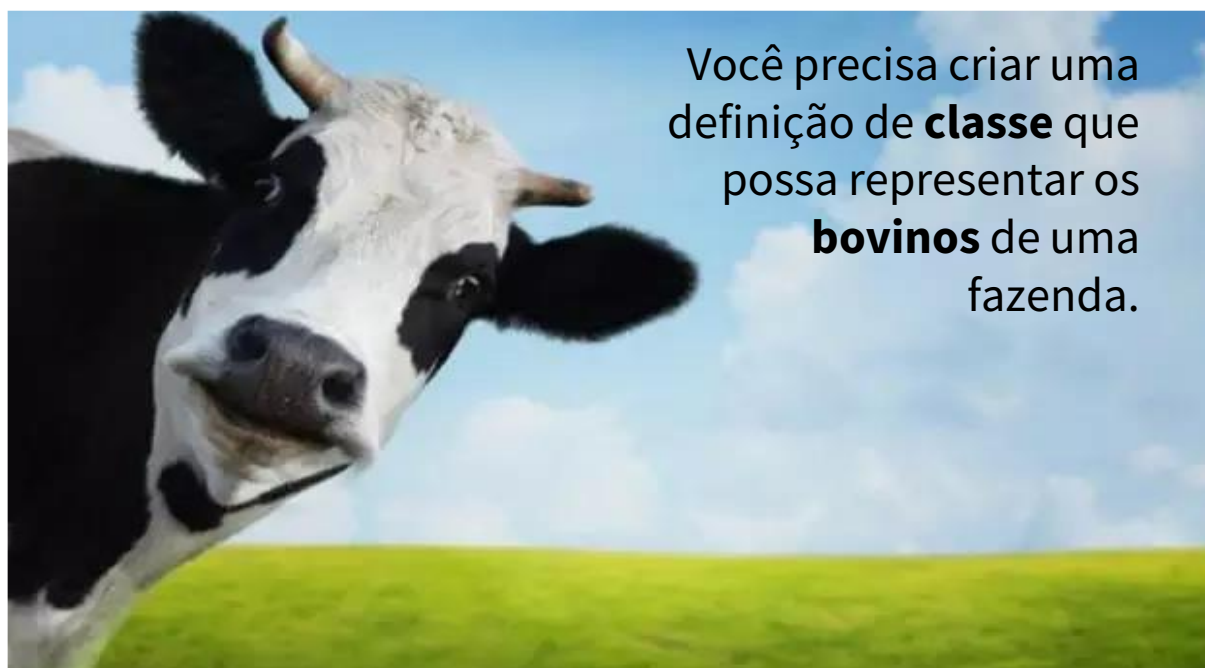
```
public class Robopolis{  
    public static void main(String[] args){  
        Robot r2d2 = new Robot();  
        r2d2.totalSensores = 2;  
        r2d2.nivelCarga = 10;  
        r2d2.nome = "Rex-B";  
        System.out.println(r2d2.totalSensores);  
        System.out.println(r2d2.nivelCarga);  
        System.out.println(r2d2.isCarregando);  
    }  
}
```



Prof. Marcos Vinicius - UFC/Russas - POO

35/43





Você precisa criar uma definição de **classe** que possa representar os **bovinos** de uma fazenda.

## VAMOS PENSAR UM POUCO...

**Eureka!** É preciso usar de **abstração** para definir quais atributos devem ser considerados...



## VAMOS PENSAR UM POUCO...

O que um bovino tem?



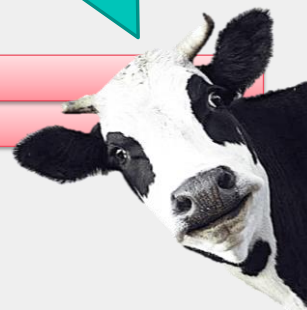
Prof. Marcos Vinicius - UFC/Russas - PÓO

39 / 43

## ATRIBUTOS DE OBJETOS BOVINOS

- Código → Código de identificação do bovino
- Nome → Nome do bovino
- Peso → Peso do bovino em quilogramas
- Raça → Raça do bovino
- Data de Nascimento → Data de nascimento do boi
- Gênero → Gênero do bovino
- Produção de Leite → Quant. de leite produzida/dia
- Signo → Signo do bovino

Sou libriano  
com ascendente  
em Sagitário!



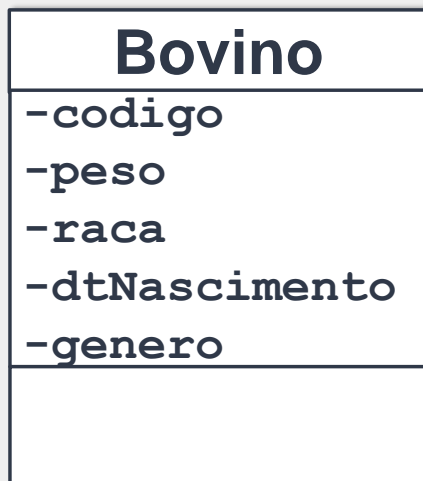
Prof. Marcos Vinicius - UFC/Russas - PÓO

40 / 43

## CLASSE BOVINO EM UML



Prof. Marcos Vinicius - UFC/Russas - POO



41/43

# Codifique **todas** as classes necessárias!

- Classe **principal** (classe executável, onde tudo começa)
- Classe **Bovino**
- Classe **Data**



# Obrigado!

## Mais alguma dúvida?

Acesse o **AME** para mais informações e treinamento do **NERDS!**

<http://ame2.russas.ufc.br>

