



**UNIFACS**  
LAUREATE INTERNATIONAL UNIVERSITIES

# Programação Orientada a Objetos

Aulas nº 02: POO, Classes e Objetos.

Prof. Luis Gustavo Araujo  
2018

# Objetivo

Conhecer o conceito de POO, Classes e Objetos e suas aplicabilidades. Implementar Classes e instanciar Objetos, visando entender, na prática, as diferenças entre os dois conceitos.

Classes

Objetos

POO

# Conceitos

Orientado?

Paradigma?

Objetos?

Classes?



# Programação Orientada a Objetos (POO)

“

*you pass to work in a way more close to human reality. For each important need we will have objects that interact with each other and that are composed of state (attributes) and behavior (methods).*

**Rodrigo  
Turini**

Desbravando Java e Orientação a Objetos

Desbravando Java e  
Orientação a Objetos  
Um guia para o iniciante da linguagem



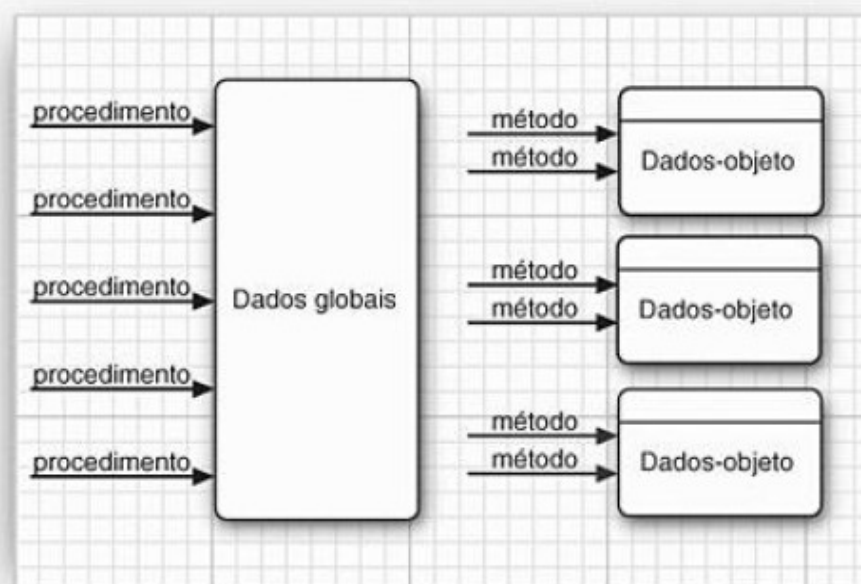
Casa do  
Código

RODRIGO TURINI

# Por que usar POO?



Essa última estrutura é muito mais fácil de entender. Também é muito mais fácil localizar *bugs*. Suponha que os dados de um determinado objeto estejam em um estado incorreto. É muito mais fácil procurar o culpado entre os 20 métodos que tinham acesso a esse item de dados do que entre 2.000 procedimentos.”  
(HOSTMANN, CORNELL, 2010)



**Figura 4-1** Programação procedural versus programação OO (orientada a objeto)

# Por que usar POO?

1

Permite manutenção mais fácil.

2

Reutilização de código.

3

Maior segurança.

4

Proximidade com o mundo real.

# Objetos

“

*Aristóteles foi provavelmente o primeiro a iniciar um estudo cuidadoso do conceito de tipo; Ele falou "a classe de peixes e a classe de pássaros. [...] todos os objetos, enquanto únicos, são também parte de uma classe de objetos que tem características e comportamentos em comum.*

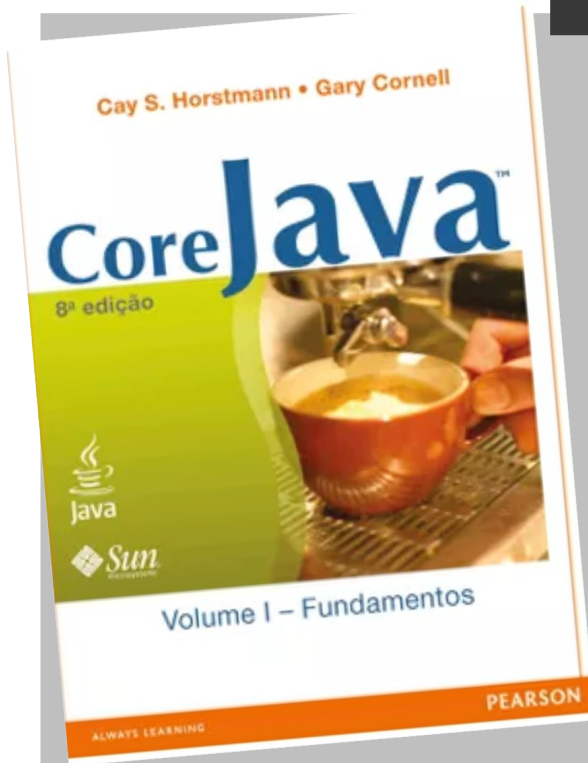
**Bruce  
Eckel**

Pensando em Java



# Classes

“



*Uma classe é um modelo ou esquema a partir do qual os objetos serão criados. Pense na classe como uma faca de cortar bolo. Os objetos são os próprios bolos. Ao construir um objeto a partir de uma classe, diz-se que você criou uma instância da classe.*

## Hortmann & Cornell

Core Java



# Classes

“



*Objetos que são idênticos exceto pelo seu estado durante uma execução de programa são agrupados juntos em “classes de objetos”.*

**Bruce  
Eckel**

Pensando em Java

# Classes

Possuem atributos que representam o seu **estado** e métodos que representam seu **comportamento**.



Orelha

Focinho

Olhos

Pelo

...

Ouvir

Cheirar

Latir

Correr

...

# Classes e Objetos na Prática



# Classes

Implementando em Java:



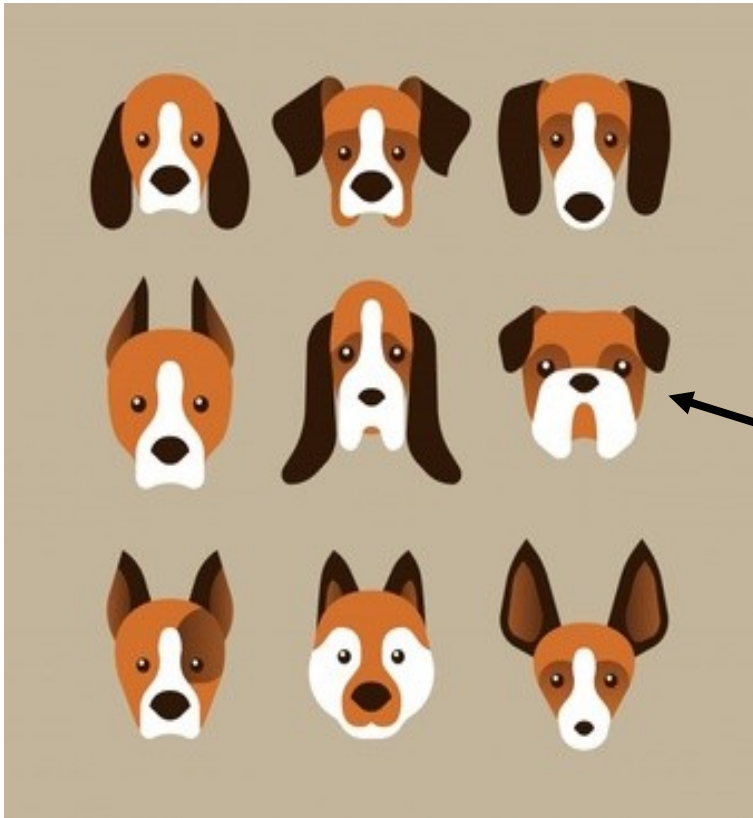
```
public class Cachorro {
```

```
    String pelo;  
    String olhos;  
    String focinho;  
    double altura;  
    String orelha;
```

```
}
```

# Objeto

Instanciando objetos em Java:

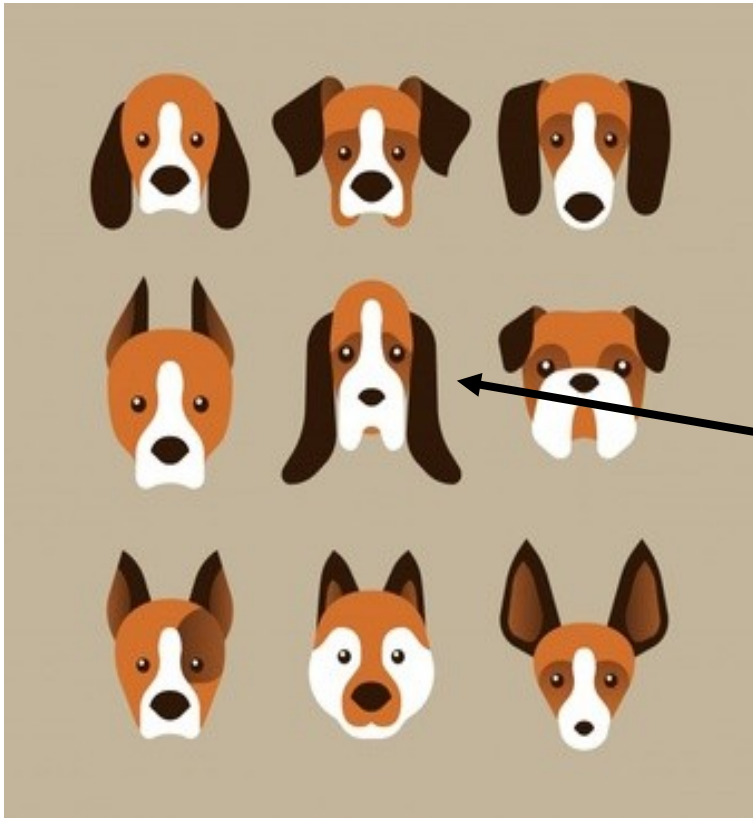


```
Cachorro dog = new Cachorro()
```

```
dog.pelo = "marrom";  
dog.olhos = "pequeno";  
dog.focinho = "caido";  
dog.altura = 40.1;  
dog.orelha = "curta";
```

# Objeto

Instanciando objetos em Java:



```
Cachorro dog = new Cachorro()
```

```
dog.pelo = "marrom";  
dog.olhos = "tristes";  
dog.focinho = "caído";  
dog.altura = 100;  
dog.orelha = "longa";
```

# Resumo

Criando uma classe

```
public class Cachorro {
```

```
    String pelo;
```

```
    String olhos;
```

```
    String fucinho;
```

```
    double altura;
```

```
    String orelha;
```

```
}
```

Tipos dos atributos

Atributos

Tipo da Classe

```
Cachorro dog = new Cachorro()
```

Construtor da Classe

Definindo estado do objeto

```
dog.pelo = "marrom";
```

```
dog.olhos = "tristes";
```

```
dog.focinho = "caido";
```

```
dog.altura = 100;
```

```
dog.orelha = "longa";
```

Referência do objeto

Instanciando objeto

# Atividade



**No site da disciplina:**

[http://luisaraujo.github.io/aulas/unifacs/estrutura\\_de\\_dados/2018/home/](http://luisaraujo.github.io/aulas/unifacs/estrutura_de_dados/2018/home/)



**Google Short URL (direto)**

**<https://goo.gl/jGsYDK>**



# Atividade

```
public class Ave {
```

```
    String bico;  
    String pena;  
    String pernas;  
    String porte;
```

```
}
```

```
Ave ave1 = new Ave()
```

```
ave1.bico = "pequeno";  
ave1.pena = "avermelhada";  
ave1.pernas = "curta";  
ave1.porte = "pequeno";
```

```
Ave ave2 = new Ave()
```

```
ave2.bico = "grande";  
ave2.pena = "azulada";  
ave2.pernas = "longa";  
ave2.porte = "médio";
```

```
Ave ave3 = new Ave()
```

```
ave3.bico = "grande";  
ave3.pena = "amarelada";  
ave3.pernas = "curta";  
ave3.porte = "médio";
```

# Referências Técnicas

ECKEL, B. **Pensando em Java** (tradução de Thinking in Java. 3. ed. Prentice-Hall, Dezembro 2002. ) - Online.

HORSTMANN, S.; CORNELL, G.; **Core JAVA**. 1. vol. São Paulo: Pearson Education, 2010.

TURNI, R. **Desbravando Java e Orientação a Objetos**: um guia para o iniciante da linguagem. São Paulo: Casa do Código, 2016.



**UNIFACS**  
LAUREATE INTERNATIONAL UNIVERSITIES

# Programação Orientada a Objetos

Aulas nº 01: POO, Classes e Objetos.

Prof. Luis Gustavo Araujo  
2018