

Linguagens de Programação II

Aula 03 – Orientação à Objetos com Java

Prof. Marc Antonio Vieira de Queiroz

Centro Universitário Filadélfia
Ciência da Computação/Sistemas de Informação

Orientação à Objetos com Java

Como os objetos se comportam

Conteúdo

- Objetos
 - Estados
 - Comportamentos
- Exemplos
- Exercícios

Objetos

- Sabemos:
 - Estado → (afeta) comportamento
 - Comportamento → (afeta) estado
- objetos:
 - Estados (variáveis de instância)
 - Comportamento (métodos)

Exemplo 01



Dog A

Name = "Fido"

Size = 37 kg



Dog B

Name = "Killer"

Size = 5 kg

EmitirSom() —————> Qual Latido mais alto?

“ **métodos** usam os valores das **variáveis de instância** ”

Comportamento de um objeto

- Cada objeto de um mesmo tipo pode ter um método com comportamento diferente?

Comportamento de um objeto

- Cada objeto de um mesmo tipo pode ter um método com comportamento diferente?
 - Mais ou menos
 - Toda instância → mesmos métodos
 - Métodos podem se comportar diferente com base no valor dos atributos

Exemplo 02



conhece

Song

title
artist

faz

setTitle()
setArtist()
play()

estado

comportamento

Politik
Coldplay

Start me Up
Stones

My Way
Sinatra



Song t2
.play() ?



Song t3
.play() ?

Exemplo 03



Dog
size
name
bark

```
void bark(){
    if(size > 60){
        System.out.println("Woof! Woof!");
    }
    else if(size > 14){
        System.out.println("Ruff! Ruff! ");
    }
    else{
        System.out.println("Yip! Yip!");
    }
}
```

bark() → usa **size** para decidir o som

Exercício 01

- Fazer TestDrive para testar o código do slide anterior (método bark) de uma classe Dog;

Enviar Valores para um Método

- d.bark(3) ? ...

Método



parâmetros

Chamador



argumentos

Exemplo 04



Dog
size
name
bark

Parâmetro
(variável local)

```
void bark(int numOfBarks){  
    while(numOfBarks > 0){  
        System.out.println("ruff");  
        numOfBarks--;  
    }  
}
```

Valores podem ser retornados por métodos

- Usamos métodos com tipo de retorno **void**
- O que você disser que retornará é bom que seja mesmo retornado!
- Você pode passar mais de um valor para um método.

O Java passa por valor

- Isso significa passar por cópia

```
public class Aula003 {  
  
    void go(int z) { //Declaração do método go com um parâmetro inteiro  
        z = 0;  
    }  
  
    public static void main(String[] args) {  
        Aula003 foo = new Aula003();  
        int x = 7; // Declarando inteiro com o valor 7.  
        foo.go(x);  
        System.out.println("Valor de x: " + x);  
    }  
}
```

Pontos importantes até agora

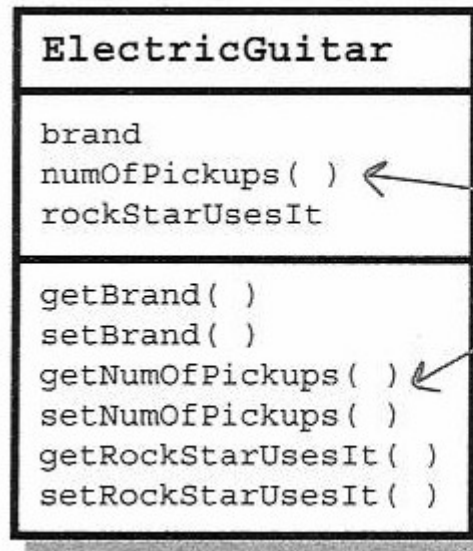
- As classes definem o que um objeto conhece e o que ele faz.
- As coisas que um objeto conhece são suas variáveis de instância (estado).
- As coisas que um objeto faz são seus métodos (comportamento).
- Os métodos podem usar variáveis de instância para que objetos do mesmo tipo possam se comportar diferentemente.

Pontos importantes até agora - II

- Um método pode ter parâmetros, o que significa que você pode passar um ou mais valores para ele.
- A quantidade e o tipo dos valores que você passar devem corresponder à ordem e tipo dos parâmetros declarados pelo método.

Coisas interessantes que você pode fazer com os parâmetros e tipos de retorno

- Métodos Getter (captura) e Setter (configuração).
- Formalizando acessadores e modificadores.



Nota: usar essas convenções de nomeação significa que você estará seguindo um importante padrão Java!

Continua ...

A series of horizontal lines in teal and light blue colors, spanning the width of the slide and partially overlapping the white area below.