

OO - Objetos, classes, atributos, métodos, estado, comportamento e identidade (Slide)



Prof. Bruno Gomes
bruno.gomes@ifrn.edu.br

Programação Orientada a Objetos

Introdução

- Programas eram lineares e com poucos módulos (Programação estruturada)
- Aumento da complexidade dos sistemas e difícil **reusabilidade** dos mesmos
- Criação de um novo Paradigma de Análise e Desenvolvimento de Sistemas:
 - **Programação Orientada a Objetos**

Introdução

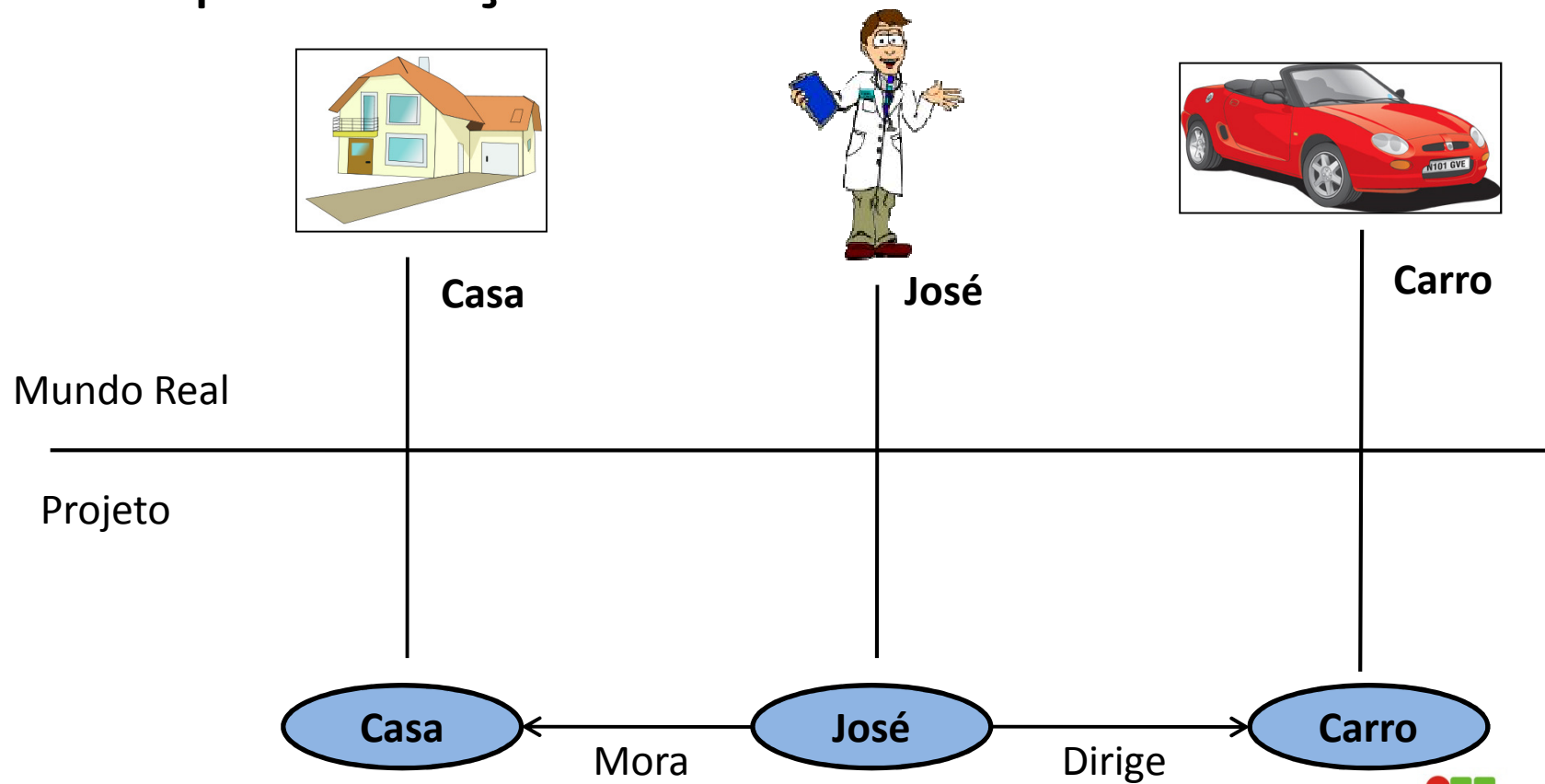
- No mundo real, pensamos em **conceitos** e em **entidades concretas e abstratas**
- Tudo é **objeto**:
 - Ex.: carro, computador, música, camisa, cliente, conta bancária, etc

Programação Orientada a Objetos

- Paradigma de Programação
 - Dominante nos dias atuais
- Substituiu as técnicas de programação procedimental (estruturada)
- “Fornece um mapeamento direto entre o mundo real e as unidades de organização utilizadas no projeto”
- Diversas unidades de software, chamadas de objetos, que interagem entre si
- Separa claramente a noção de **o que** é feito de **como** é feito

Programação Orientada a Objetos

- Representação:



Programação Orientada a Objetos

- Vantagens:
 - Flexibilidade
 - Reusabilidade
 - Robustez
 - Modularidade

Programação Orientada a Objetos

- Elementos básicos:
 - Objetos
 - Classes
 - Instâncias

Objetos

- Entidades concretas ou abstratas
- Tem características e podem executar ações
- “um objeto representa um item identificável, uma unidade ou entidade, individual, seja real ou abstrato, com uma regra bem definida”

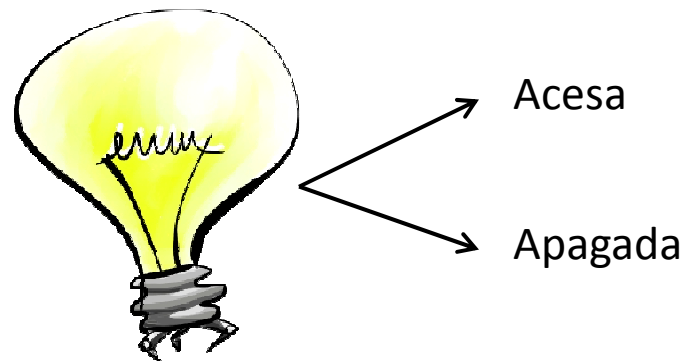
OBJETO = DADOS + OPERAÇÕES

- Possuem:
 - Estado
 - Comportamento
 - Identidade

Objetos

- Estado:
 - Define os estados possíveis que um objeto pode assumir
 - São os valores dos atributos (propriedades)

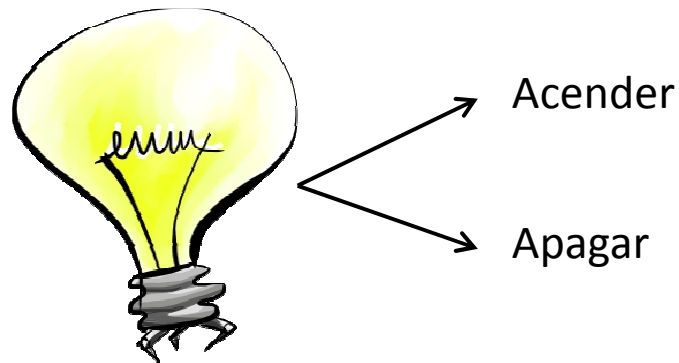
- Ex.:
 - Lâmpada



Objetos

- Comportamento:
 - São as funções que podem ser executadas por um determinado objeto
 - Corresponde aos métodos
 - O que você pode fazer com esse objeto

- Ex.:
 - Lâmpada

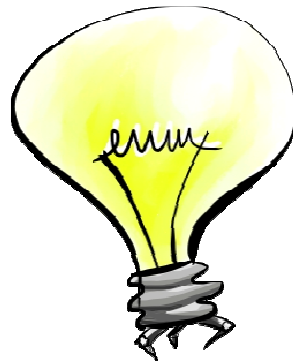


Objetos

- Identidade:
 - Um objeto é único, mesmo que o seu estado seja idêntico ao de outro

- Ex.:

- Lâmpada



→ Cód. De Fabricação 001; Incandescente;

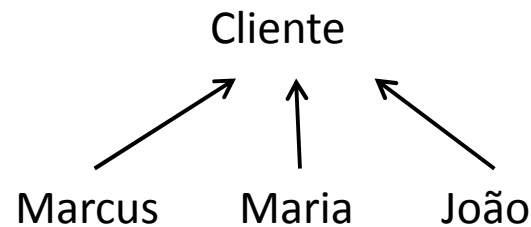
→ Cód. De Fabricação 002; Incandescente;

Classes

- Modelo ou esquema a partir do qual os objetos são criados (instanciados)
- Modelam os objetos definindo:
 - Tipo de dados que o objeto armazena, ou seja, os estados possíveis que ele pode assumir (**atributos**)
 - Tipos de operações que podem ser executadas pelo objeto, ou seja, o seu comportamento (**métodos**)
- Abstração de objetos de características semelhantes (molde)
- É a essência do objeto

Classe

- Objetos são instâncias de classes



- Lembrando: Todo código Java está dentro de uma Classe
- A biblioteca Java padrão fornece milhares de classes para vários propósitos

Classes em Java

- Declaração de uma classe em Java:

```
[<modificadores da classe>] class <nome_classe>  
    [extends <nome_superclasse>]  
    [implements <inteface_1>, <interface_2>, ...] {  
  
    // Variáveis e métodos da classe  
  
}
```

[] = Opcionais

< > = Identificadores e palavras reservadas

Classes em Java

- Exemplo de declarações de classe:

```
class Lampada {
```

```
    // Variáveis  
    // Métodos
```

```
}
```

```
public class Lampada {
```

```
    // Variáveis  
    // Métodos
```

```
}
```

Atributos em Java

- Definem as características do objeto

```
[<modificadores_atributo>] <tipo_atributo> <nome_atributo> [= valor_inicial];
```

[] = Opcionais

< > = Identificadores e palavras reservadas

Atributos em Java

- Exemplo:

```
public boolean estadoLampada = false;  
double valor;  
String marca = "fluorescente";
```

Métodos em Java

- Definem as ações que um objeto pode executar
- Sua definição corresponde a duas partes:
 - Assinatura
 - Corpo

```
[<modificadores_método>] <tipo_retorno> <nome_método> ([<parametros>]){  
  
    // Corpo do Método  
  
}
```

[] = Opcionais

< > = Identificadores e palavras reservadas

Métodos em Java

```
public void acenderLampada(){  
    estadoLampada = true;  
}
```

```
public int somar(int a, int b){  
    int resultado = a + b;  
    return resultado;  
}
```

Métodos em Java

- Exemplo Completo:

```
public class Lampada {  
    public boolean estadoLampada = false;  
  
    public void acenderLampada(){  
        estadoLampada = true;  
    }  
  
    public void apagarLampada(){  
        estadoLampada = false;  
    }  
  
    public boolean verEstadoLampada(){  
        return estadoLampada;  
    }  
}
```

1º Desafio – 22/03/10

- Prova: FCC - 2008 - TRT - 18ª Região (GO) - Técnico Judiciário - Tecnologia da Informação

```
int cont1 = 3;  
int cont2 = 2;  
int cont3 = 1;  
cont1 += cont3++;  
cont1 -= cont2;  
cont3 = cont2++;
```

- Após a execução deste trecho de código, os valores de cont1, cont2 e cont3 serão, respectivamente?