



Sejam bem-vindos!

PROGRAMAÇÃO ORIENTADA A OBJETOS



Diogenes Carvalho Matias



Formação:

- **Graduação: Sistemas de Informação;**
- **Especialista em: Engenharia e Arquitetura de Software;**
- **MBA EXECUTIVO EM BUSINESS INTELLIGENCE (em andamento);**
- **Mestrado Acadêmico em Engenharia de Computação (UPE em andamento);**

Maiores informações :



[Linkedin](#)

Programação Orientada a Objetos

O termo Programação Orientada a Objetos (POO) foi criado por Alan Kay, autor da linguagem de programação Smalltalk.

Mas mesmo antes da criação do Smalltalk, algumas das idéias da POO já eram aplicadas, sendo que a primeira linguagem a realmente utilizar estas idéias foi a linguagem Simula 67, criada por Ole-Johan Dahl e Kristen Nygaard em 1967.

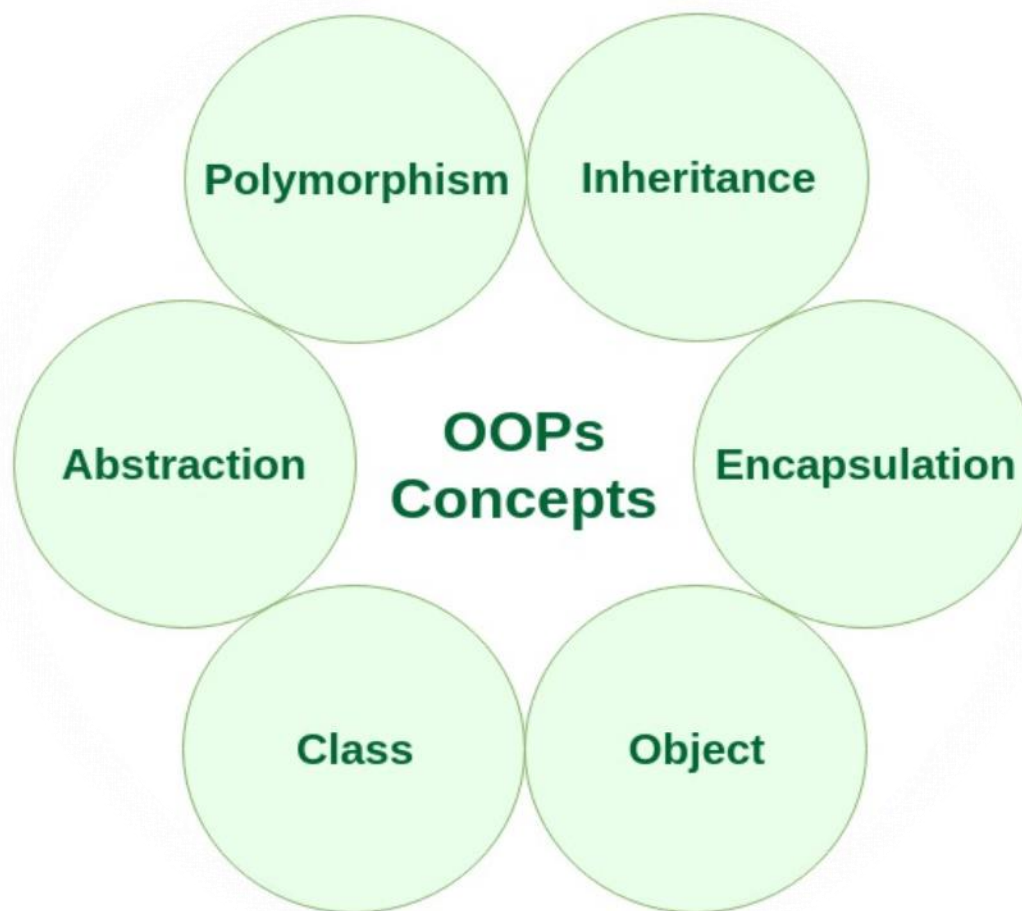
Programação Orientada a Objetos

Criada então para resolver e tentar aproximar o mundo real e o mundo virtual: a ideia fundamental é tentar simular o mundo real dentro do computador. Para isso, nada mais natural do que utilizar objetos, afinal, nosso mundo é composto de objetos.

Programação Orientada a Objetos

Na Programação Orientada a Objetos, o programador é responsável por moldar o mundo dos objetos, e definir como os objetos devem interagir entre si.

Programação Orientada a Objetos



Programação Orientada a Objetos

Abstração

A abstração consiste em um dos pontos mais importantes dentro de qualquer linguagem Orientada a Objetos.

Como estamos lidando com uma representação de um objeto real (o que dá nome ao paradigma), temos que imaginar o que esse objeto irá realizar dentro do Sistema.

Programação Orientada a Objetos

Classe

A palavra classe vem da taxonomia da biologia. Todos os seres vivos de uma mesma classe biológica têm uma série de atributos e comportamentos em comum, mas não são iguais, podem variar nos valores desses atributos e como realizam esses comportamentos.

Programação Orientada a Objetos

Classe

Uma classe é uma forma de definir um tipo de dado em uma linguagem orientada a objeto. Ela é formada por dados e comportamentos.

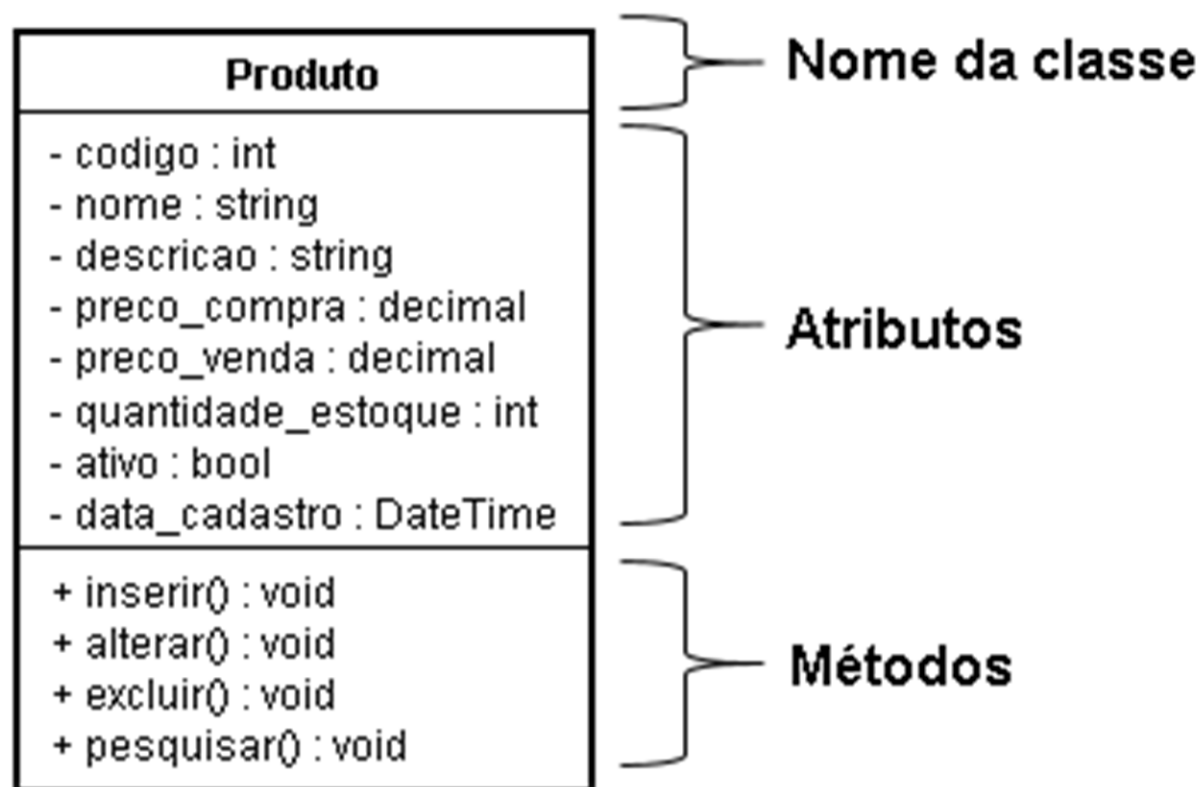
Programação Orientada a Objetos

Classe

Para definir os dados são utilizados os atributos, e para definir o comportamento são utilizados métodos. Depois que uma classe é definida podem ser criados diferentes objetos que utilizam a classe.

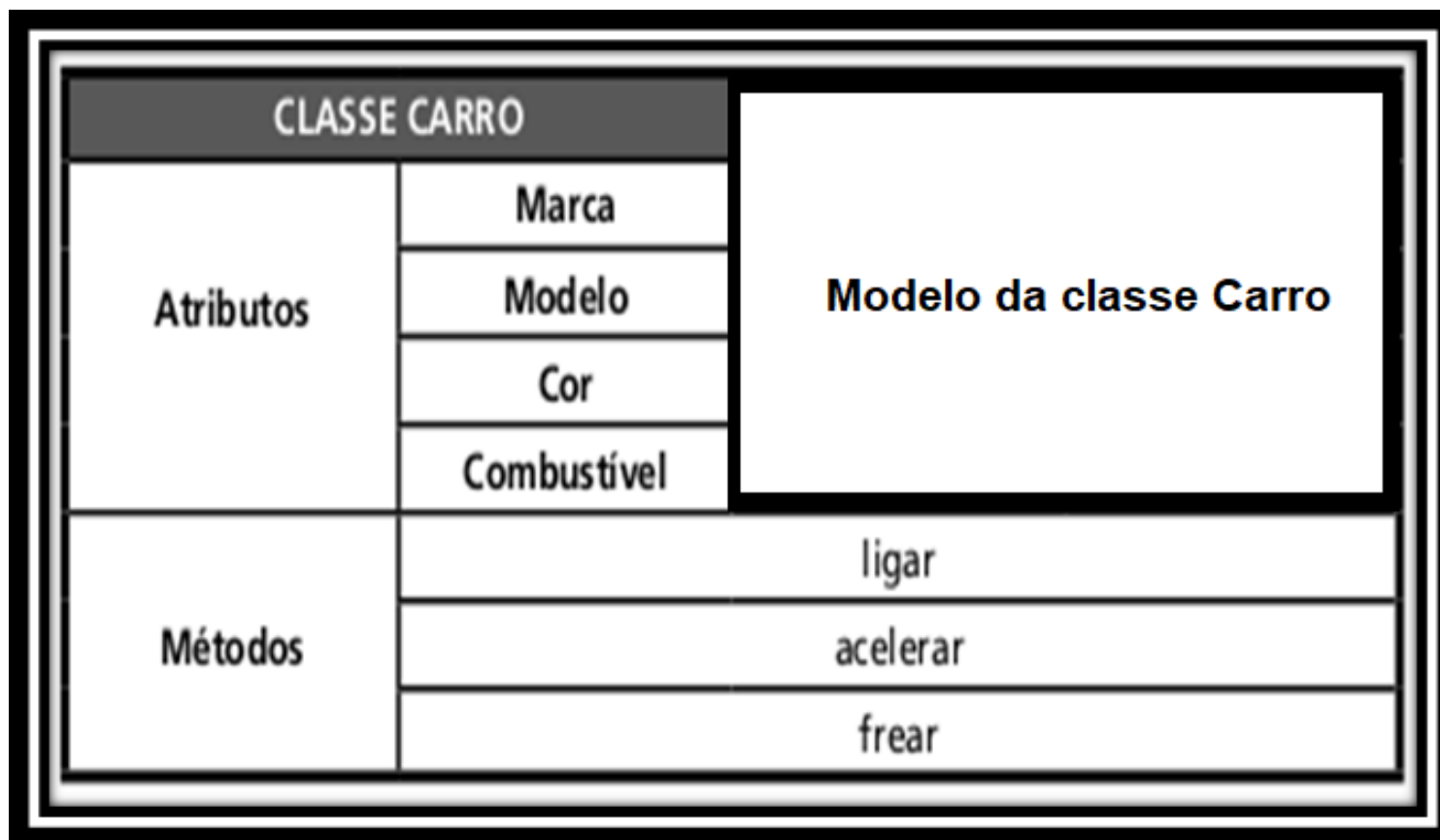
Programação Orientada a Objetos

Classe



Programação Orientada a Objetos

Classe



Programação Orientada a Objetos

Classe



Programação Orientada a Objetos

Classe



```
class Casa{
```

```
}
```

Programação Orientada a Objetos

Atributos

Um atributo consiste em um dado ou informação de estado, para o qual cada tem seu próprio valor. Existem dois tipos de atributos em um sistema orientado a objetos: os atributos de objetos e os atributos de classes.

```
class Casa{  
    private int tamanho;  
    private String endereco;  
    private int quantidade_quartos;  
}
```

Programação Orientada a Objetos

Métodos

São similares a procedimentos e funções e consistem nas descrições das operações ou seja ações a ser executada.

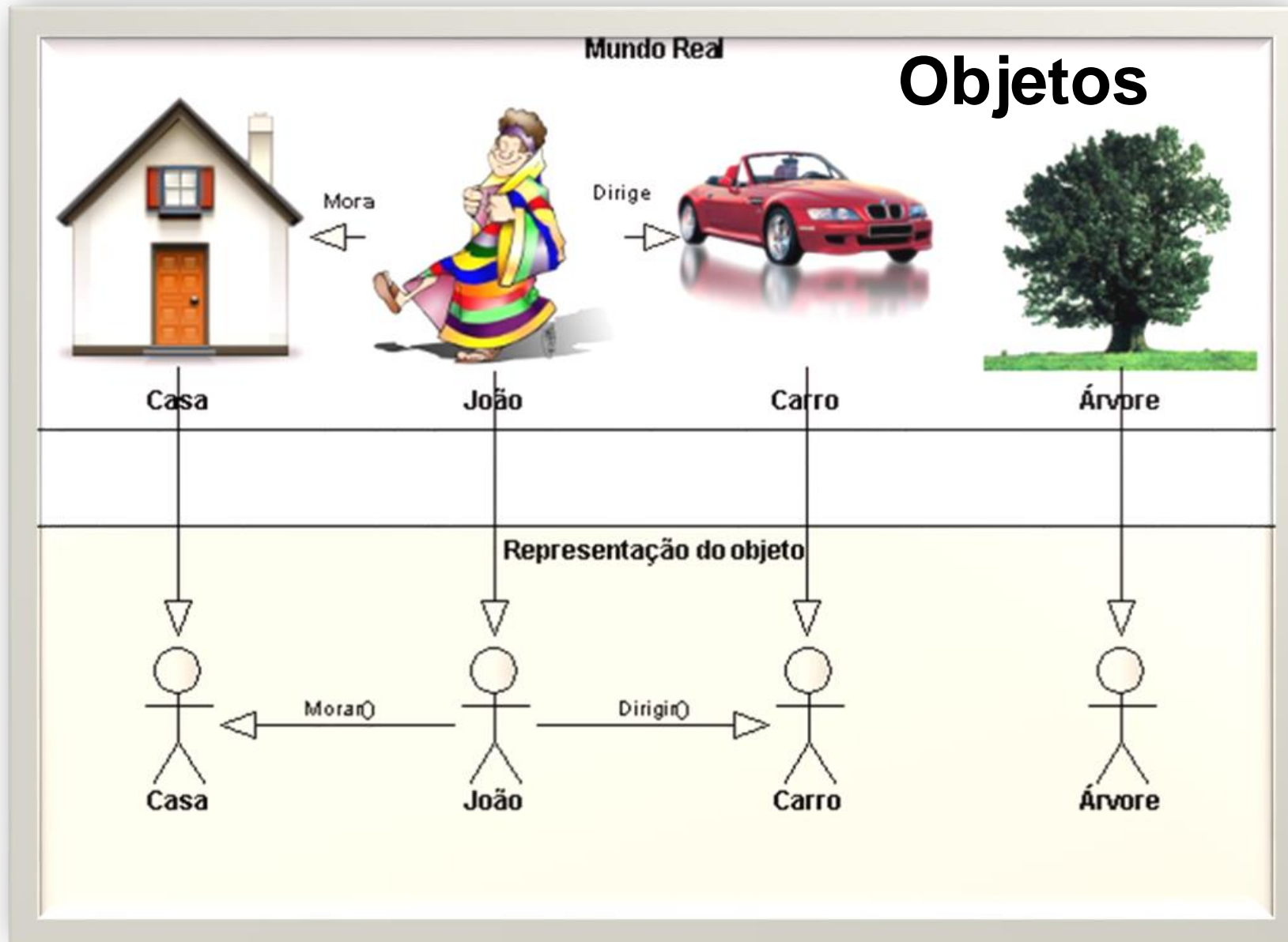
```
class Casa{  
    private int tamanho;  
    private String endereco;  
    private int quantidade_quartos;  
    public void acende_luz(){  
        ...  
    }  
    public double consume_energia(double conta_luz){  
        ...  
    }  
}
```


Programação Orientada a Objetos

Objetos

Refere-se a um modelo materializado de uma classe ,que passa a existir a partir de uma instância da classe.

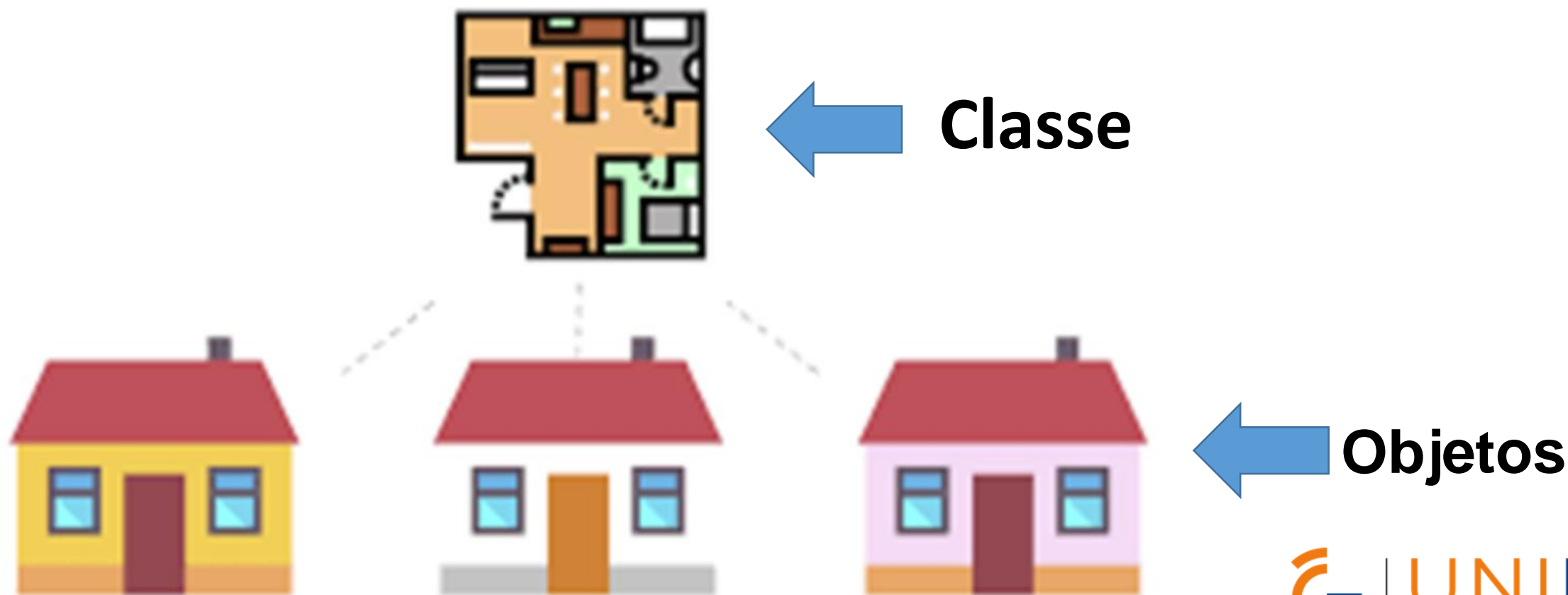
Programação Orientada a Objetos



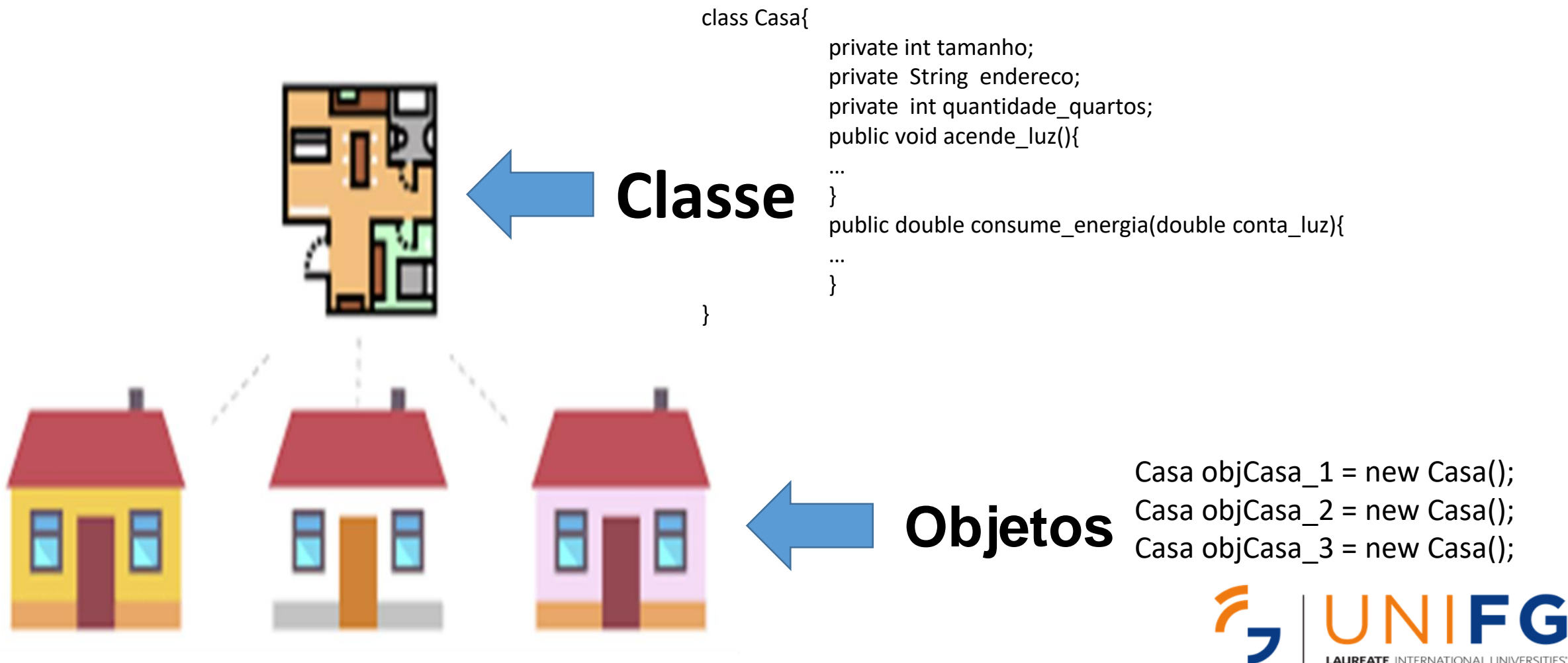
Programação Orientada a Objetos

CLASSE CARRO		OBJETO CARRO A	OBJETO CARRO B
Atributos de objeto	Marca	Ford	Mitsubishi
	Modelo	Fiesta	L-200
	Cor	branco	azul royal
	Combustível	gasolina	diesel
Métodos	ligar		
	acelerar		
	frear		

Programação Orientada a Objetos



Programação Orientada a Objetos



Programação Orientada a Objetos

Obs: arquivo Casa.java

```
class Casa{
    private int tamanho;
    private String endereco;
    private int quantidade_quartos;
    private double consumoLuz;
    public void acende_luz(){
        System.out.println("Luz esta ligada!")
    }
    public double consume_energia(double conta_luz){
        this.consumoLuz = conta_luz;
        return this.consumoLuz;
    }
}
```

Obs: arquivo Meu_programa_Diogenes.java

```
class Meu_programa_Diogenes{
    public static void main(String[] args) {

        Casa objCasa_1;
        Casa objCasa_2;
        Casa objCasa_3 ;

        objCasa_1 = new Casa();
        objCasa_2 = new Casa();
        objCasa_3 = new Casa();

        objCasa_1.endereço = "Avenida General Manoel Rabelo";
        objCasa_1.quantidade_quartos = 3;
        objCasa_1.tamanho = 120;
        objCasa_1.consume_energia(234);

    }
}
```

Programação Orientada a Objetos

Exercício para ser feliz

1- Pegando como base a figura abaixo faça a implementação da classe carro com seus respectivos objetos, exibindo o resultado na classe que tem o método Main.

CLASSE CARRO		OBJETO CARRO A	OBJETO CARRO B
Atributos de objeto	Marca	Ford	Mitsubishi
	Modelo	Fiesta	L-200
	Cor	branco	azul royal
	Combustível	gasolina	diesel
Métodos	ligar		
	acelerar		
	frear		

Programação Orientada a Objetos

Exercício para ser feliz

2- Pegando como base a figura abaixo faça a implementação da classe produto com seus respectivos objetos, exibindo o resultado na classe que tem o método Main.

