Sumário

[Paradigmas de Programação 1](#_Toc78298940)

[Programação Orientada a Objetos 2](#_Toc78298941)

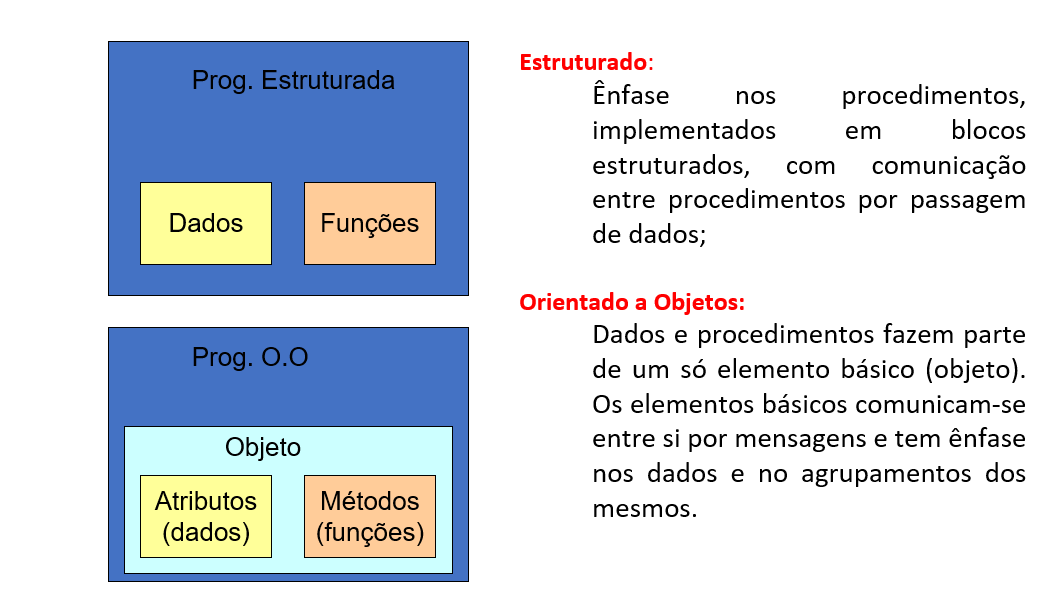
[POO – Por que? 2](#_Toc78298942)

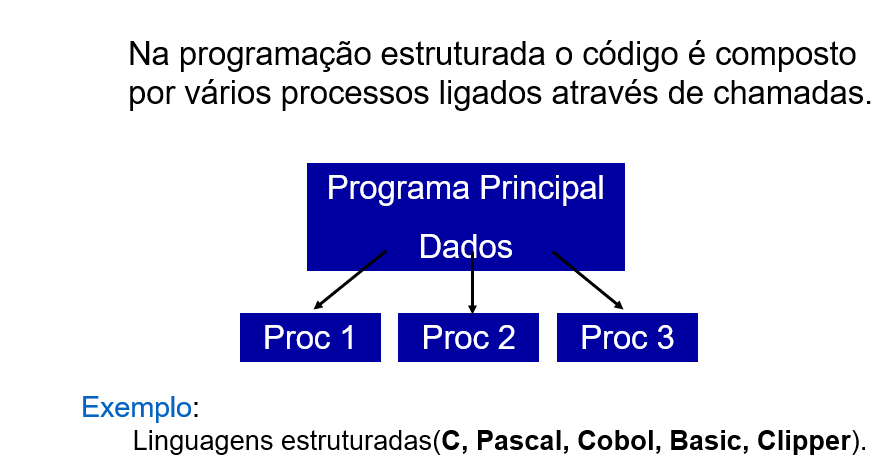
***Conteúdo:***

* **Paradigmas de Programação**
* **Conceitos fundamentais de POO**
* **Classes e Objetos**

# Paradigmas de Programação

* **Um paradigma de programação fornece (e determina) a visão que o programador possui sobre a estruturação e execução do programa**
* **Assim como ao resolver um problema podemos adotar uma entre variadas metodologias, ao criar um programa podemos adotar um determinado paradigma de programação para desenvolvê-lo.**

****

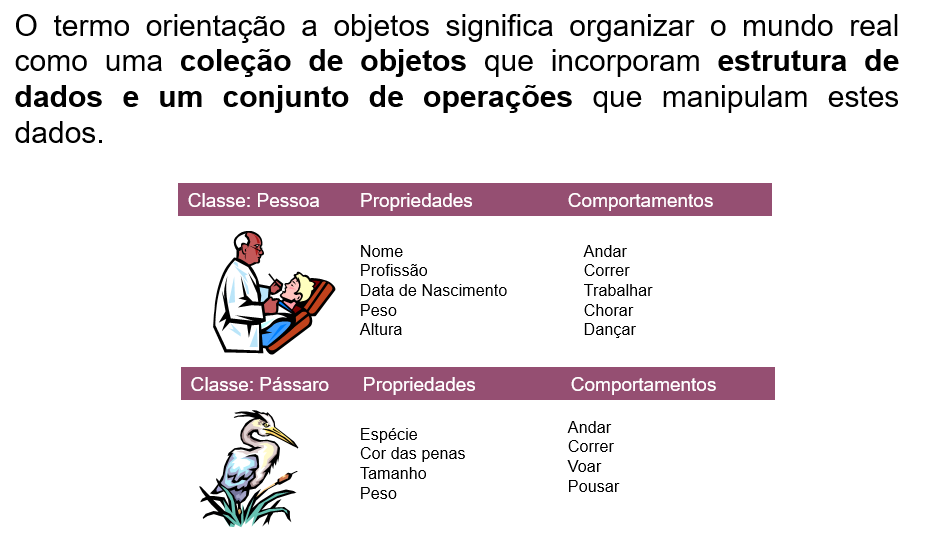
****

# Programação Orientada a Objetos

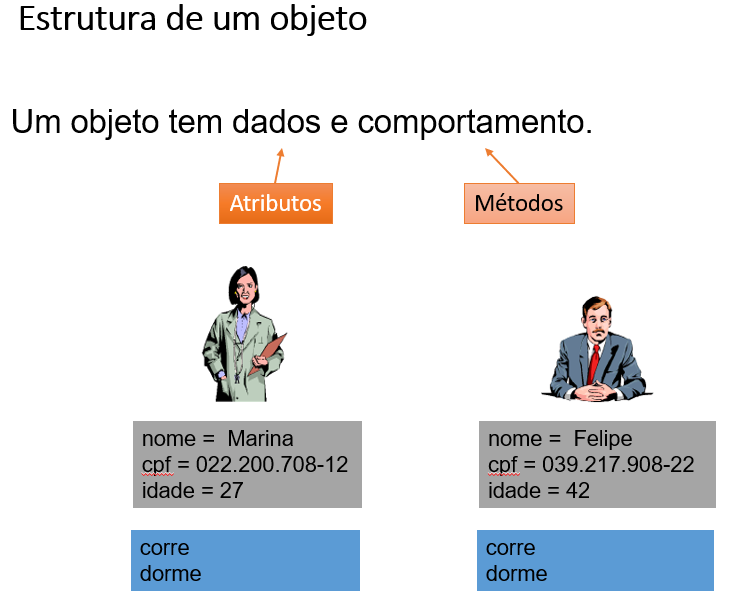
* **Diminuiu a distância entre mundo real e solução computacional.**
* **programa = dados + procedimentos sobre os dados**
* **Principais características:**
  + **Classes e objetos, encapsulamento, herança e polimorfismo.**
* **Programação se dá pela comunicação entre objetos.**
* **Pouco ou nenhum dado global.**
* **Linguagens Orientadas a Objetos:**
  + **JAVA, C++, Object Pascal, C#, Python e Ruby.**

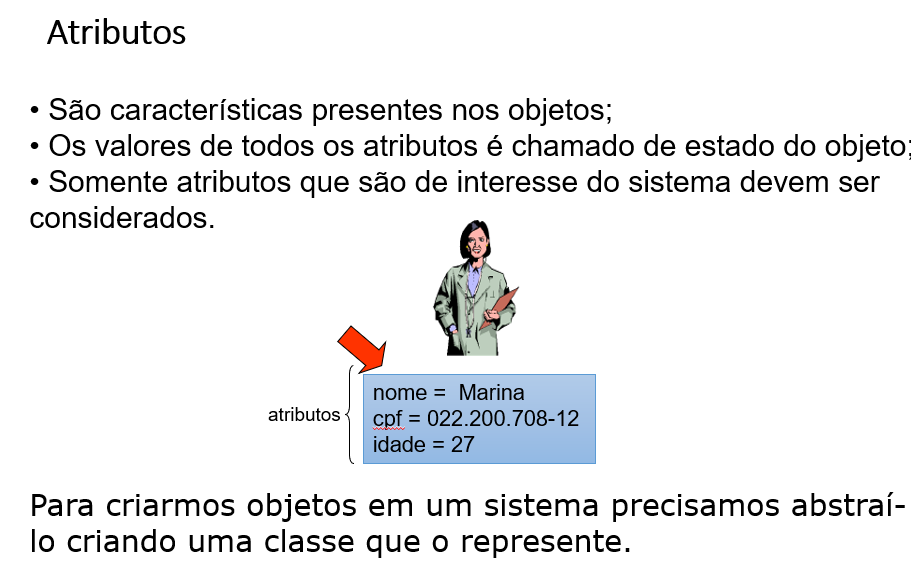
## POO – Por que?

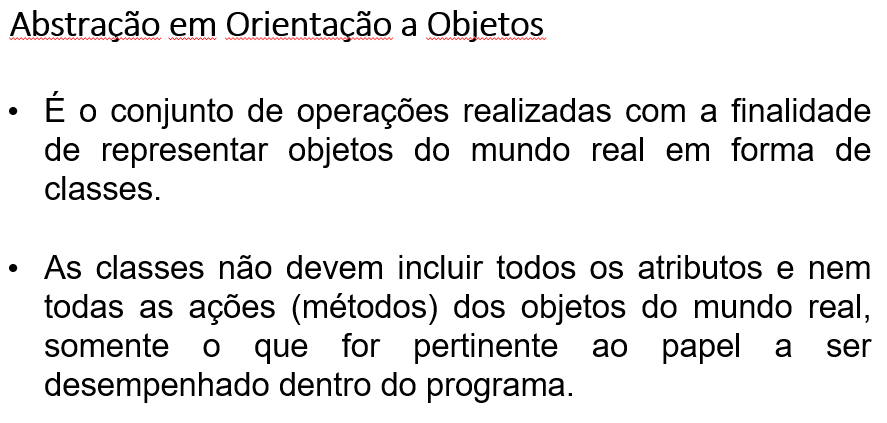
* **A metodologia Orientação a Objetos é baseada em “*objetos do mundo real*”, e por este motivo é mais intuitiva, pois oferece recursos como: objetos e atributos, classes e membros, estruturas e componentes, ação e reação.**
* **Em um sistema orientado a objetos, os dados e todas as operações, que manipulam esses dados, são agrupados em estruturas chamadas de “*classes*”.**
* **Os principais problemas do software hoje são:**
  + **Diminuir o custo e o tempo da mudança**
  + **Aumentar a capacidade e facilidade de adaptação**
* **Objetos são especialmente bons para:** 
  + **Reduzir o tempo necessário para adaptar um sistema existente (reação mais rápida à mudanças no seu ambiente de negócio)**
  + **Reduzir o esforço, a complexidade e os custos associados à mudança**

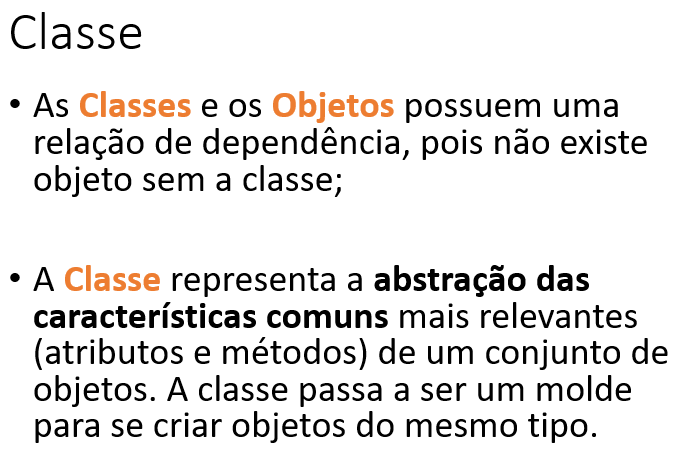
****

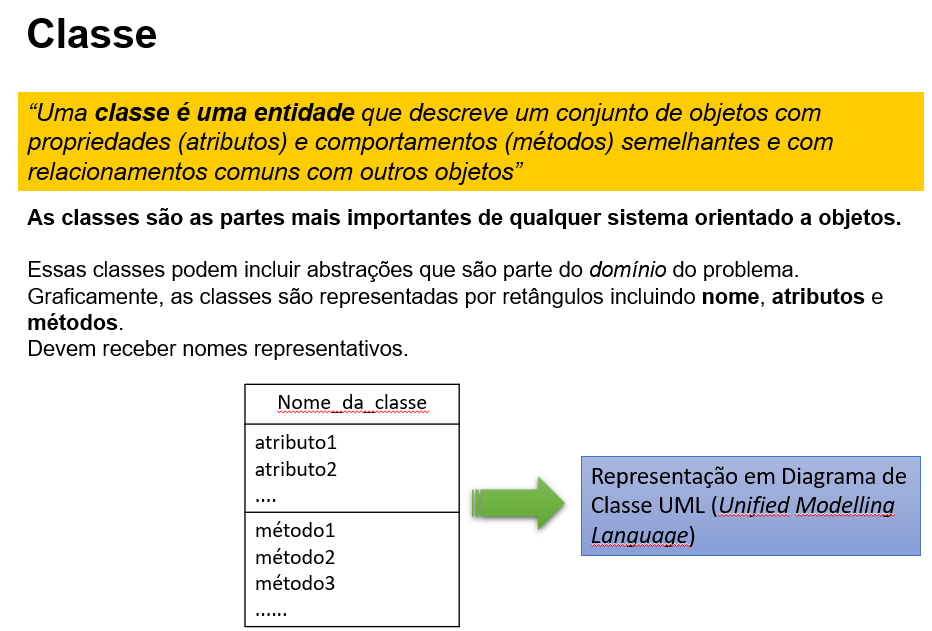
****

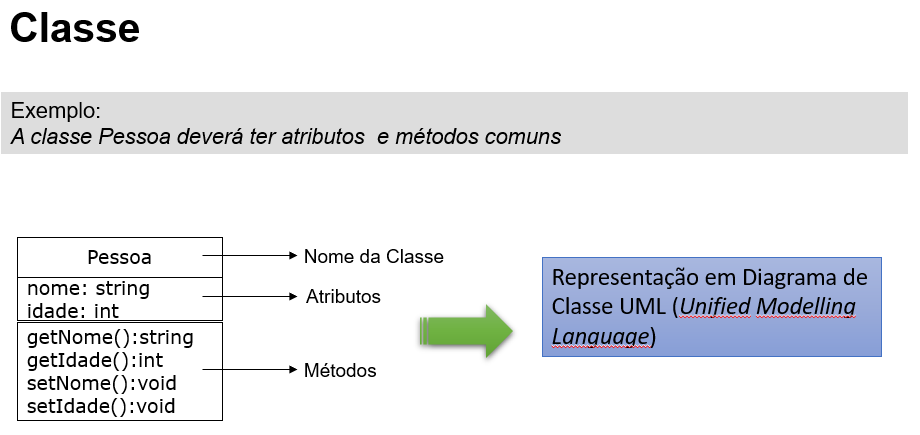
****

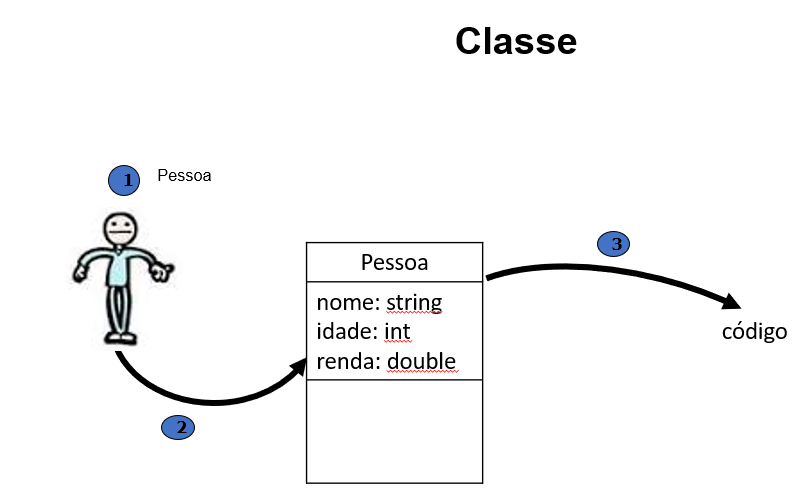
****

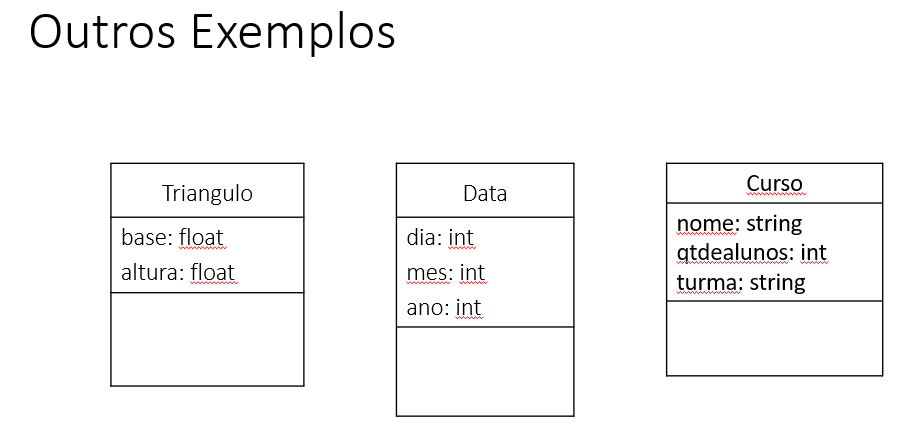
****

****

****

****

****

****

1. **Crie um diagrama de classes UML para abstrair os atributos dos seguintes objetos:**

**a) Eletrodoméstico**

**b) Carro**

**c) Programador**