

Programação orientada a objetos

- Introdução
- Paradigma de objetos e mensagens
- Classes e instâncias

Paradigma de programação

- Programação orientada a objetos - novo paradigma de programação.
- Paradigma: conjunto de teorias, métodos e padrões que juntos representam uma forma de organizar o conhecimento
- Em programação: modelo, padrão ou estilo de programação suportado por linguagens que agrupam certas características comuns.

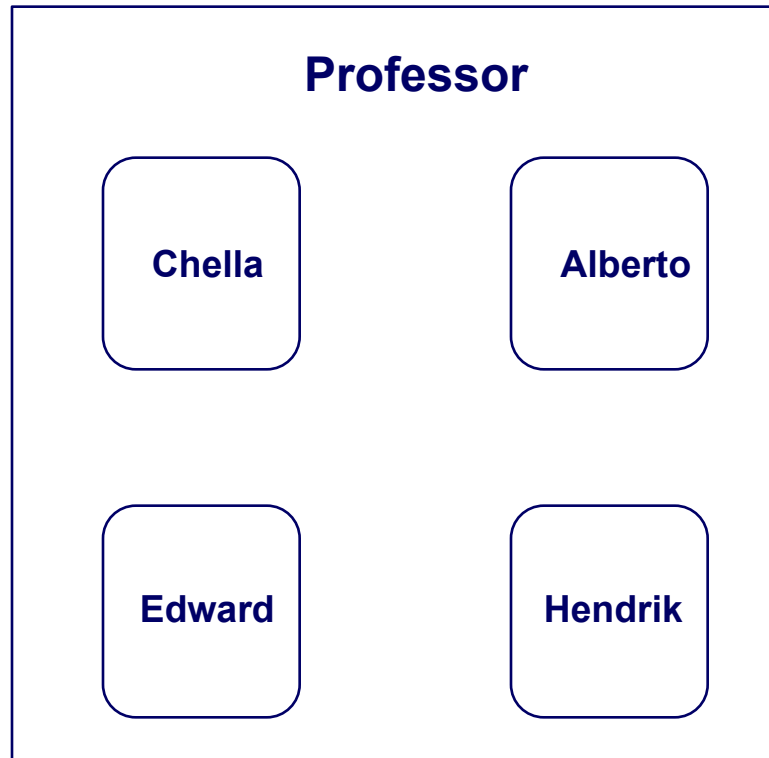
Inadequação das linguagens imperativas para construção de programas de grande porte

- Falta de correspondência entre o programa e o mundo real
- Organização interna dos programas
 - **Procedimentos versus dados**
 - **Problema das variáveis globais**

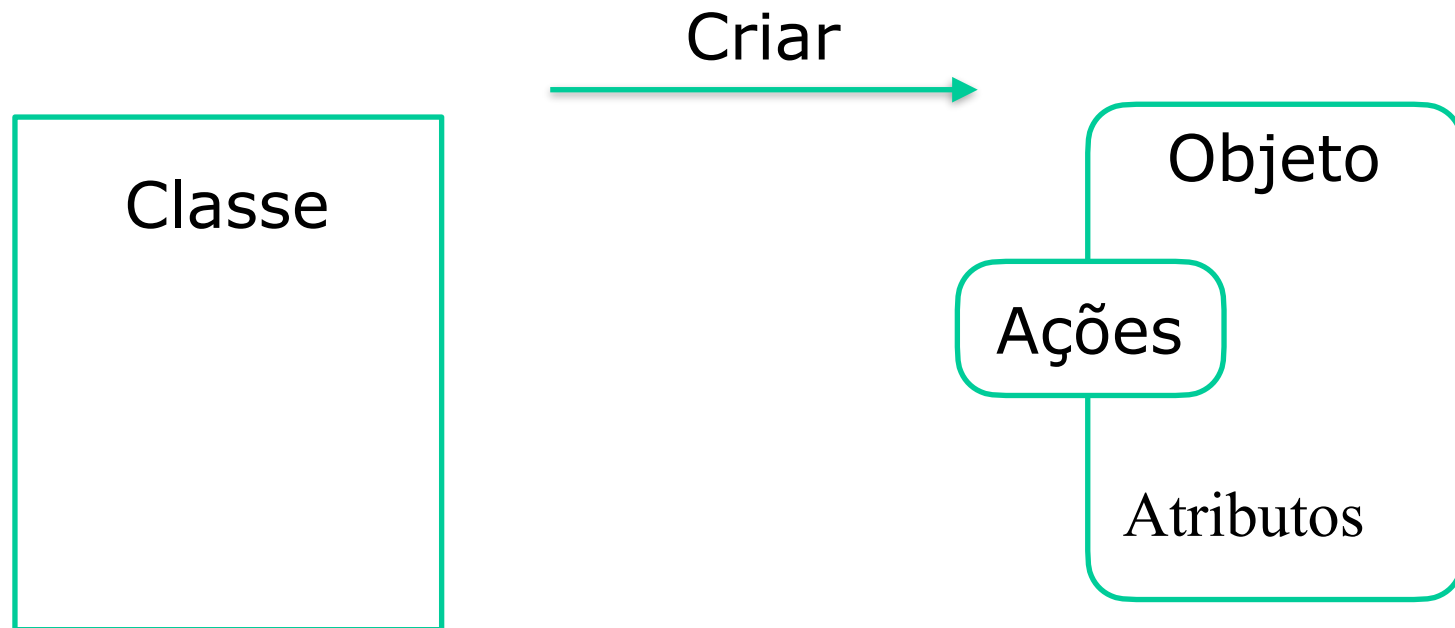
Objetos do mundo real agrupados em classes



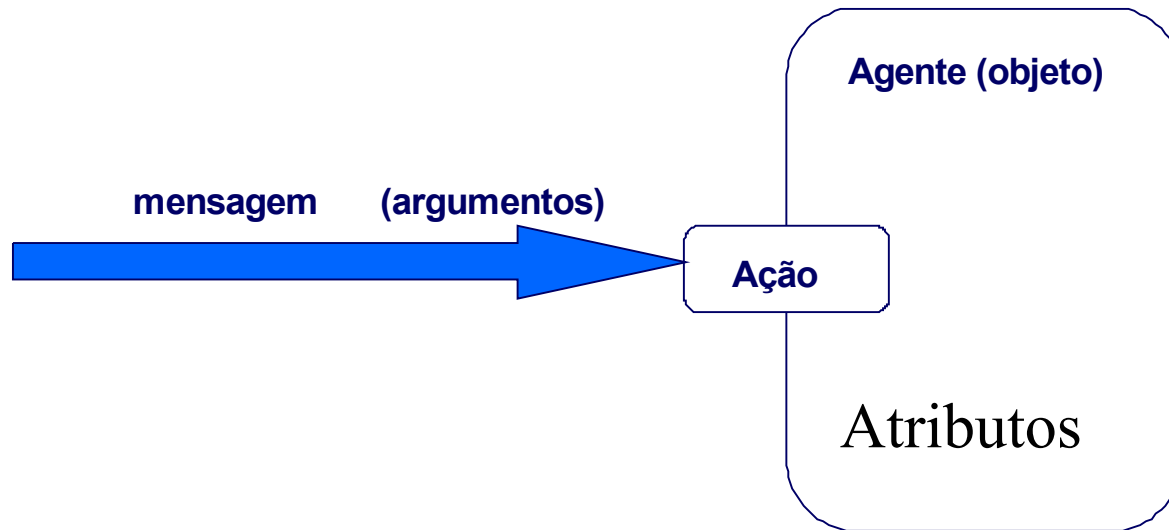
Objetos do mundo real agrupados em classes



Paradigma de objetos e mensagens



Paradigma de objetos e mensagens



Paradigma de objetos e mensagens

Ações:

Estudar
Assistir aulas
Fazer avaliações

Atributos:

Matricula
Nome
Curso



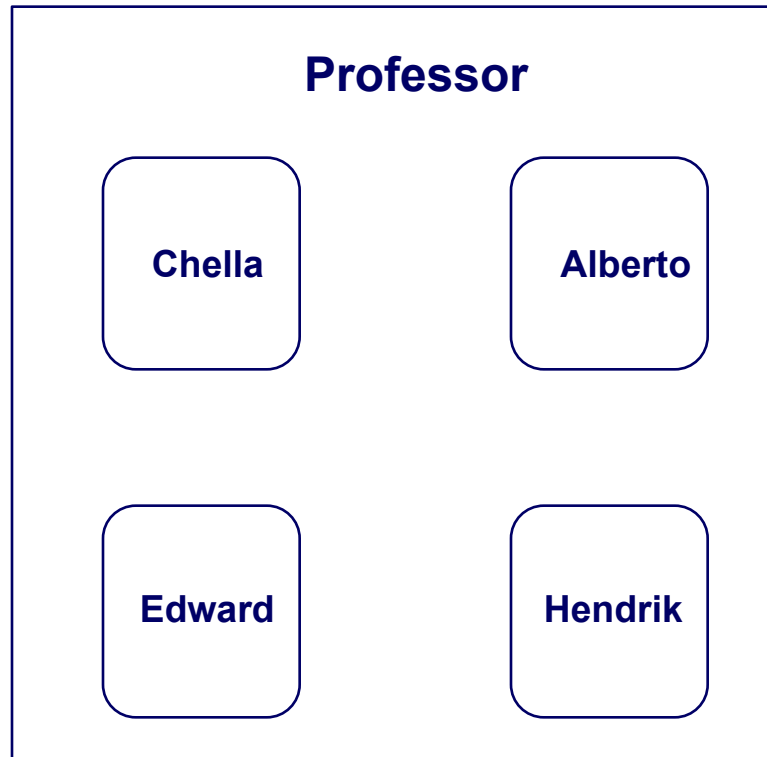
Objetos do mundo real agrupados em classes

Ações:

Pesquisar
Ministrar aulas
Elaborar avaliações

Atributos:

SIAPE
Nome
Salário



Paradigma de objetos e mensagens

- Em programação imperativa quem ficava responsável pelos atributos?
 - **As variáveis**
- Quem ficava responsável pelas operações?
 - **As funções (procedimentos)**
- Em PI, criamos funções para alterar o estado do processamento representados pelas variáveis

Paradigma de objetos e mensagens

Estado inicial

var1



em PI

var3



var2



var4



var5



var6



Paradigma de objetos e mensagens

var1



var3



var2



var4



var5

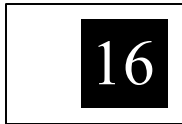


var6



Paradigma de objetos e mensagens

var1



var3



var2



var4



var5



var6



Paradigma de objetos e mensagens

var1



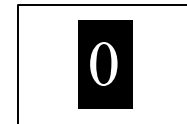
var3



var2



var4



var5



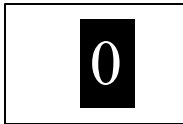
var6



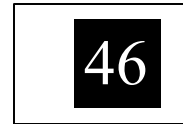
Paradigma de objetos e mensagens

Estado final

var1



var3



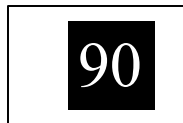
var2



var4



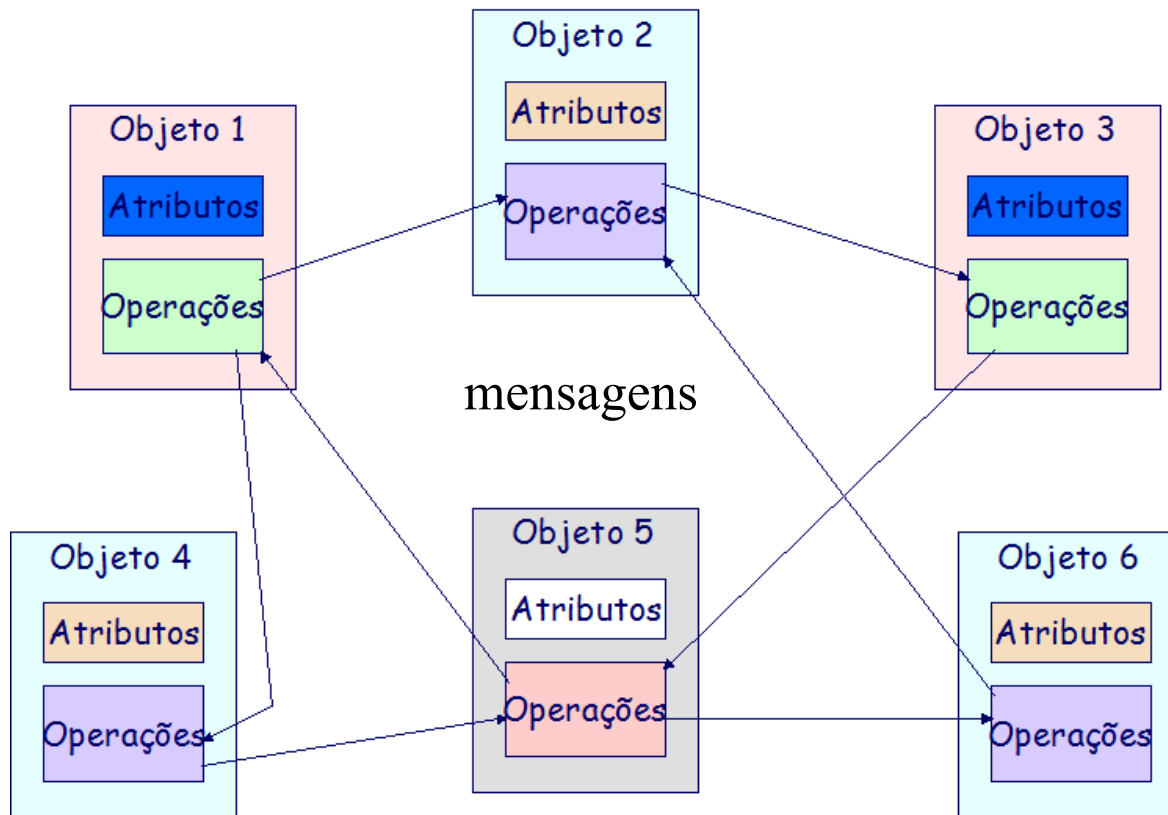
var5



var6



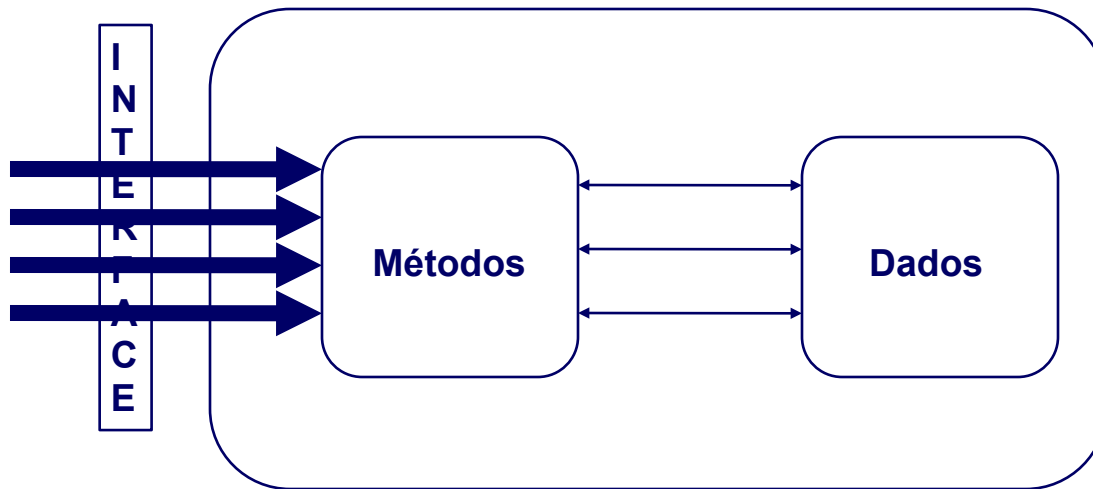
Paradigma de objetos e mensagens



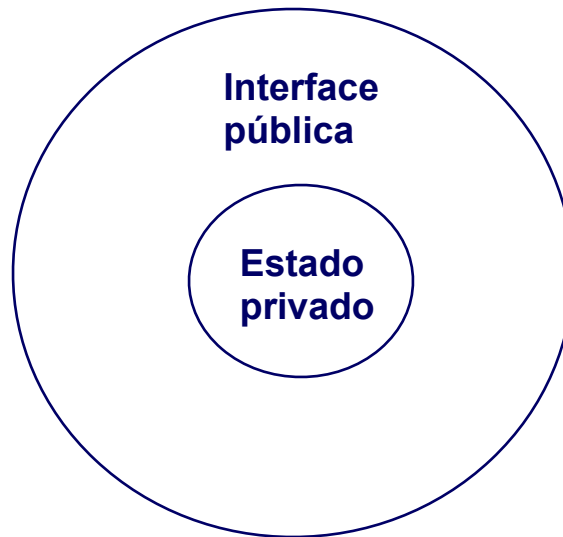
Objetos

- Um objeto possui:
 - um estado que pode ser alterado ao longo do seu tempo de vida.
 - Um comportamento bem definido (formado pelo conjunto de operações da sua interface pública)
 - Uma identidade única (propriedade que o distingue de outros objetos)

Objetos



Objetos



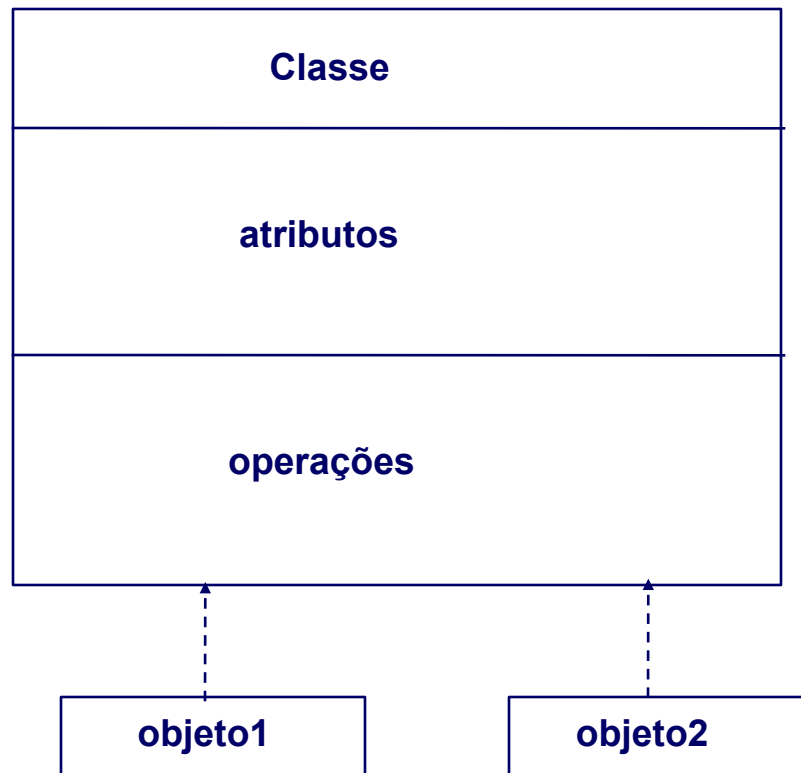
Classes

- Em PI, criamos funções para alterar o estado do processamento
 - **Estado representado por variáveis simples e compostas (vetores, structs)**
- Em POO, criamos classes para instanciar (inicializar) o objeto em determinado estado (estado inicial)
- Uma classe serve para definir um objeto de um determinado tipo

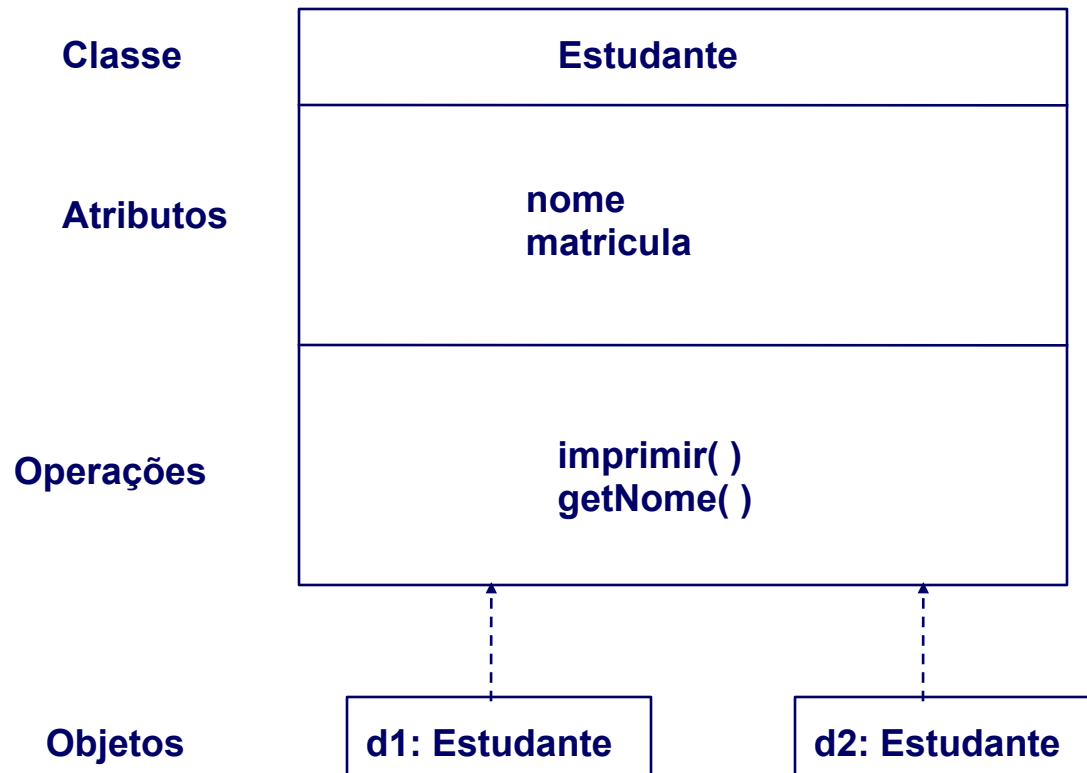
Classes

- Classe é a **descrição** de um molde que especifica as propriedades e o comportamento para um conjunto de objetos similares.
- Todo objeto é instância de apenas uma classe.
- Atributos e operações são parte da definição de uma classe.

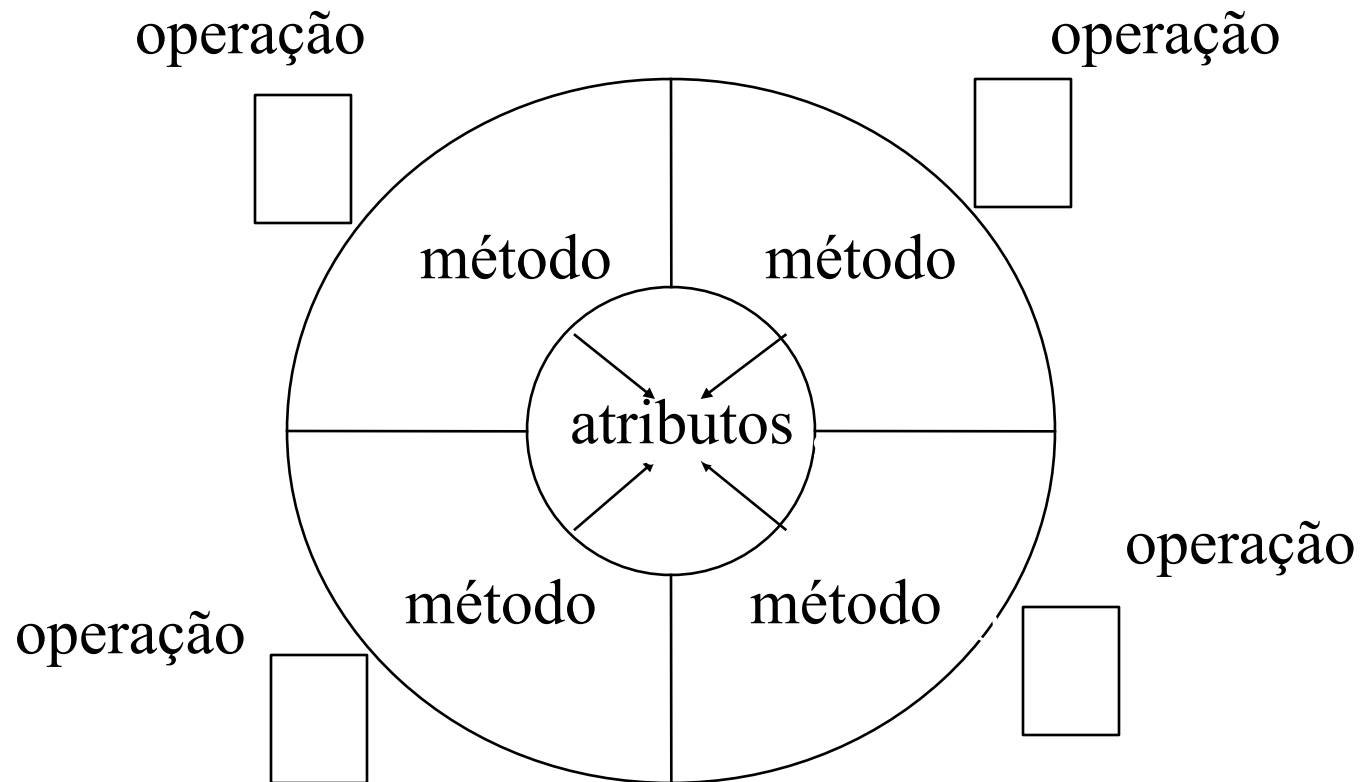
Classes



Classes



Classes



Programação em OO

- Definir classes
- Instanciar objetos
- Trocar mensagens entre objetos
- Parar o processamento em um estado final

Declaração de uma classe Python

```
class Estudante (object) :  
    """attributes: dataDeRegistro, nome"""  
    def getData(self):  
    def getNome(self):  
    def imprimir(self):  
    def __init__(self):
```

Métodos em Python

```
def getNome(self):  
    return self.nome
```

Criação de objetos em Python

```
d1 = Estudante()
```

```
d1.imprimir()
```

Normalmente feito em um programa principal

Exercícios

- Estabeleça diferenças entre tipo de dados e tipo abstrato de dados
- Estabeleça diferenças entre tipo abstrato de dados e classe
- Estabeleça diferenças entre classe e objeto
- Estabeleça diferenças entre função e método
- Estabeleça diferenças entre atributo e variável