

# Programação II

Prof. Alcemir Rodrigues Santos

*Atributos, Métodos, Classe, Instância*



## Agenda

- Paradigma OO
- Objetos
- Classes
- Construindo Classes



9/17/19

Programação II | Alcemir Santos

2



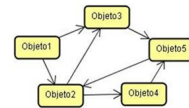
## Paradigma de Programação OO

- Paradigma de Programação é a maneira como pensamos e fazemos nossos programas.
- "A Programação Orientada a Objetos (POO) é um paradigma de programação de computadores que usa os conceitos de Objetos e Classes como elementos centrais para representar e processar dados usados nos programas."



## Orientação a Objetos

- Utilizando o paradigma OO, um programa é encarado como um conjunto de objetos que interagem entre si através de mensagens;
- Cada objeto mantém informações internas sobre si, chamadas de atributos;
- Um sistema Orientado a Objetos é constituído de um conjunto de objetos que interagem entre si em busca de um objetivo em comum.



## O que são objetos ?

- No mundo real:
  - Tudo aquilo que se pode tocar
  - Tudo aquilo que se pode sentir
  - Tudo aquilo que se pode conceituar
- Todo objeto possui propriedades e comportamentos
- Uma definição de objeto
  - "Os objetos são usados para representar entidades do mundo real ou computacional"
  - Poder de abstração: Trazer para o computador tudo aquilo que é relevante e possa ser observado no mundo real



## O que são objetos ?

- Todos os objetos possuem características que os identificam e finalidades à quais os utilizamos.
- Essas características são chamadas de atributos
- Além disso, os objetos possuem comportamentos associados a eles
- Os comportamentos são chamados comumente métodos ou operações
- Dica para identificação de atributos ou métodos :
  - Atributos são geralmente designados por adjetivos
  - Métodos são geralmente designados por verbos



### Exemplo de objeto



- Características:
  - nome: Conceição
  - cor do cabelo: negro
  - biotipo: magro
- Comportamentos:
  - andar
  - correr
  - dirigir Carro



### Exemplo de objeto

- Características: (dados, atributos)
  - tipo: Ferrari
  - placa: KZE1018
  - cor: vermelha
  - número de portas: 2
- Comportamentos: (operações, métodos)
  - ligar
  - desligar
  - acelerar
  - frear



### Atividade

- Observe os objetos a sua volta, escolha dois deles e descreva de 3 a 5 características (atributos) e comportamentos (métodos) para cada um deles.



### O que são classes ?

- Nossa vida é organizada em torno de objetos!
- Pensamos e organizamos as coisas em função das características e finalidades
- Comumente criamos grupos de objetos, com base em suas características
- Por exemplo, quando falamos de uma pessoa, que características e comportamentos conseguimos enxergar?!

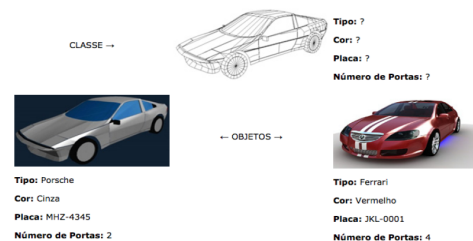


### O que são classes ?

- A modelagem e programação de um conjunto de objetos que possuem características (atributos) e comportamentos (métodos) comuns é feita na Programação Orientada a Objetos usando o conceito de Classe
- Cada classe funciona com um molde para a criação de objetos
- Os objetos são representações concretas (instâncias) de uma classe



### Exemplo de classe



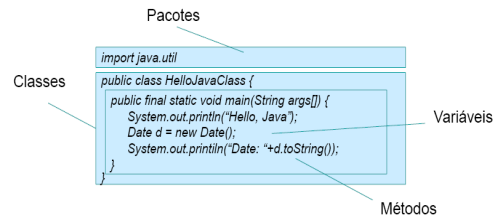
## Atividade

- Observando o exemplo anterior, imagine as seguintes classes:
  - Livro, Celular, Mamíferos, Aves e Flores.
- 1. Como seriam seus objetos?
- 2. Escreva apenas as características (ou atributos) das classes e objetos!



## Programa em Java

- Todos os programas em Java possuem quatro elementos básicos:



## Construindo classes

- Vamos considerar um contexto de uma escola, que precisa de um sistema para registrar alunos, professores, disciplinas e turmas
- Identifiquem pelo menos 3 classes desse contexto
- Para cada classe identificada, identifiquem pelo menos 4 atributos



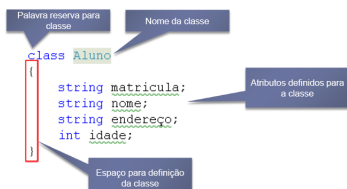
## Construindo classes

- Vamos agora construir uma classe utilizando a linguagem Java
- O objetivo nesse momento é apenas escrever e verificar os elementos iniciais
- Alguns aspectos serão desconsiderados nesse momento, mas serão retomados depois
- As classes são utilizadas em dois momentos:
  - Definição: Dizer quais são os atributos e métodos da classe
  - Uso: Criar instâncias a partir da classe



## Construindo classes: Definição

- Considerando a classe Aluno, com os atributos:
  - matrícula, nome, endereço e idade
- O código inicial obtido é o seguinte



## Construindo classes: Definição

- Agora vamos adicionar um comportamento para essas classes
- Os primeiros comportamentos que iremos adicionar são métodos utilizados para ler/escrever os valores dos Atributos
- São chamados de métodos get/set

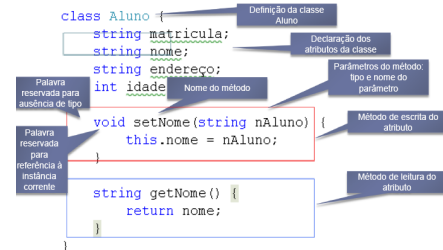


### Construindo classes: Definição

- Considere que no contexto apresentado temos a classe Aluno e que essa classe possui como característica o nome do aluno, que é um atributo do tipo String
- Definiremos um par de métodos do tipo get/set para o atributo Nome
- Lembrem-se que os métodos, correspondem as ações da classe, onde efetivamente serão colocadas as instruções do programa, dando "vida" as classes



### Construindo classes: Definição



### Construindo classes: Definição

- Agora para as demais atributos da classe do nosso exemplo e seguindo o padrão visto anteriormente, escreva os métodos get/set



### Construindo classes: Uso

- Uma vez definida a classe é possível utilizá-la na criação de instâncias (objetos)

```

public static void main(String[] args) {
    Aluno novoAluno = new Aluno();
    novoAluno.setNome("João");
    System.out.println("O nome do novo aluno é: " + novoAluno.getNome());
}

```

Construção de uma instância

Alteração do conteúdo do atributo Nome

Leitura do conteúdo atributo Nome

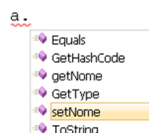


### Construindo classes: Uso

```

Aluno a = new Aluno();
a.setNome("João");

```



- Uma vez instanciado o objeto da classe é possível acessar suas informações
- Cada objeto utiliza uma quantidade de memória
- A alocação/desalocação da memória é feita automaticamente



### Construindo classes: Uso

- Utilizado o exemplo do sistema de controle de alunos faça as seguintes tarefas:
- Defina as classes restantes do sistema de controle de alunos
  - Classes
  - Atributos
  - Métodos (get/set)
- Construa 3 instância de cada classe e altere pelo menos um atributo para cada instância

