

FIA/P GRADUAÇÃO

DOMAIN DRIVEN DESIGN

Prof. Me. Thiago T. I. Yamamoto

#01 - ORIENTAÇÃO A OBJETOS

TRAJETÓRIA



Orientação a Objetos



#01 - AGENDA



- Apresentação da disciplina
- Orientação a Objetos
- Classes
 - Atributos e Comportamentos
- Relação de classes com objetos
- Modelo Visual

THIAGO T. I. YAMAMOTO



Mestre em Ciências, Gestão e Informática em Saúde pela Universidade Federal de São Paulo. Pós-graduado em Engenharia de Sistemas. Bacharel em Ciências da Computação pela Universidade Estadual Paulista - Unesp/Bauru. Mais de 10 anos de experiência na área de TI, como desenvolvedor de sistemas nas empresas: **Autbank, UOL e Ericsson Telecomunicações**. Professor em curso graduação e pós-graduação da FIAP - ministra várias disciplinas de desenvolvimento de sistemas. Certificado ITIL V3, PSM I e OCJA.

Apaixonado por ensinar!



DOMAIN DRIVEN DESIGN

- Preparar o profissional para o **Mercado de Trabalho**;
- Aprender os conceitos de **Programação Orientada a Objetos** com a **linguagem Java**;
- Desenvolver *back end* de aplicações **Java**;

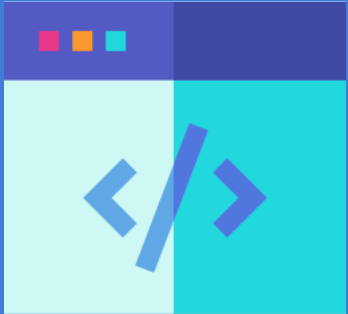


- Programação orientada a objetos;
- Plataforma Java;
- Ambiente de execução: Compile/Runtime;
- Detalhes da linguagem:
 - Tipos primitivos, operadores, controle de fluxo, loops e etc.;
- Java Beans;
- Arrays/Collections Framework;
- Tratamento de Erros;
- JDBC;
- Design Patterns;



METODOLOGIA

- Aulas “Hands On”;
- Projetos no **Github**;
 - <http://www.github.com/thiagoyama>
 - **Git** é um **versionador de arquivos**, será abordado em outra disciplina;

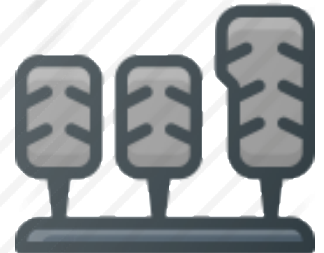


“Quem ouve, esquece. Quem vê, lembra. Quem faz, aprende.”
Provérbio chinês.



ORIENTAÇÃO A OBJETOS

- O que é Programação Orientada a Objetos?



- **Programação**
 - Uma **linguagem de programação** é um método padronizado para **expressar instruções** para um computador;
 - É um conjunto de **regras sintáticas** (gramatical) e **semânticas** (significado) usadas para definir um programa de computador;
- **Objeto**
 - Um objeto representa uma **entidade** que pode ser **física, conceitual** ou de **software**;
- **Programação Orientada a Objetos**
 - É um paradigma de **análise, projeto e programação** de sistemas de informação, baseado na composição e interação entre diversas unidades de software chamadas de **objetos**;

- **Cadeira** (material, cor, tem braço?, tem rodas?, etc.)
 - **Praia** (alumínio, pano, de deitar, etc.);
 - **Escritório** (ferro, estofado macio, preta, com braço, etc.);
 - **Rodas** (ferro, branca, com rodas, automática, manual, etc.);
 - **Banco**
 - Carro (couro, preto, regulável, etc.);
 - Praça (tijolo, verde, com encosto, etc.);
- **Bola** (material, formato, cor, etc.)
 - **Futebol** (couro, redonda, 40cm de diâmetro, branca, etc.);
 - **Tênis** (tecido, redonda, 5cm de diâmetro, amarela, etc.);
 - **Ping-Pong** (pvc, redonda, 1.5cm de diâmetro, branca, etc.);
 - **Futebol Americano** (couro, oval, 50cm de largura, marron, etc.);



- Bola (material, formato, cor, etc.) → Características ou Propriedades



Futebol

Material: couro;
Formato: redondo;
Tamanho: 40cm;
Cor: Branca;



Tênis

Material: tecido;
Formato: redondo;
Tamanho: 5cm;
Cor: Amarela;



Futebol Americano

Material: couro;
Formato: oval;
Tamanho: 50cm;
Cor: Marron;

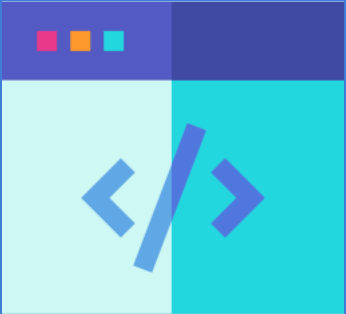


Basquete

Material: couro;
Formato: redondo;
Tamanho: 60cm;
Cor: Laranja;

PRÁTICA

- Busque ao menos **3** exemplos do cotidiano;
- Descreva as **propriedades** dos exemplos encontrados;



- Sistema de Caixa Eletrônico

- Objeto: Cliente

- Nome
- Endereço
- CPF
- RG



- Objeto: Conta Corrente

- Agência
- Número
- Saldo
- Cliente



- Sistema de E-Commerce

- Objeto: **Produto**
 - Nome
 - Descrição
 - Valor
- Objeto: **Estoque**
 - **Produto** ←
 - Quantidade
 - Prazo de Validade
- Objeto: **Cliente**
 - Cadastro
 - Senha do Cadastro
 - Nome
 - Endereço
 - CPF
 - RG





CLASSES

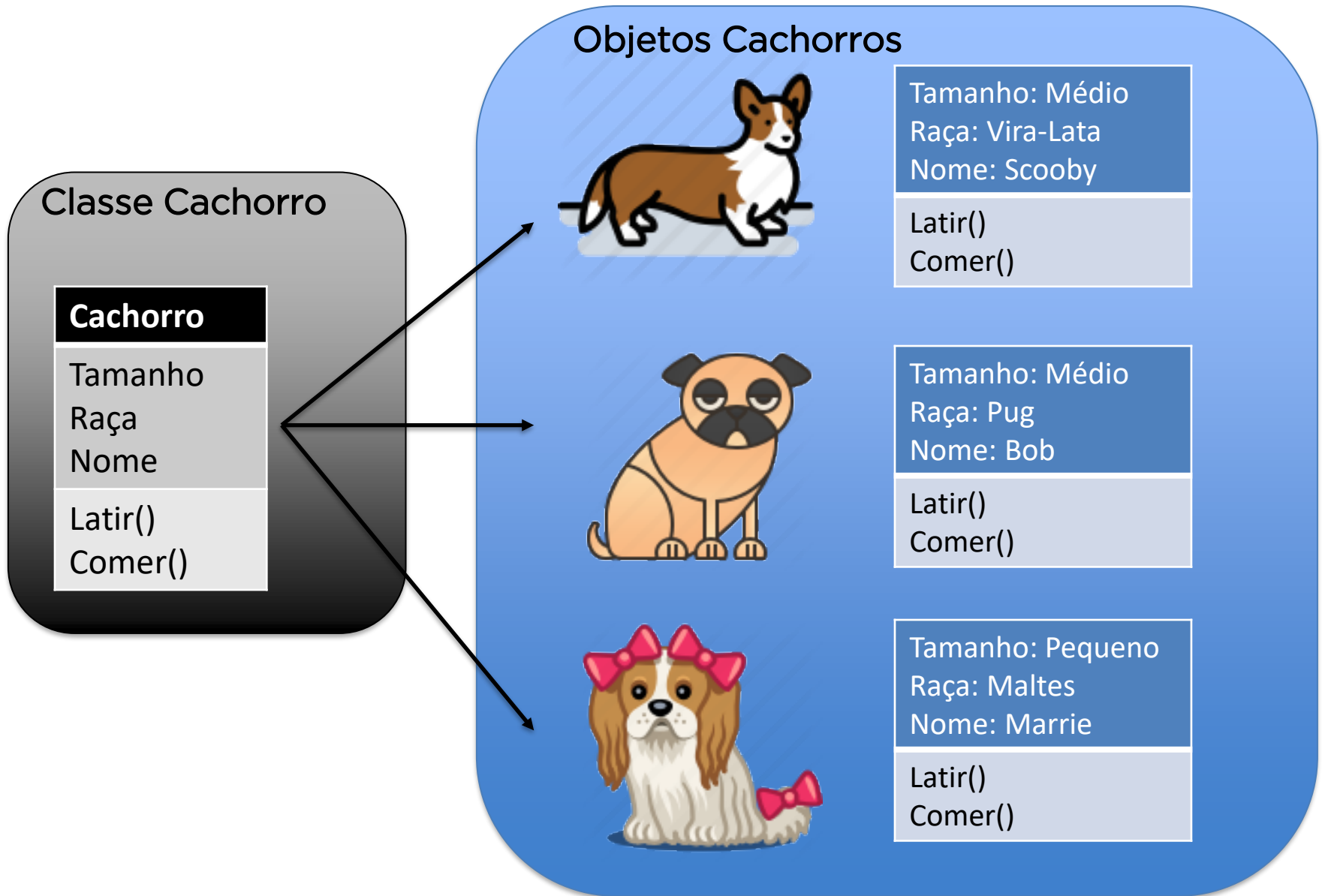
- As **abstrações** são representadas pelas **classes**;
- Uma classe deve **conter apenas os elementos necessários** para resolver um aspecto bem definido do sistema;
- A **classe é uma descrição nomeada** para um grupo de entidades (chamadas de **objetos** ou instâncias de classe) **que têm as mesmas características**;

Cachorro
Tamanho
Raça
Nome
Latir()

Uma classe cachorro, vários objetos

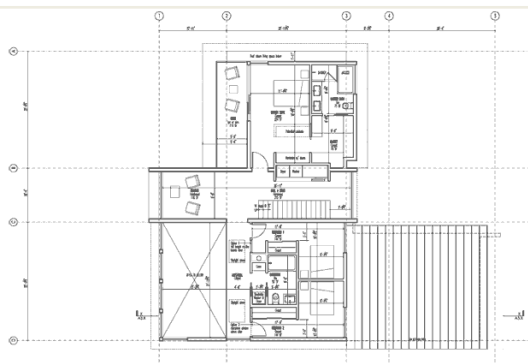


- Estas características são os **atributos** (propriedades, campos de dados) e as **operações** (comportamentos, métodos, funções) que podem ser executadas nestes objetos;
- Em outros termos, uma classe descreve os **serviços providos por seus objetos** e **quais informações** eles podem armazenar;
- Na programação orientada a objetos a classe é a **unidade básica de programação**;
- Todos os **programas são escritos como um conjunto de classes**, e todos os códigos que você escrever devem fazer parte de uma classe;



O QUE É UMA CLASSE?

- Uma classe é a descrição de um conjunto de objetos que compartilham os mesmos *atributos, operações, relações, e semânticas*;
 - *Um objeto é uma instância de uma classe*;
- Uma classe é uma abstração, uma vez que:
 - Enfatiza características relevantes;
 - Suprime outras características;



Classe



Instanciação



Objeto

Classe Curso

Propriedades

Nome

Local

Dias oferecidos

Carga horária

Hora de Início

Hora de Término



Comportamentos

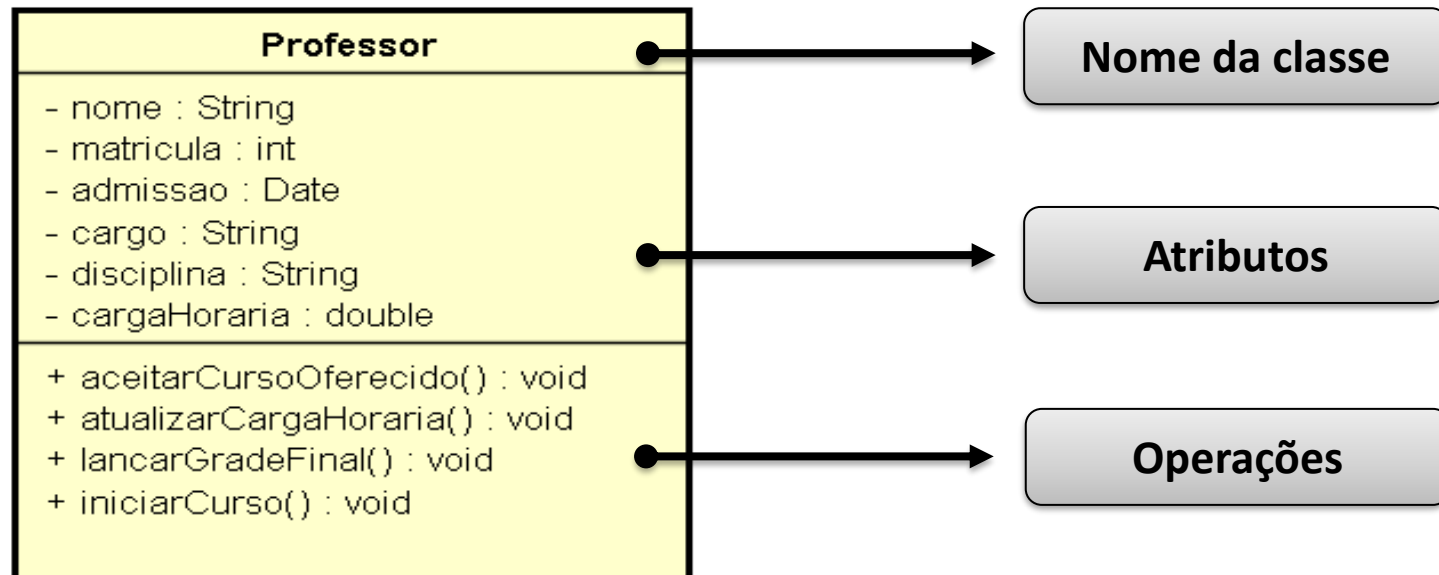
Adicionar um aluno

Excluir um aluno

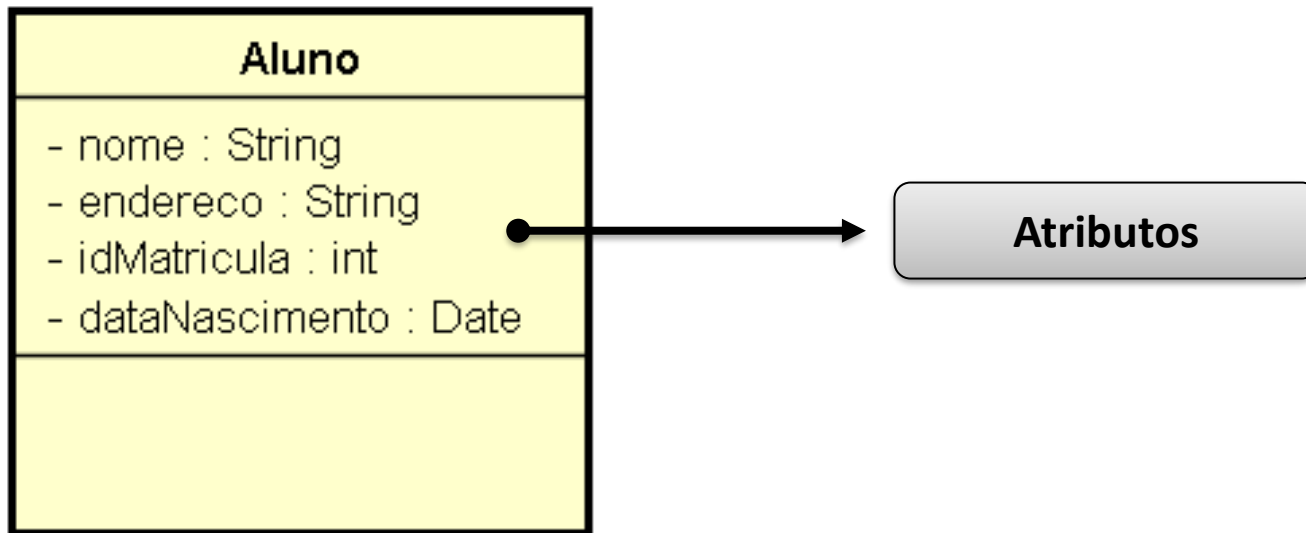
Obter lista de alunos

Verificar se está cheio

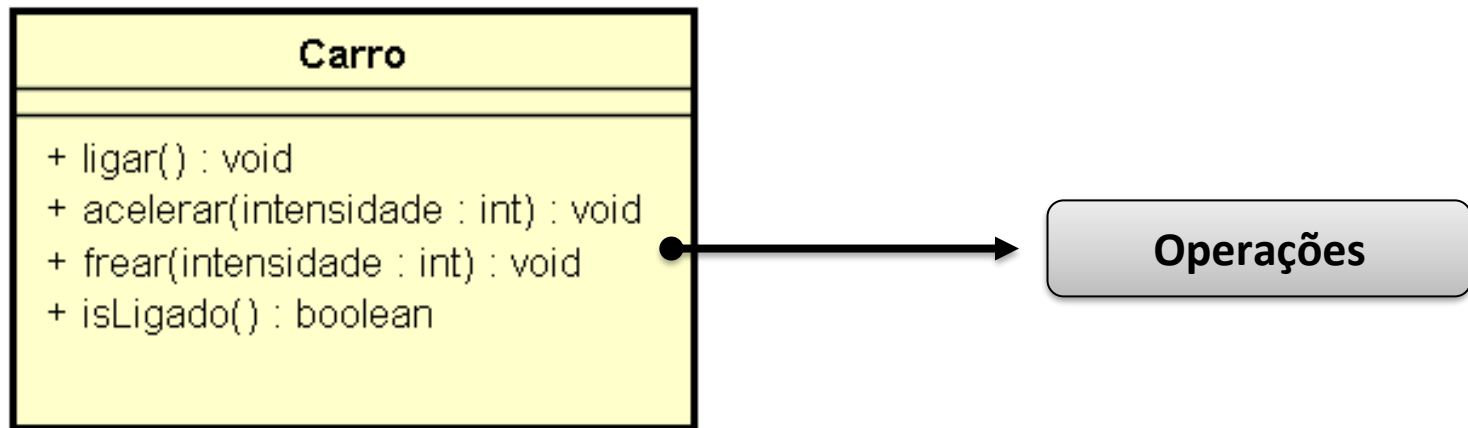
- É possível representar **graficamente** uma classe através de um **diagrama de classes** (UML), este diagrama é uma representação da **estrutura e relações das classes** que servem de modelo para objetos;
- Uma classe é representada através de um **retângulo com três compartimentos**;



- Um **atributo** é o **nome** que se dá à **propriedade** de uma classe;
- O **atributo** **descreve** o **tipo** de **valores** que a propriedade possui;
 - Um classe pode ter **qualquer número de atributos** ou nenhum atributo;

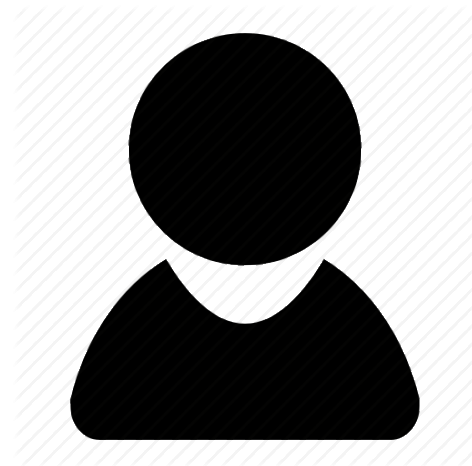


- **Operação é um serviço** que pode ser solicitado a partir de um objeto para **executar de comportamento**. Uma operação tem uma assinatura, que pode restringir os parâmetros reais que são possíveis;
- Um **classe** pode ter qualquer número de operações ou nenhuma operação





- A classe **Pessoa** possui os seguintes atributos e operações:

Pessoa
<ul style="list-style-type: none">- nome : String- sexo : String- idade : int- casa : Casa- carro : Carro
<ul style="list-style-type: none">+ exibirDadosPessoais() : void+ exibirPatrimonio() : void



- A classe **Pessoa** pode gerar vários objetos:


Objeto #1

nome = "Pedro"
idade = 52
sexo = "Masculino"
casa = 
carro = 

exibirDadosPessoais()
exibirPatrimonio()



Objeto #2

nome = "Telma"
idade = 25
sexo = "Feminino"
casa =
carro = 

exibirDadosPessoais()
exibirPatrimonio()



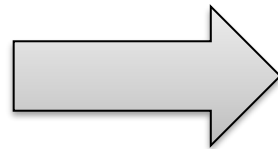
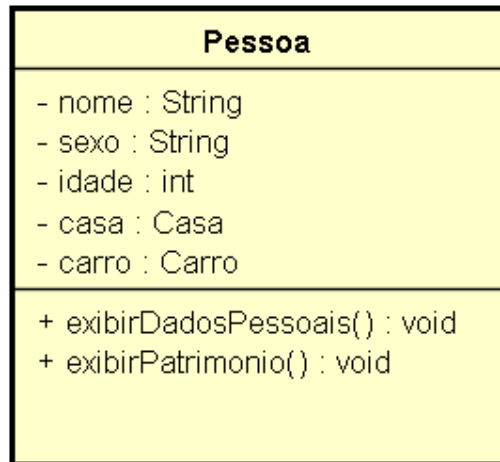
Objeto #3

nome = "Julio"
idade = 20
sexo = "Masculino"
casa =
carro =

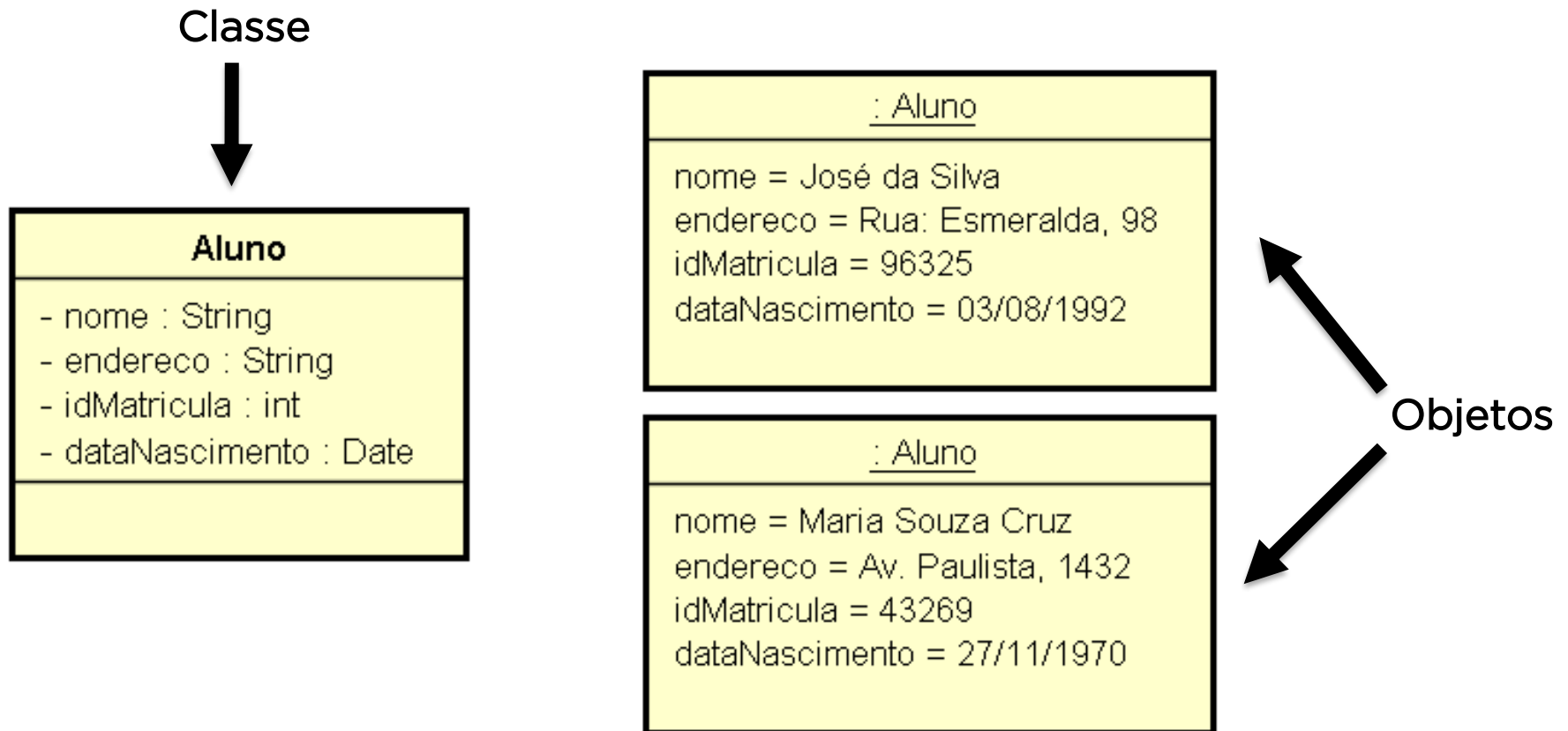
exibirDadosPessoais()
exibirPatrimonio()



- Uma classe é uma definição abstrata de um objeto;
 - Ela define a **estrutura** e **comportamento** de cada objeto da classe;
 - Ela serve como um **modelo** para a **criação de objetos**;
- Classes **não** são coleções de objetos;



- Atributos em classes e objetos:

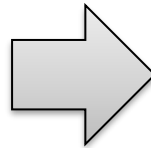
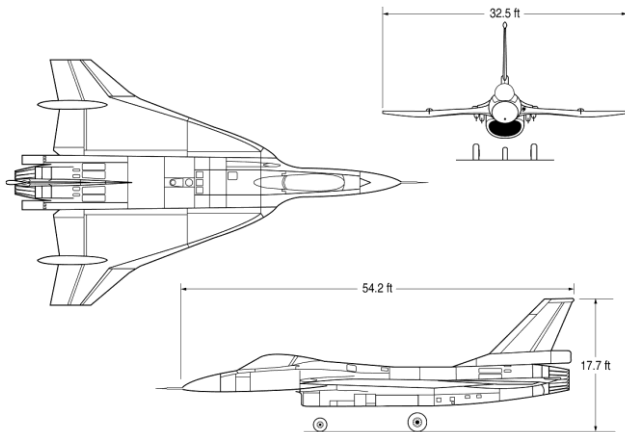


- Um **conjunto de princípios** (abstração, encapsulamento, herança e polimorfismo) **guiando a construção do software**, em conjunto com linguagens, bancos de dados e outras ferramentas que suportam esses princípios. (*Object Technology - A Manager's Guide, Taylor, 1997.*)
- **Vantagens** da orientação a objetos:
 - Facilidades arquiteturais e reuso de código;
 - Reflete em modelos de mundo real;
 - Incentiva a estabilidade;
 - É adaptável à mudanças;



O QUE É UM MODELO VISUAL?

- Um modelo é a simplificação da realidade;



- É possível construir um prédio sem a maquete, as plantas, a estruturação total de elétrica, gás e hidráulica?

Sim é possível(heheh), mas **NÃO** faça isto!

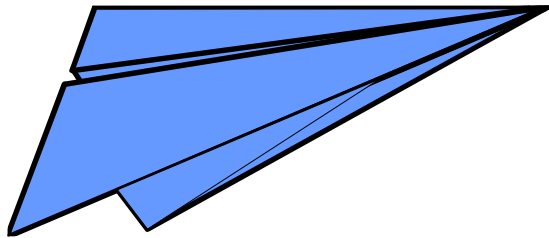




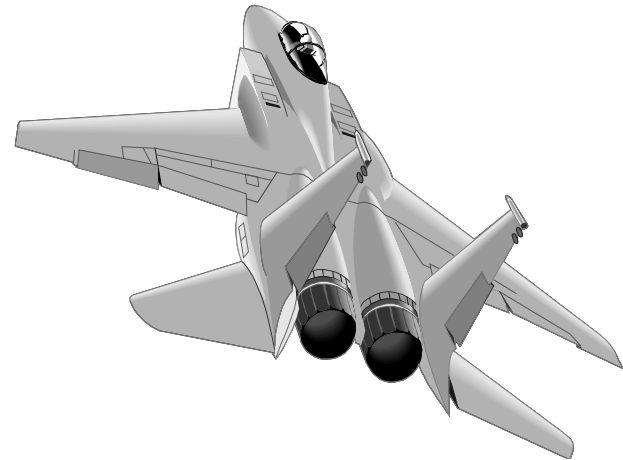
- **Modelagem atinge quatro objetivos:**
 - Ajuda a visualizar um sistema como deseja que ele seja;
 - Permite especificar a estrutura ou o comportamento de um sistema;
 - Disponibiliza um modelo que orienta na construção de um sistema;
 - Documenta as decisões realizadas;
- Os **modelos de sistemas** são construídos porque não é possível **compreender o sistema em sua totalidade**;
- Os **modelos** são construídos para **melhor entendimento** do sistema que está sendo desenvolvido;

Menos Importante

Mais Importante



Avião de papel



Avião à Jato

PRÁTICA 1



- **Represente graficamente** as classes de forma que elas que abstraíam:
 - Um candidato - no contexto de uma agência de empregos;
 - Um médico - no contexto de um hospital;
 - Um piloto - no contexto de uma corrida de Fórmula 1;
- Apresente de forma gráfica uma **instância (objeto)** de cada classe definida anteriormente;

PRÁTICA 2

Sistema de e-Commerce para venda de livros digitais (e-book)

Com o intuito de fazer um estudo prévio você ficou responsável por identificar as **classes relacionadas** aos seguintes processos:



- O cliente encontra um e-book e então o coloca num carrinho de compras;
- Conforme o cliente informa outros e-books, estes devem ser adicionados ao carrinho de compras;
- Ao finalizar a compra o cliente deve informar como quer obter seu e-book, as opções são: download ou e-mail;
- Se o cliente não possuir cadastro, o mesmo deve ser realizado na finalização da compra;
- O cliente pode ser uma pessoa física ou jurídica, caso seja uma pessoa jurídica será necessário cadastrar também o nome e telefone de uma pessoa para contato;

PRÁTICA 2

Com base no case anterior:

- Realizar a abstração para encontrar as classes;
- Montar as classes com os atributos, definido os seus tipos:
 - String - para texto
 - int - para número inteiro
 - double - para número real
 - Date - para data
 - boolean - para verdadeiro/falso
- Montar as classes com operações;



Copyright © 2020 - 2023 Prof. Thiago T. I. Yamamoto

Todos direitos reservados. Reprodução ou divulgação total ou parcial deste documento é expressamente proibido sem o consentimento formal, por escrito, do Professor (autor).

“Se você traçar metas absurdamente altas e falhar, seu fracasso será muito melhor que o sucesso de todos” – James Cameron