

Algoritmos e Programação II

Aula 02

Prof. Jânio E. Teixeira Júnior



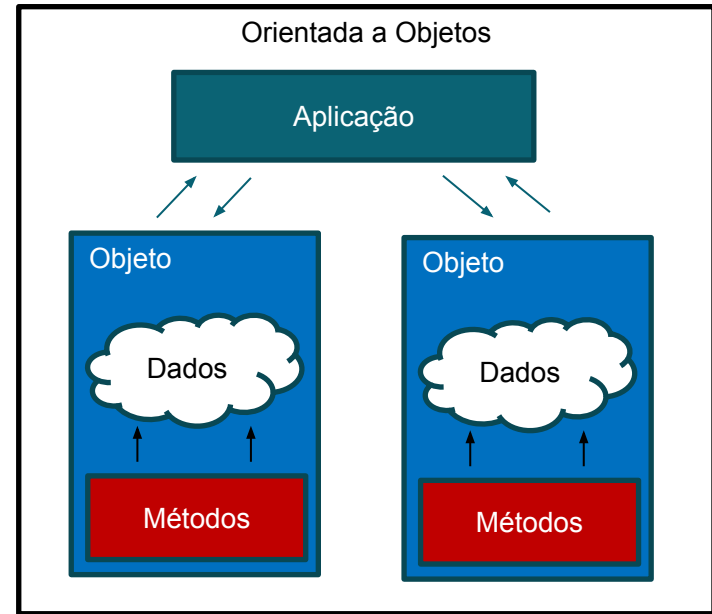
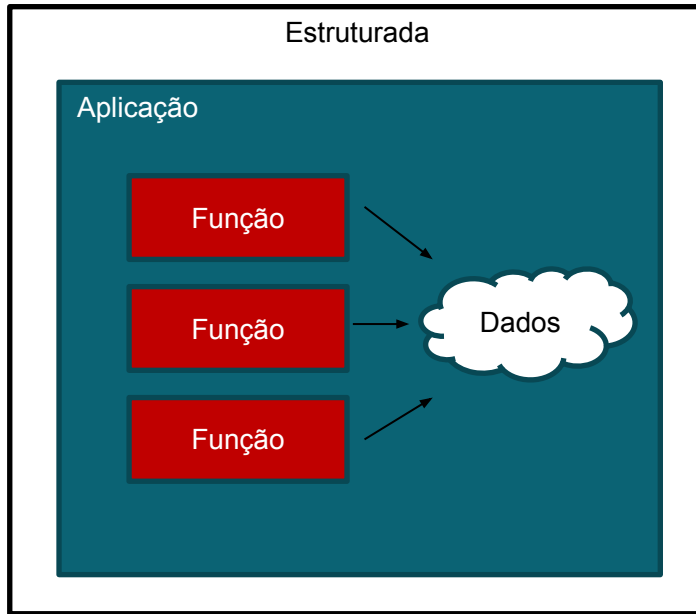
Roteiro

- Paradigmas de Programação
- Abstração de dados
 - Classes
 - Atributos
 - Métodos
- UML – Linguagem de Modelagem Unificada

- [illegible]



Paradigmas de Programação



Quatro Pilares da Orientação a Objetos



1. **Abstração**
2. Encapsulamento
3. Herança
4. Polimorfismo



Orientação a Objetos – Abstração de Dados

- “A abstração é considerada uma operação intelectual que consiste em isolar um elemento ou um conceito à exclusão de outros”
- “Abstrair significa selecionar aspectos específicos de um problema a ser analisado, deixando de lado outros aspectos”
- “Abstração é a representação de uma entidade do mundo real na forma de ideias”
- Abstrair = Simplificar
- Abstração gera um Modelo

Orientação a Objetos – Abstração de Dados



Abstrair os
dados de uma
pessoa.





Concreto x Abstrato



Concreto x Abstrato

Concreto

- É algo do mundo real, que tem corpo (opõe-se a abstrato).

Abstrato

- É algo que existe somente como conceito ou ideia.

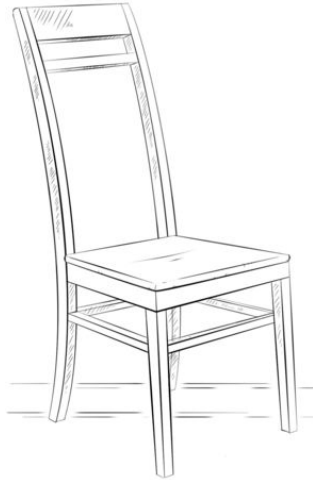
Arte Abstrata

- O pintor russo **Wassili Kandinsky** (1866-1944) é considerado o pioneiro da arte abstrata.



Orientação a Objetos – Abstração de Dados

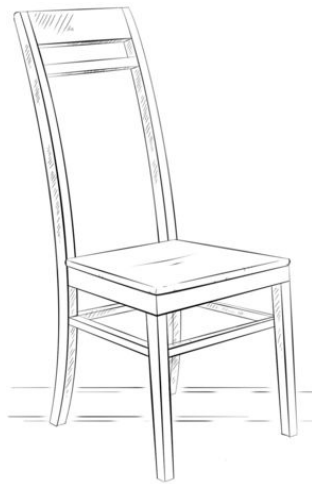
Abstrato - Classes



Concreto - Objetos

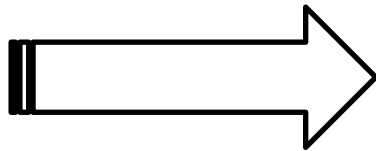


Orientação a Objetos – Abstração de Dados



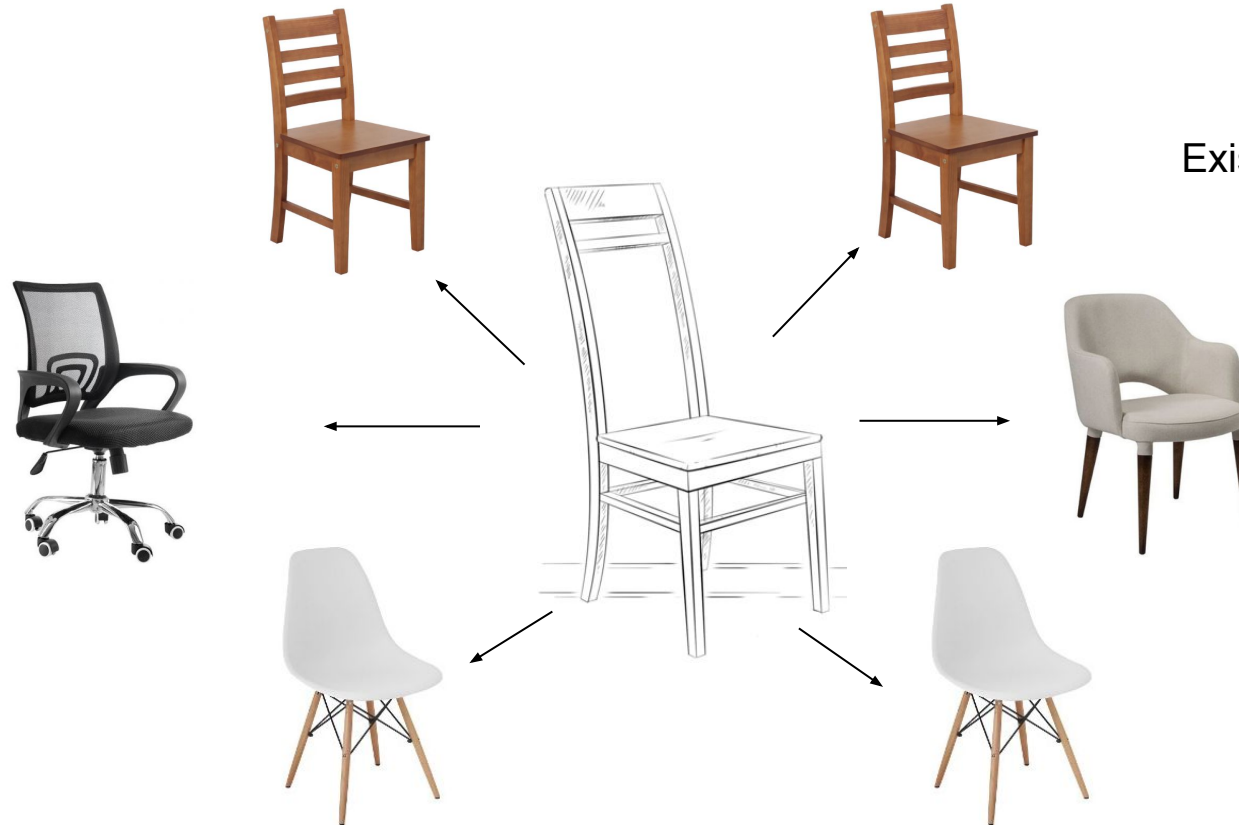
Classe (Abstrato)

Através de uma classe,
cria/ instancia um objeto



Objeto (Concreto)

Orientação a Objetos – Abstração de Dados





Classe

- Uma **Classe** pode ser definida como uma representação de uma entidade:
 - **do mundo real** (Ex. Cadeira, Empresa, Cliente, Produto, etc);
 - **que faz parte do domínio da solução** (Ex. Configuracao, Log, Excecao, etc).
- Em uma **Classe** pode existir:
 - **atributos**, que são as **características** (nome, qtd, valor);
 - **métodos**, que são as **ações** (irPara, novaEmpresa, buscar).



Linguagem UML

- UML - *Unified Modeling Language*;
- 1997 – versão 1.0
 - Oracle, HP e Rational.
- *Objetivo*
 - Auxiliar a visualizar, especificar, construir e documentar a estrutura e o comportamento de um sistema.



Classe - UML

- Classe: Cadeira
- Atributos:
 - nome, qtdPes, cor, preco.
- Métodos:
 - possuiEstoque;
 - obterPreco;
 - obterCor.

Cadeira
+ nome: String + qtdPes: Integer + cor: String + preco: Double
+ possuiEstoque(): Boolean + obterPreco(): Double + obterCor(): String



Exercício

- Abstrair dados das seguintes entidades:
 - Porta;
 - Celular;
 - Máquina de Lavar;
 - Geladeira;
 - Sala Cinema;
 - Quarto de Hotel.

Trabalho - Corrida

Construir um programa que realize uma corrida (km de arrancada) entre diferentes tipos de carros e motos. Ganha o veículo que atingir a maior velocidade.

1. Faça a abstração dos dados que identifique cada tipo de veículo;
 - a) Classe e Atributos.
2. A corrida deve possuir no mínimo 4 veículos (de cada tipo).
3. A corrida deve acontecer em 20 segundos:
 - a) A cada segundo os veículos aumentam a aceleração;
 - i. Exceto para os segundos 6, 11 e 15 (os veículos realizam uma frenagem).
4. A corrida dos carros acontece antes da corrida das motos.

Cálculo para Acelerar um veículo:

velocidade = velocidade + 5.

velocidade(km/h) += velocidade * (15% da cilindrada) – (1 para cada 300 kg do veículo).

Cálculo para Frear um veículo:

Toda vez que frear o veículo perde 25% da velocidade.

No final deve-se imprimir os dados de todos os veículos.

Algoritmos e Programação II

Aula 02

Obrigado.

