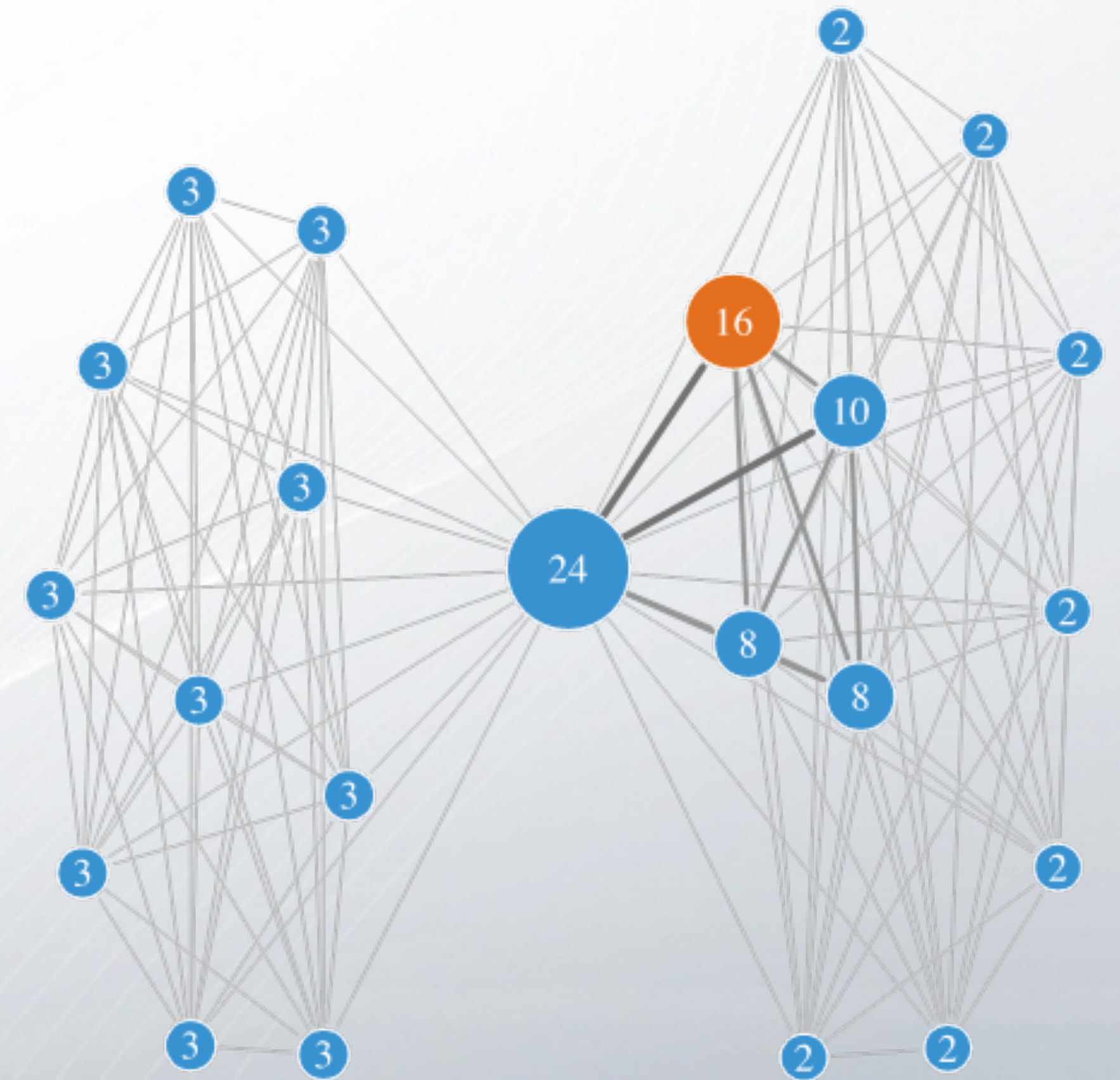


Linguagem de Programação Estatística

Orientação a Objetos em R



Robson Fernandes

Acadêmico

Mestrando em Matemática, Estatística e Computação Aplicadas (Data Science & Machine Learning) - USP
Especialização em Reconhecimento de Padrões e Análise de Imagens - UNICAMP
Pós-Graduado em Arquitetura de Software Distribuído - PUC-MG
MBA em Engenharia de Software Orientada a Serviços – SOA – METROCAMP
Certificado – JavaScript e HTML5 Developer – W3C INTERNACIONAL
Autor do Livro Gestão da Tecnologia da Informação: Teoria e Prática

Profissional

Cientista de Dados Sênior – Finch Soluções
Docente Pós-Graduação - MBA em Data Science & Machine Learning - UNIP
Docente Pós-Graduação em Engenharia de Software - USC
Docente Graduação em Ciência da Computação - UNIP

Site

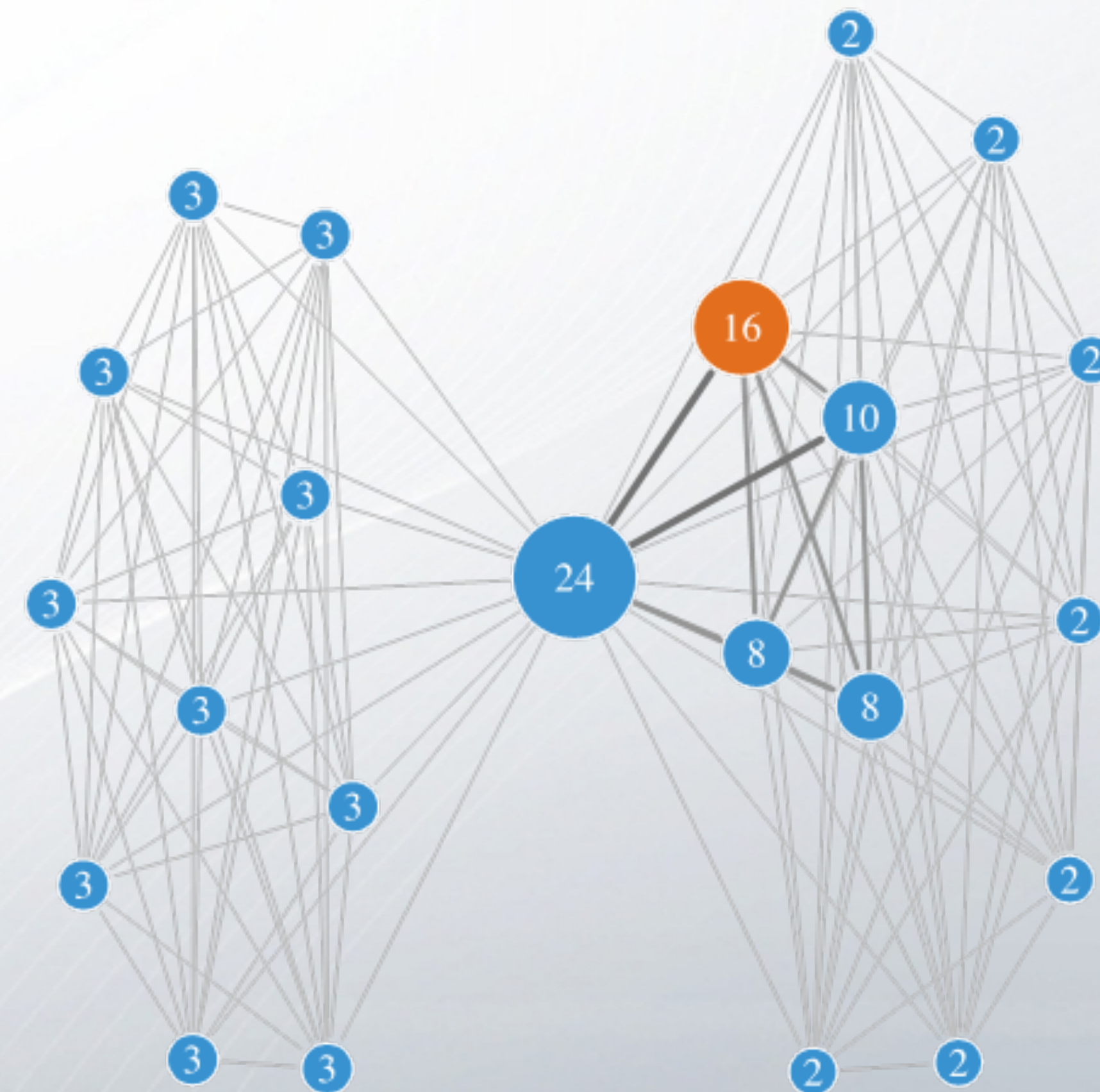
<http://robsonfernandes.net>

e-mail

robson.fernandes@usp.br /
robs.fernandes@outlook.com

Agenda

- Introdução a Orientação a Objetos
- Conceitos básicos de Orientação a Objetos
- Classes
- Objetos
- Atributos
- Métodos
- Herança
- Encapsulamento
- Polimorfismo
- Coesão e Acoplamento



Introdução a Orientação a Objetos

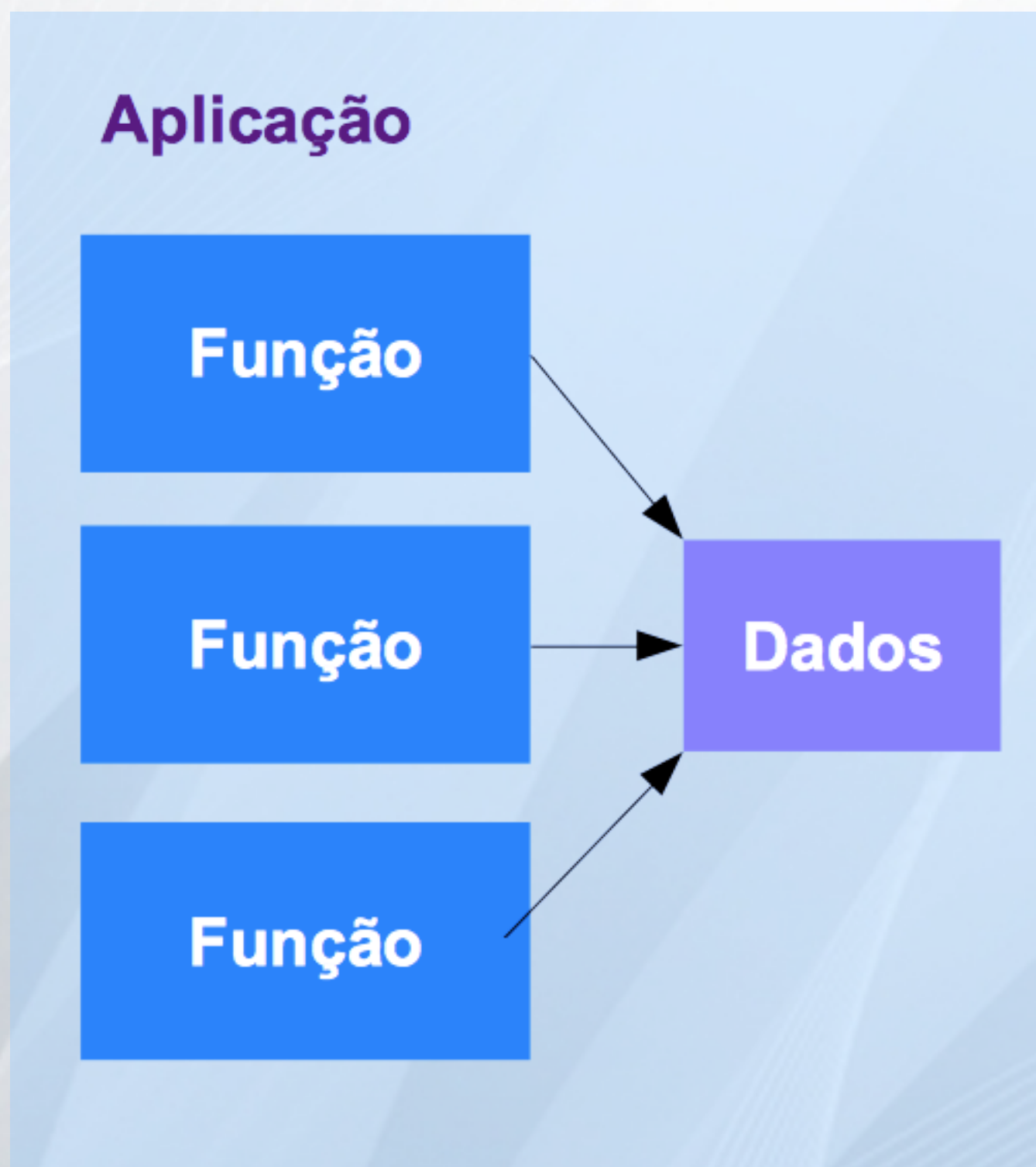
Modelos de desenvolvimento de software

- **Estruturado**
- **Orientado a Objetos**



Desenvolvimento de Software Tradicional

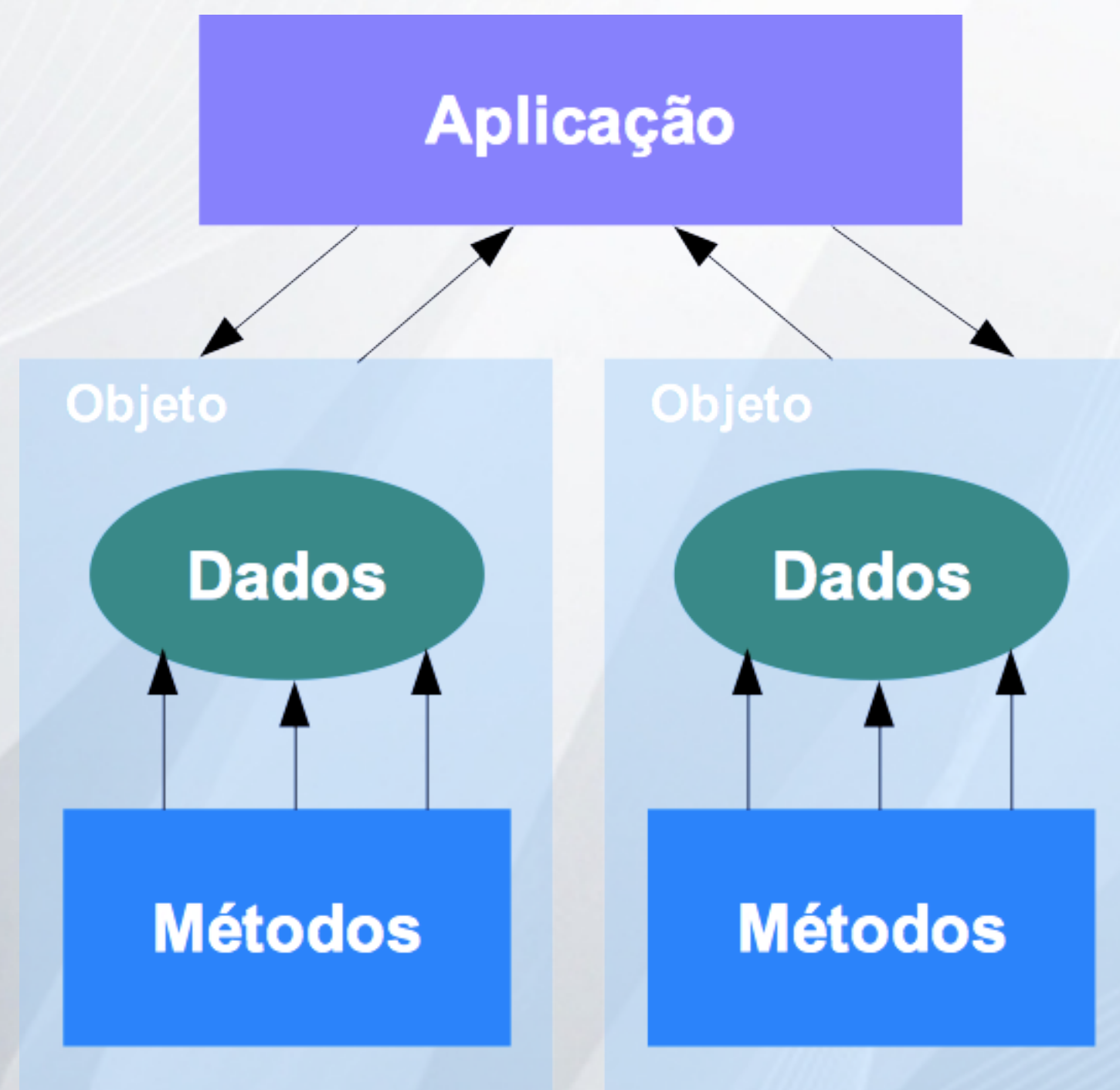
- **Estruturado**



- Visão Tradicional usa perspectiva de Algoritmo.
- O principal bloco de construção é o procedimento ou função.
- Conduz o foco de atenção para questões referentes a decomposição de algoritmos maiores em outros menores.
- Modelagem de dados divide as informações em tabelas, criando mecanismos para junção posterior.

Desenvolvimento de Software Tradicional

- **Orientação a Objetos**



- Objetos do mundo real transformam-se em objetos de software.
- Programação orientada a objetos é frequentemente referenciada como um novo paradigma de programação.
- Viabiliza a montagem de sistemas a partir de componentes.

Outros paradigmas de programação

- **Orientação a Objetos**

Lógico	Imperativo/Procedural	Funcional/Descritiva
Prolog – QLISP	Pascal – Fortran - Basic	ML – F# - Haskell
E muito mais... Orientada a Teste, Escalar, Restritiva, Genérica....		

Conceitos: Orientação a Objetos - Classe

CLASSE

Uma classe é uma estrutura que abstrai um conjunto de objetos com características similares. É como um projeto, a partir dela que criamos os objetos utilizados em nossos programas.

Conceitos: Orientação a Objetos - Classe

- **Classes**
- Classes são os elementos mais importantes de qualquer sistema orientado a objetos
- Uma classe é uma descrição de um conjunto de objetos com os mesmos **atributos, operações, relacionamentos e semântica**.
- Classes são usadas para capturar o **vocabulário** de um sistema
- Classes são abstrações de elementos do domínio do problema, como “Cliente”, “Banco”, “Conta”



Conceitos: Orientação a Objetos - Atributos

ATRIBUTOS

Os Atributos em Programação Orientada a Objetos são os elementos que definem a estrutura de uma classe, ou seja suas características.

Conceitos: Orientação a Objetos - Atributos

- **Atributos**
- Um atributo representa alguma propriedade do que está sendo modelado, que é compartilhada por todos os objetos da classe
- Os atributos descrevem os dados contidos nas instâncias de uma classe
- Em um momento dado, um objeto de uma classe conterá valores para todos os atributos descritos na sua classe



Conceitos: Orientação a Objetos - Método

MÉTODO

Um método é uma subrotina que é executada por um objeto ao receber uma mensagem. Os métodos determinam o comportamento dos objetos de uma classe.

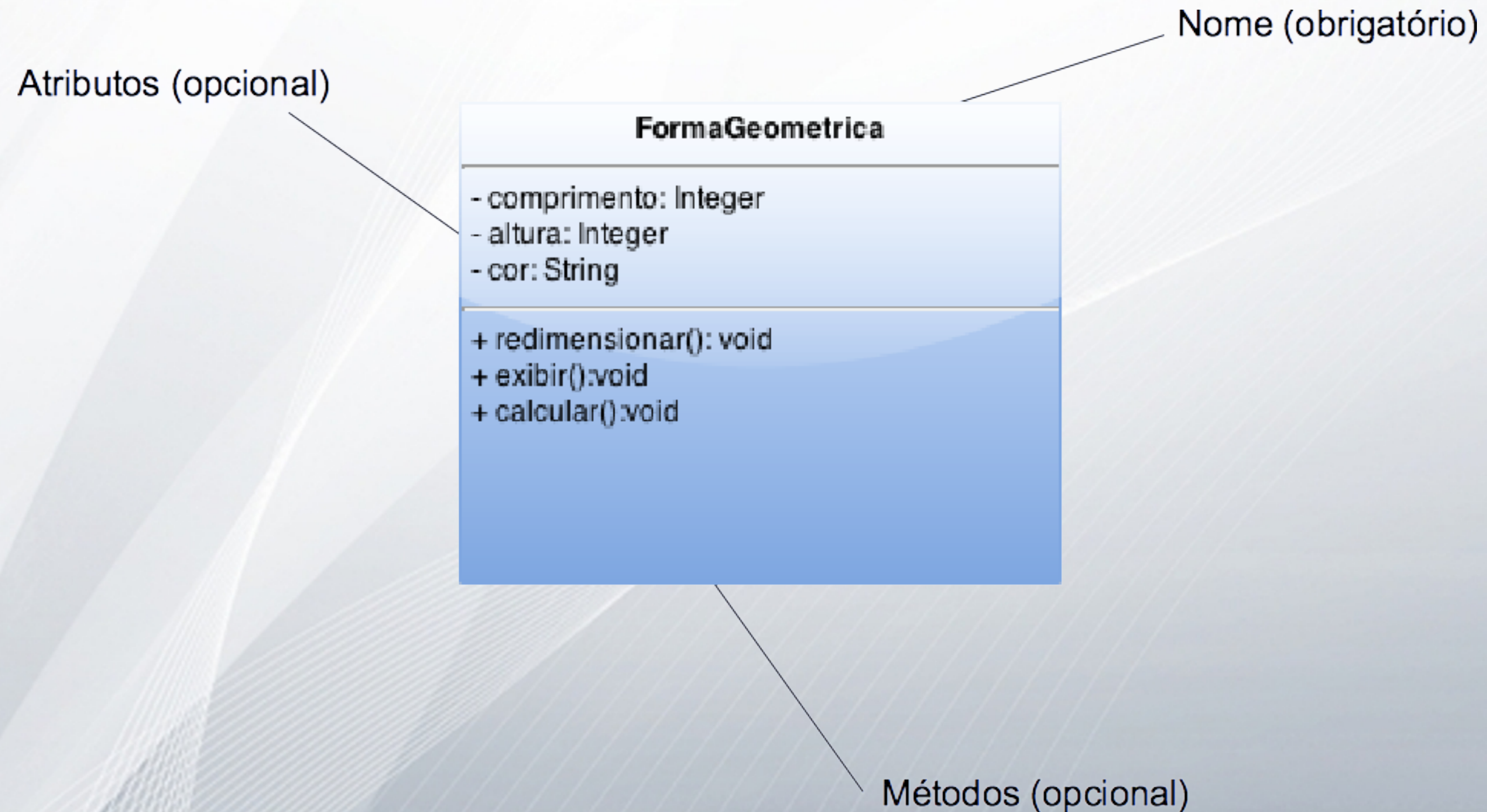
Conceitos: Orientação a Objetos - Método

- **Método**
- Um método é uma abstração de alguma coisa que se pode fazer com um objeto e que é compartilhada por todos os objetos da classe.
- Um classe pode ter qualquer quantidade de métodos.
- Métodos são o meio de alterar os valores dos atributos.



Conceitos: Orientação a Objetos - Classe

- **Notação Básica - Classe**



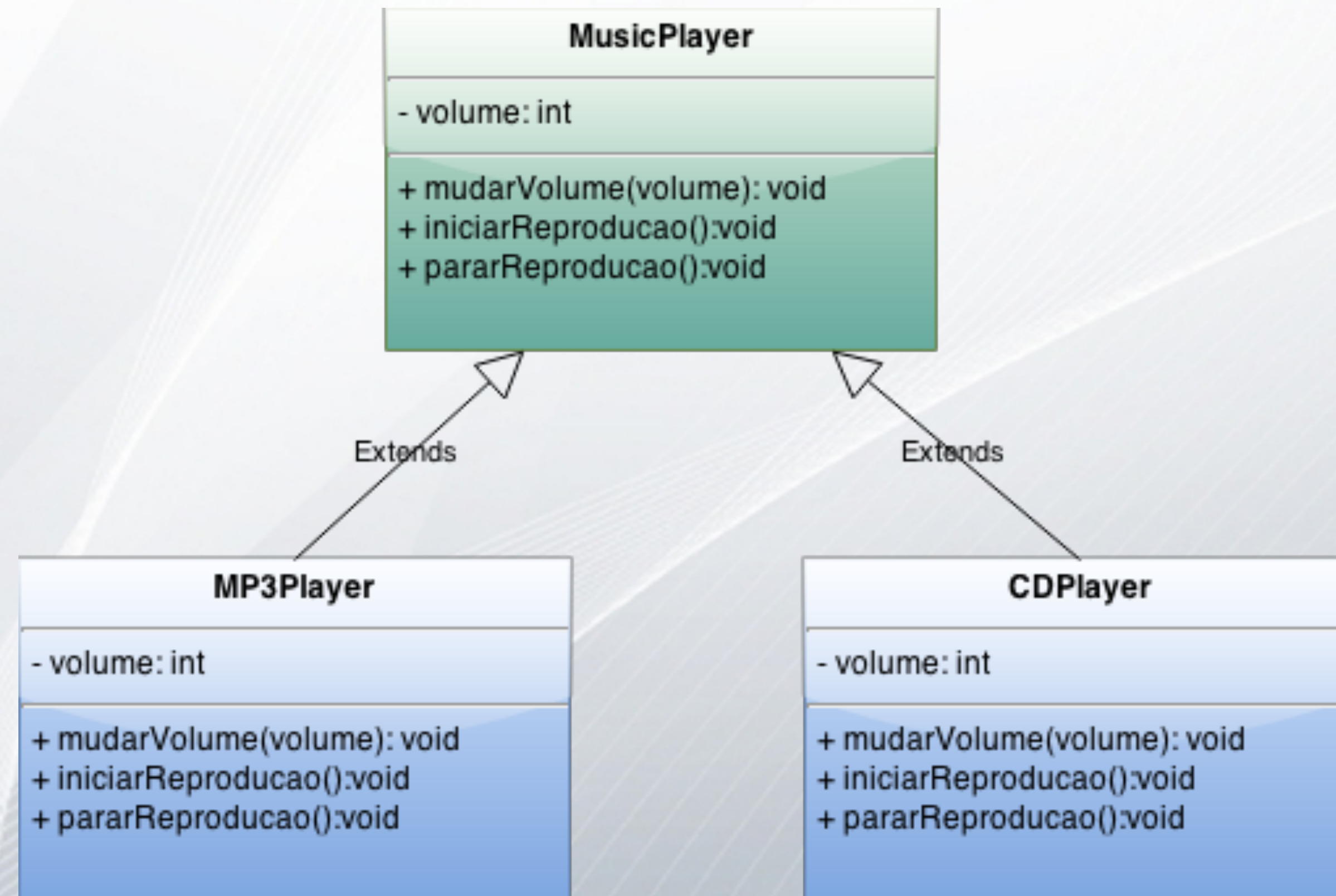
Conceitos: Orientação a Objetos - Herança

HERANÇA

Herança é um mecanismo que permite que características comuns a diversas classes sejam fatoradas em uma classe base, ou superclasse. A partir de uma classe base, outras classes podem ser especificadas.

Conceitos: Orientação a Objetos - Herança

- Uma classe pode ter várias filhas, mas pode ter apenas uma mãe, é a chamada herança simples.



Conceitos: Orientação a Objetos - Encapsulamento

ENCAPSULAMENTO

Encapsulamento é o processo de separar a interface de um objeto dos detalhes de seu funcionamento interno.

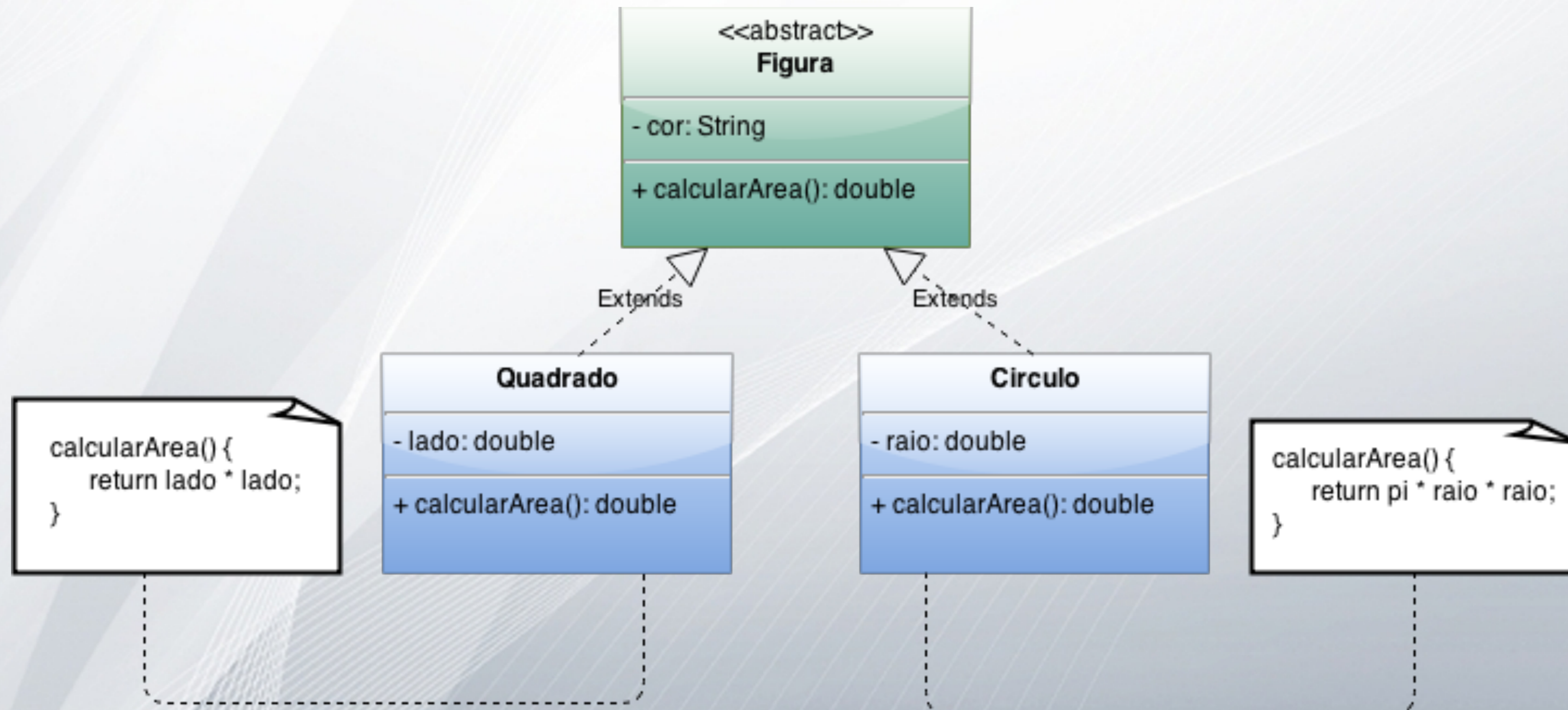
Conceitos: Orientação a Objetos - Polimorfismo

POLIMORFISMO

O polimorfismo permite que referências de tipos de classes mais abstratas representem o comportamento das classes concretas que referenciam. Assim, um mesmo método pode apresentar várias formas, de acordo com o seu contexto.

Conceitos: Orientação a Objetos - Polimorfismo

- Com o Polimorfismo, os mesmos métodos podem ser utilizados em objetos distintos, porém, com implementações lógicas diferentes.



Conceitos: Orientação a Objetos - Coesão / Acoplamento

COESÃO / ACOPLAMENTO

Acoplamento é o grau que uma classe depende, interage com outra classe. Coesão refere a finalidade que a classe foi criada.

Quanto menos uma classe depende da outra, menor o acoplamento, e maior a especialização dela, isso significa alta coesão.

Bibliografia

- LARMAN. Graig. **Utilizando UML e Padrões: uma introdução à análise e projetos orientados a objetos.** Bookman, 2004.
- SILVA. L. A.; PERES. S. M; BOSCARIOLI C. **Introdução à Mineração de Dados.** Elsevier. 2016
- FACELI, Katti; Lorena, Ana Carolina; Gama, João ; de Carvalho, A. C. P. L. F. (2011). **Inteligência Artificial - Uma Abordagem de Aprendizado de Máquina.** 1. ed. Rio de Janeiro: LTC.
- PROVOST, F.; Fawcett, T. **Data Science for Business: What you need to know about data mining and data-analytic thinking** by O'Reilly Media, 2013.
- FLACH, P. (2012). **Machine Learning: The Art and Science of Algorithms that Make Sense of Data.** Cambridge University Press.
- ALPAYDIN, E. (2004). **Introduction to Machine Learning.** MIT Press.