Aonde você quer chegar?
Vai com a

Disciplina: Análise e Projeto OO

Prof. Maurício P. de Freitas MSc.

Aula 02 – 29/02/2024
Conceitos de Orientação a Objetos

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

A **Orientação a Objetos (OO)** é uma técnica de programação **baseada na construção e utilização de objetos**. Um objeto combina dados e operações específicas, o que define um conjunto particular de responsabilidades. **Um sistema OO é um conjunto de objetos** que se relacionam para produzir os resultados desejados.

Junior, Peter Jandl. Java Guia do Programador - 4ª Edição: Atualizado para Java 16 (Portuguese Edition) (p. 131). Novatec Editora.

Características:

- Eficiente;
- Manutenção de código facilitada;

Classes, objetos e instanciação

Objeto:

- É a instância de uma classe, ou seja, representa uma entidade individual de uma categoria individual de objetos;
- Pertence a uma classe que define as características e comportamentos de um conjunto de objetos;

Objeto Funcionário 2 (instância)

PEDRO ALVES
Rua Brasil, 399
(43) 9999-9876

Objeto Funcionário 1 (instância)

JOSÉ DA SILVA
RUA DOS PATRIOTAS, 299
(43)9999-9876

Classes, objetos e instanciação

Classe:

- Uma classe é um modelo definido pelo programador para um novo tipo de objeto;
- Este modelo relaciona seus atributos (características e estados) e seus comportamentos (funcionalidades);
- Pode representar uma entidade real (concreta) ou conceitual (abstrata);

CLASSE FUNCIONÁRIO

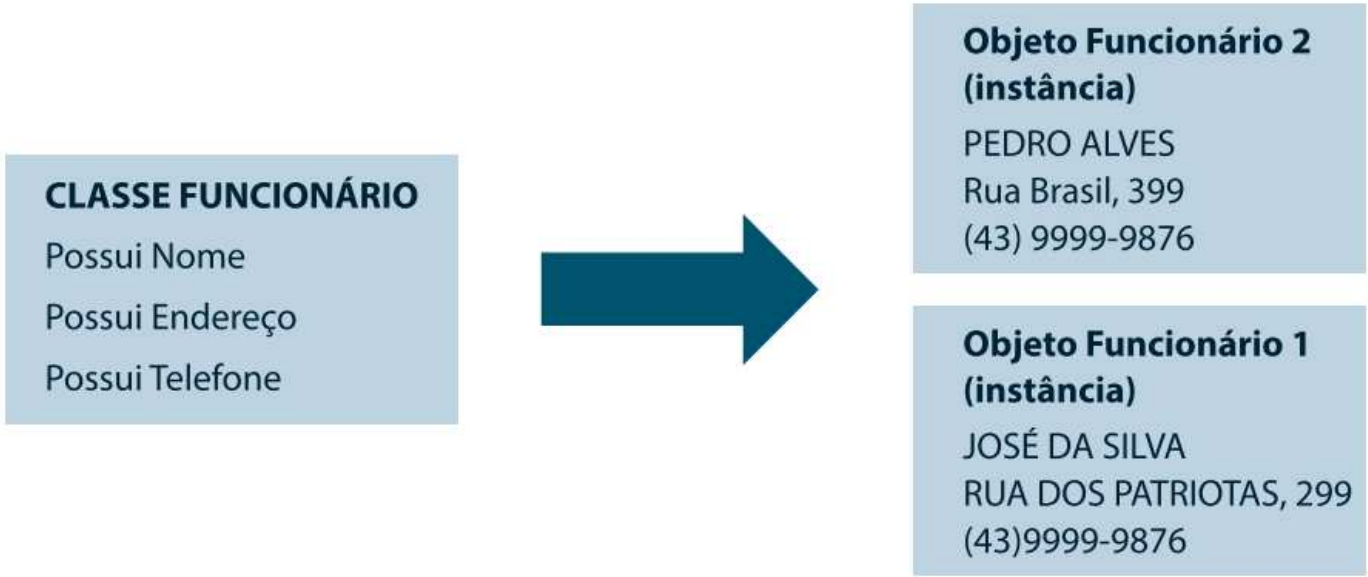
Possui Nome

Possui Endereço

Possui Telefone

Classes, objetos e instanciação

Instanciação: processo de criação de uma entidade (objeto) a partir do seu modelo (classe).

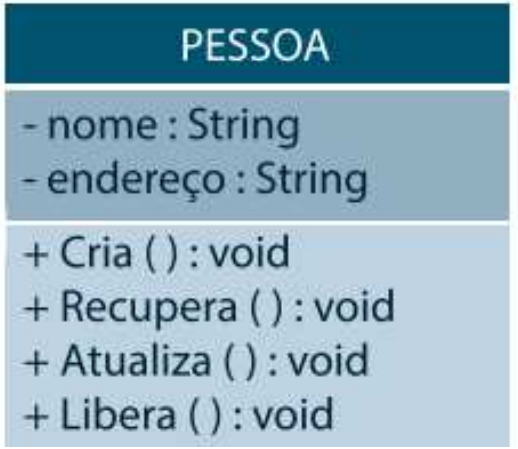


Classes, objetos e instanciação

Classe:

- **Atributos (campos, variáveis-membro)**: são variáveis destinadas a armazenar dados que caracterizam o objeto, seu estado ou suas partes;

- **Métodos (operações, funções-membro)**: são sub-rotinas associadas aos objetos, i.e., trechos de código que permitem realizar ações ou transformações sobre os valores dos atributos desse código, modificando **seu estado** e proporcionando o comportamento desejado.



Classes: Podem representar objetos concretos ou abstratos.

Objetos concretos

Objetos abstratos

?

?

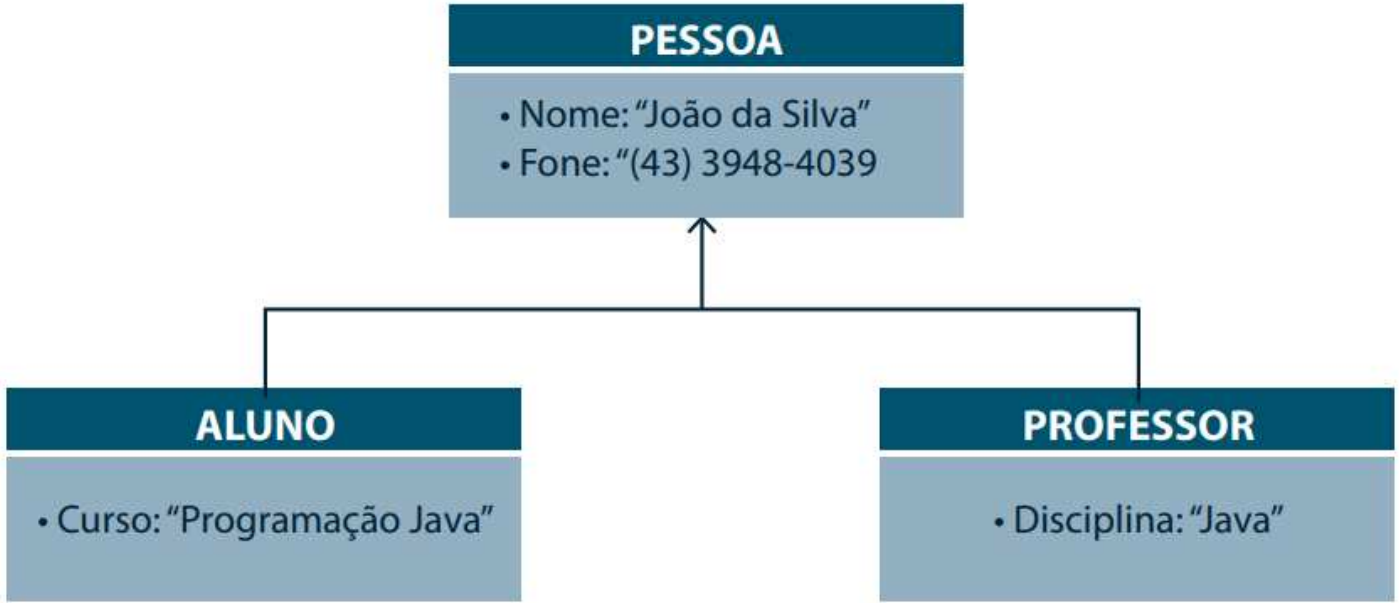
Análise e Projeto **Orientado a Objetos**

Estados:

Transição de estados:

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

Estado:


PESSOA
• Nome: "João da Silva"
• Fone: "(43) 3948-4039
ALUNO
• Curso: "Programação Java"
PROFESSOR
• Disciplina: "Java"

Associação: é a forma com que objetos se relacionam.

Tipos de Associação:

- Associação Unária;
- Associação Binária;
- Associação Ternária;
- Classe Associativa;
- Associação de Agregação;
- Herança.

Análise e Projeto **Orientado a Objetos**

Pilares da OO:

Abstração: Abstrair um objeto do mundo real em POO significa criar uma representação simplificada desse objeto em termos de uma classe.

Abstração de um Carro

Atributos Relevantes:

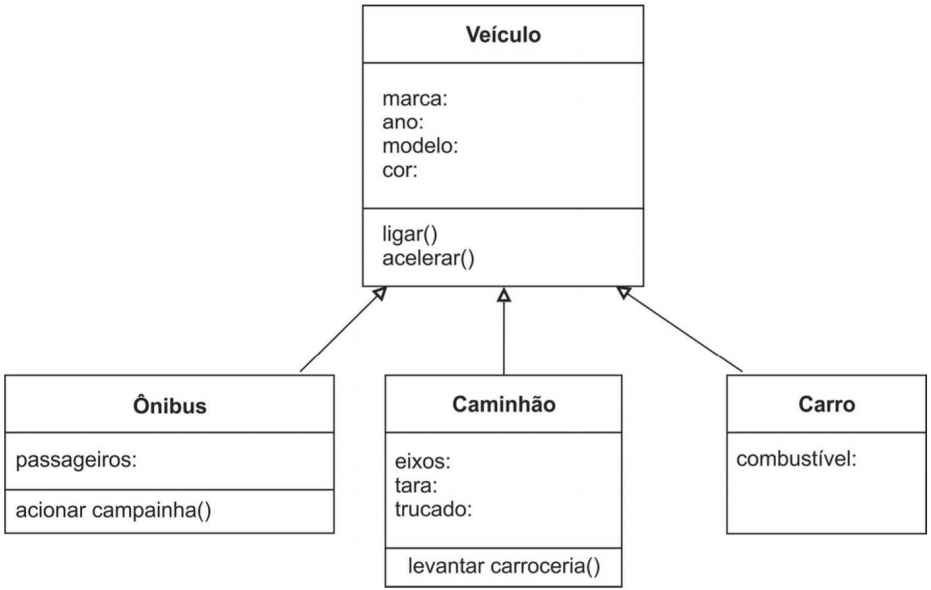
- Modelo: O modelo do carro (por exemplo, Sedan, SUV);
- Cor: A cor do carro;
- Ano de Fabricação: O ano em que o carro foi fabricado;
- Quilometragem: A quilometragem percorrida pelo carro.

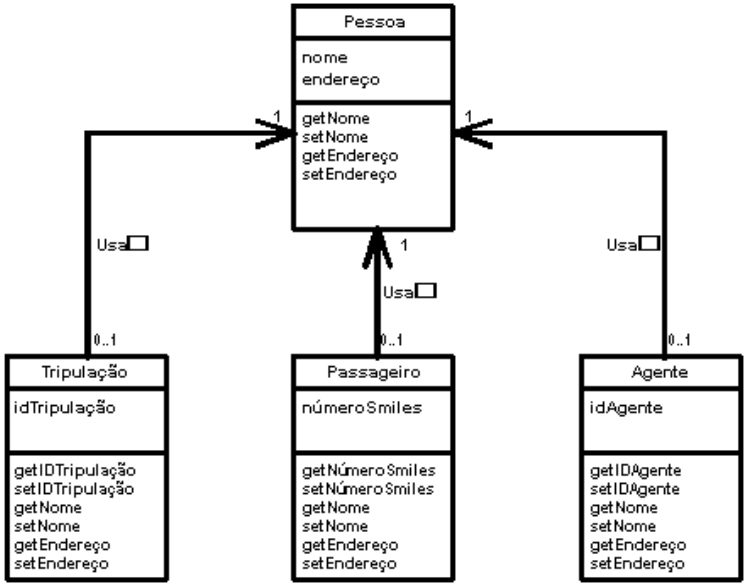
Comportamentos Relevantes:

- Acelerar: Aumentar a velocidade do carro.
- Frear: Diminuir a velocidade ou parar o carro.
- Virar: Mudar a direção do carro.

Análise e Projeto **Orientado a Objetos**

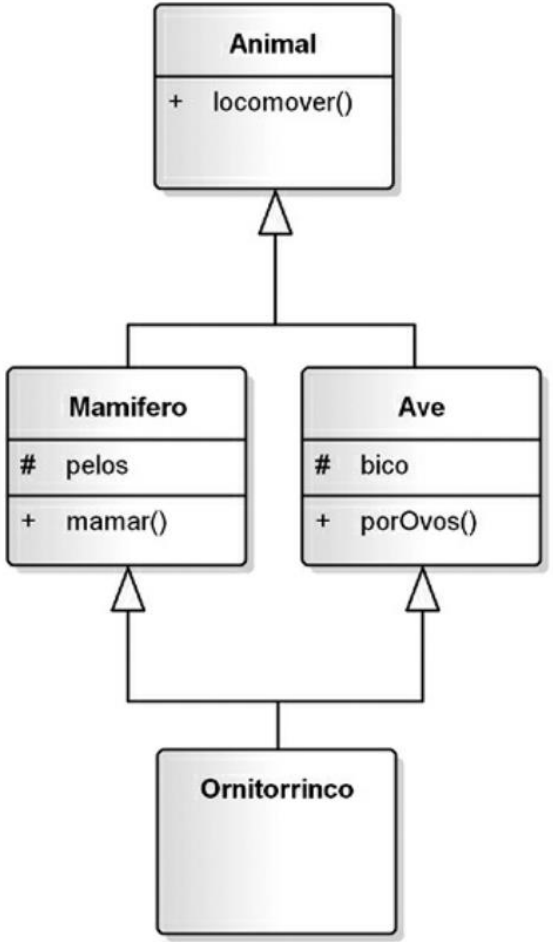
Herança: Herança permite que uma classe (classe filha ou subclasse) herde atributos e métodos de outra classe (classe pai ou superclasse).





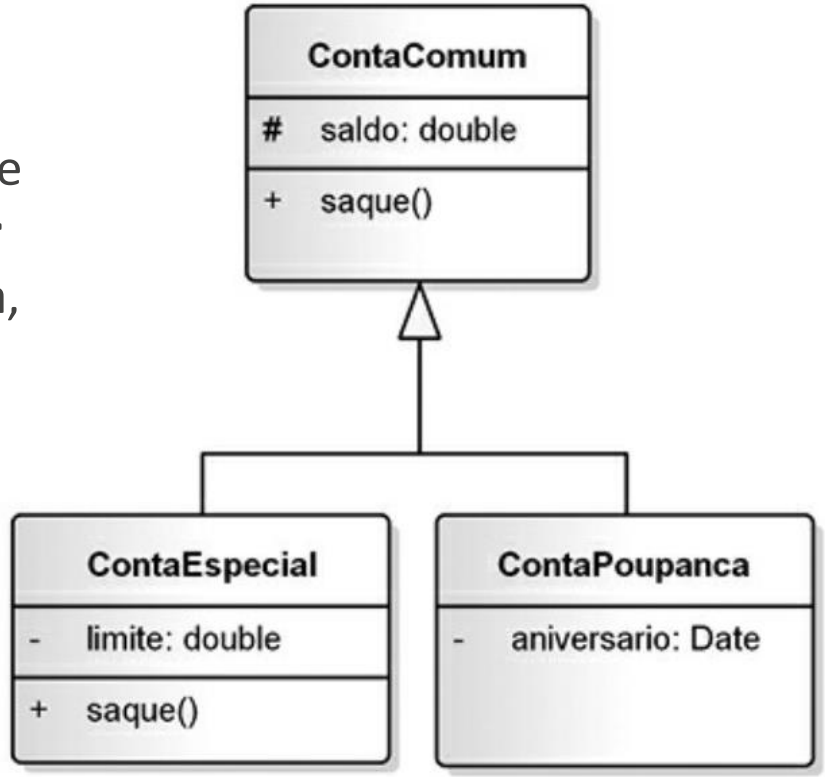
Herança múltipla: a capacidade de uma classe herdar características e comportamentos de mais de uma superclasse.

- **Desvantagem**: a herança múltipla pode complicar o design e a manutenção do código;
- **Linguagem que implementam:** C++, Pytho e Smalltalk.



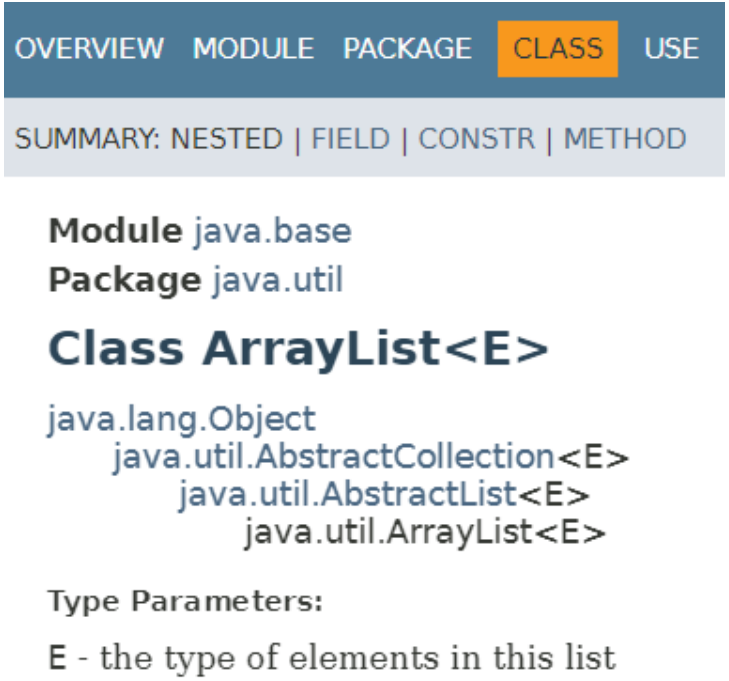
Polimorfismo:

- É o princípio em que classes derivadas de uma mesma superclasse **podem invocar operações que têm a mesma assinatura**, mas comportamentos diferentes em cada subclasse**, produzindo resultados diferentes**, dependendo de como cada objeto implementa a operação.

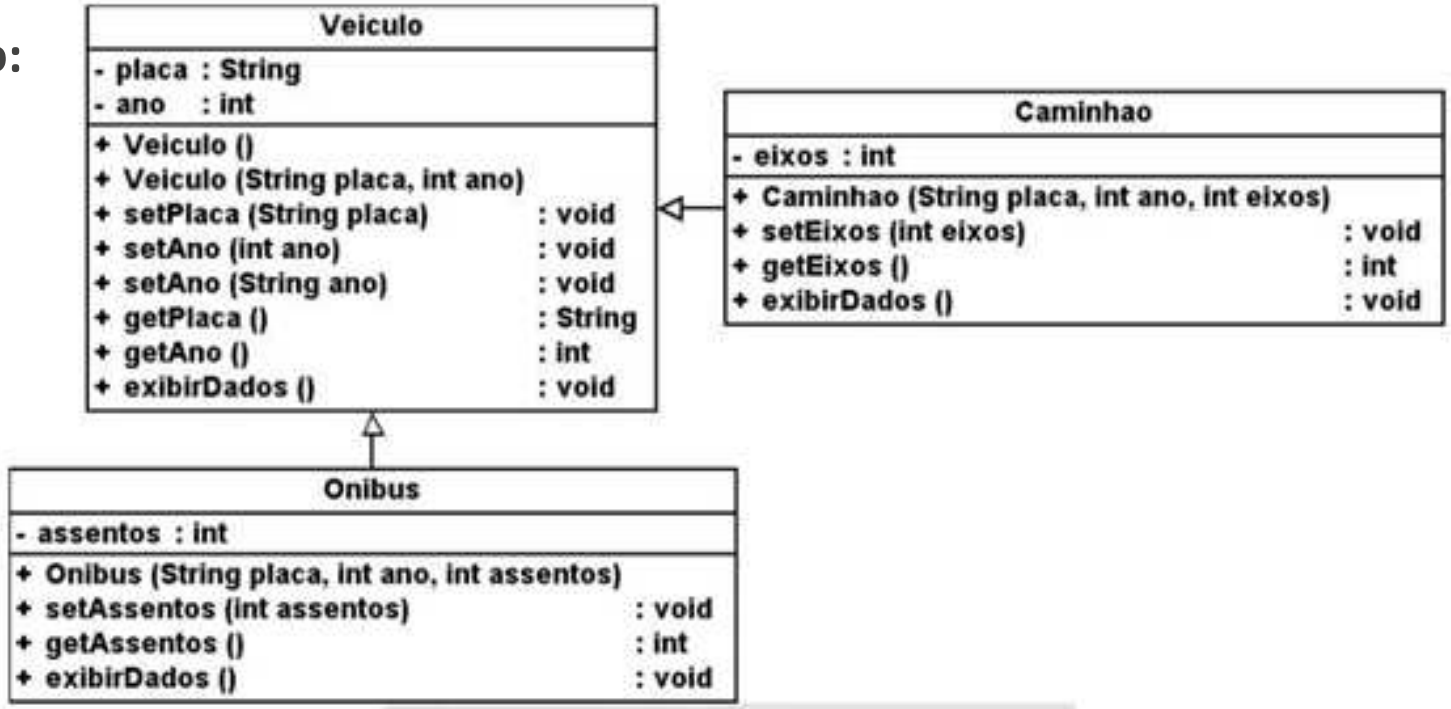


Polimorfismo:

- É o princípio em que classes derivadas de uma mesma superclasse **podem invocar operações que têm a mesma assinatura**, mas comportamentos diferentes em cada subclasse**, produzindo resultados diferentes**, dependendo de como cada objeto implementa a operação.

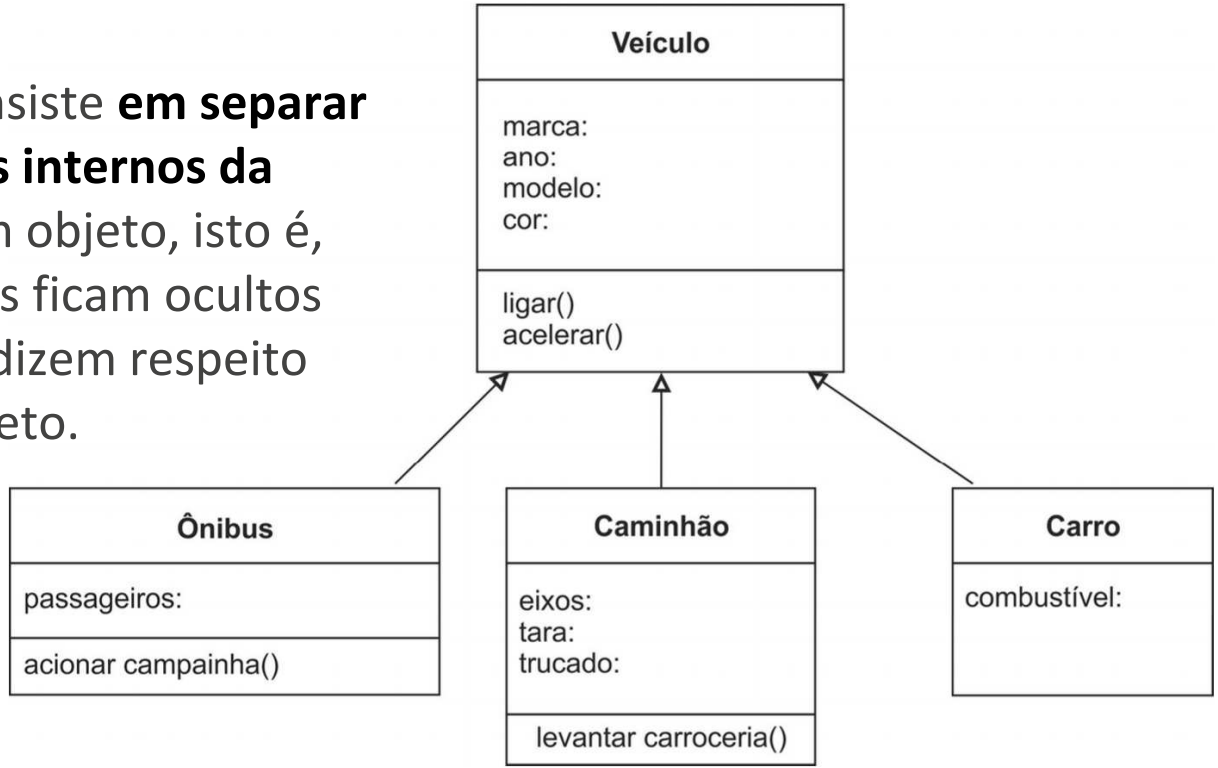


Polimorfismo:



Encapsulamento (ou ocultar informações):

- É uma técnica que consiste **em separar aspectos externos dos internos da implementação** de um objeto, isto é, determinados detalhes ficam ocultos aos demais objetos e dizem respeito apenas ao próprio objeto.



Visibilidade:

- A visibilidade é utilizada para indicar o nível de acessibilidade de um determinado atributo ou método, sendo representada à esquerda destes. Existem basicamente quatro modos de visibilidade: público, protegido, privado e pacote.

Privada (-): significa que somente os objetos da classe detentora do atributo ou método poderão enxergá-lo.
Protegida (#): determina que, além dos objetos da classe detentora do atributo ou método, também os objetos de suas subclasses poderão ter acesso a este.
Pública (+): determina que o atributo ou método pode ser utilizado por qualquer objeto.
Pacote (~): determina que o atributo ou método é visível por qualquer objeto dentro do pacote.

Visibilidade:

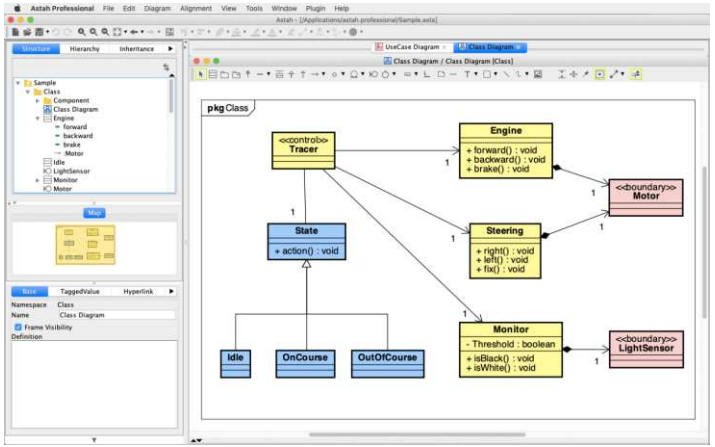
- A visibilidade é utilizada para indicar o nível de ace det mé rep des qua visi pro

Privada (-): significa que somente os objetos da classe detentora do atributo ou método poderão enxergá-lo.
Protegida (#): determina que, além dos objetos da classe detentora do atributo ou método, também os objetos de suas subclasses poderão ter acesso a este.
Pública (+): determina que o atributo ou método pode ser utilizado por qualquer objeto.
Pacote (~): determina que o atributo ou método é visível por qualquer objeto dentro do pacote.

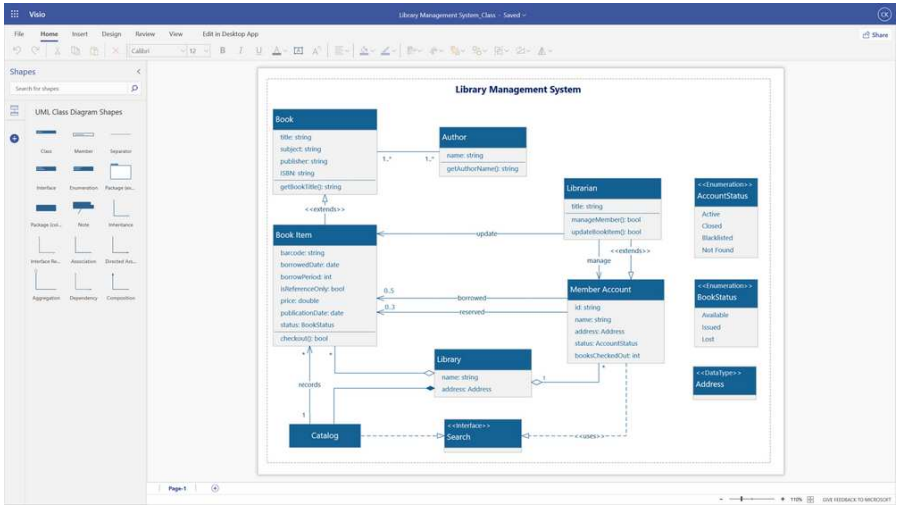
Análise e Projeto **Orientado a Objetos**

Pilares da OO:

UML - Ferramentas CASE



Visio

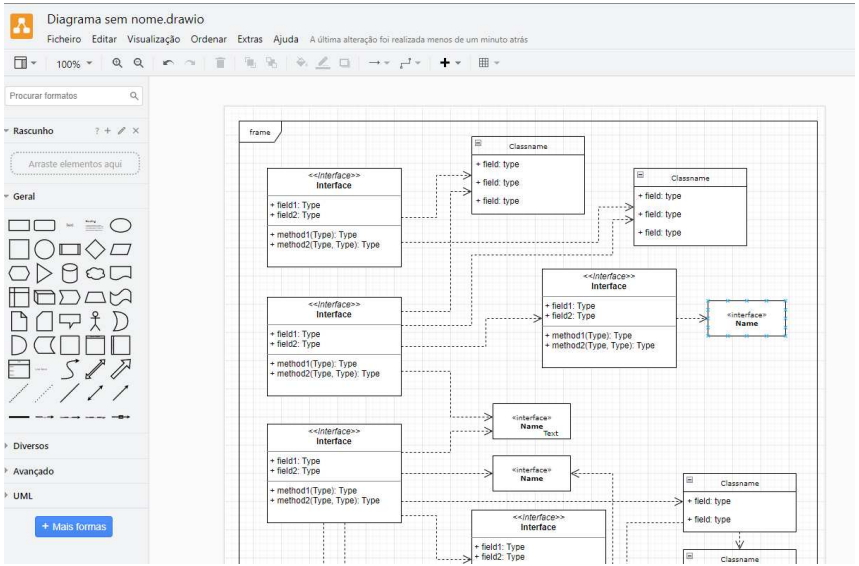


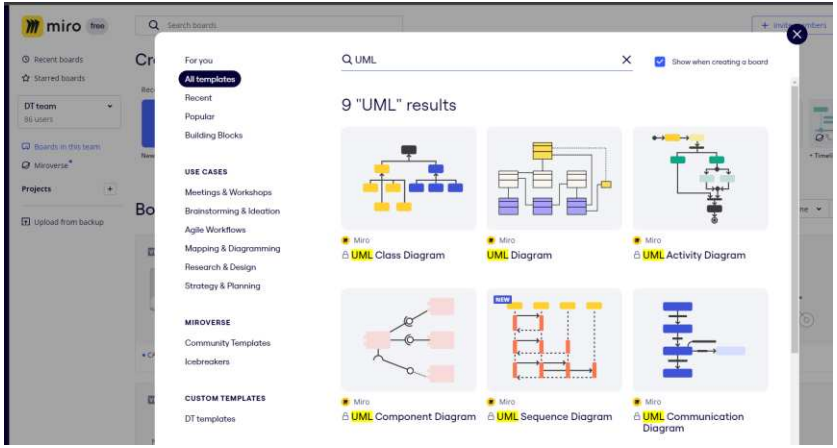
astah

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

UML - Ferramentas CASE

draw.io





miro

Lucidchart

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA

“Sucesso é o acúmulo de pequenos esforços, repetidos dia e noite.”

Robert Collier

UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA


UniCesumar
EDUCAÇÃO PRESENCIAL E A DISTÂNCIA