

CENTRO UNIVERSITÁRIO DE BELO
HORIZONTE - UNIBH

Modelagem de Software

Samara Soares Leal - samara.leal@prof.unibh.br



OBJETIVOS

ANÁLISE E PROJETO ORIENTADO A OBJETOS

Mapeamento dos relacionamentos entre as classes do software orientado a objetos para o desenvolvimento do projeto (Apoio à codificação).

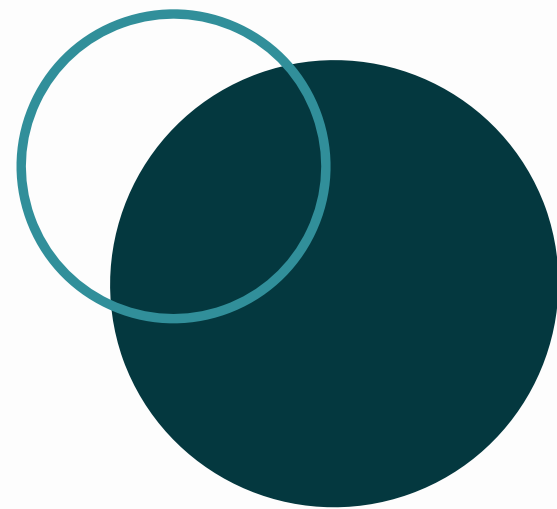
MODELAGEM DE PROBLEMAS

Modelagem dos requisitos do software a partir de diagramas da UML e user stories

MODELAGEM DE BANCO DE DADOS

Modelo entidade e relacionamento.
Modelo relacional e normalização.
Modelo lógico e físico de banco de dados

REPRESENTANDO GRAFICAMENTE O PROBLEMA COM O DIAGRAMA DE CLASSES



- DIAGRAMA DE CLASSES
- MAPEAMENTO DOS RELACIONAMENTOS ENTRE AS CLASSE PARA O DESENVOLVIMENTO DO PROJETO (APOIO À CODIFICAÇÃO).

DIAGRAMA DE CLASSES

Apresenta o relacionamento entre as classes do sistema.

Os requisitos funcionais do sistema são implementados a partir do relacionamento entre os objetos das classes mapeadas no diagrama de classes.

Exemplo: O usuário deve conseguir realizar o empréstimo de livros.

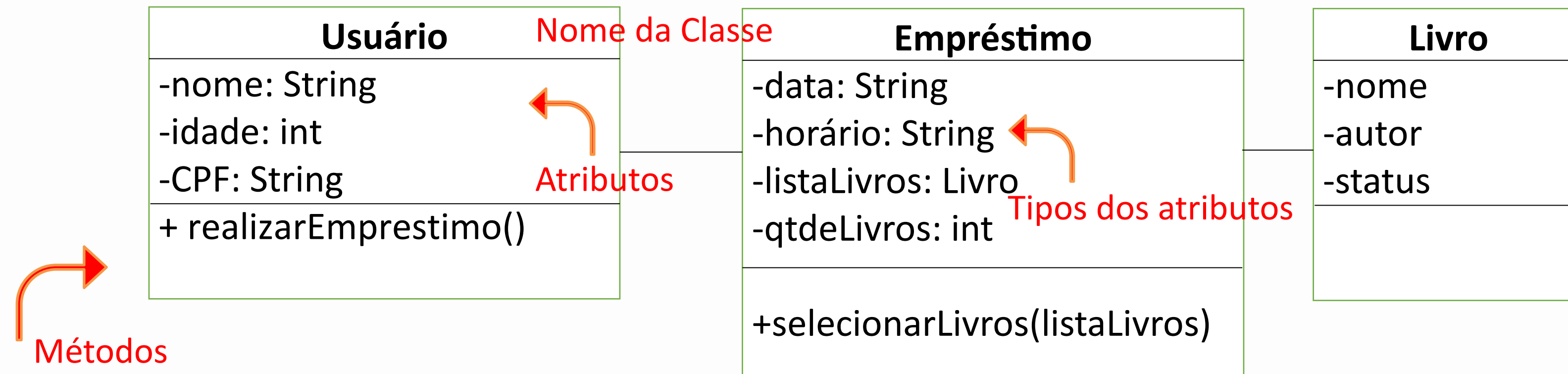


DIAGRAMA DE CLASSES

REPRESENTAÇÃO GRÁFICA DO RELACIONAMENTO ENTRE AS CLASSES

O que colocar no diagrama de classes:

- Classes que representam abstração dos objetos inerentes ao domínio do problema.
- Os relacionamentos entre as classes representam as funcionalidades do sistema.

O que **não** colocar no diagrama de classes:

- Classes que representam interfaces gráficas.
- Classes de conexão e acesso a banco de dados.

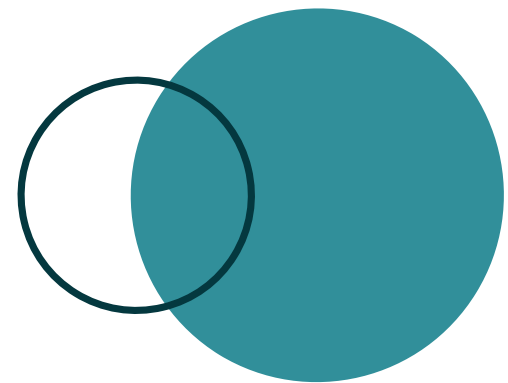


DIAGRAMA DE CLASSES

VISIBILIDADE

- **Privado (*private*)**: visível dentro da própria classe
- + **Público (*public*)**: visível por qualquer classe
- # **Protegido (*protected*)**: visível dentro da própria classe e classes hierárquicas

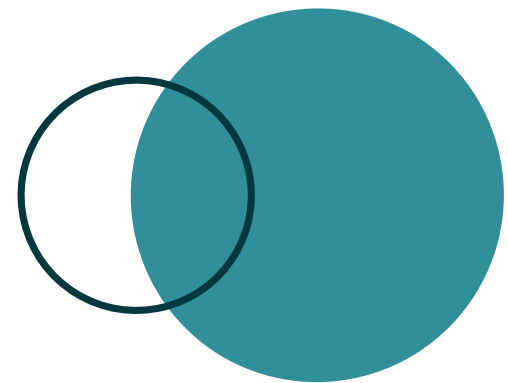
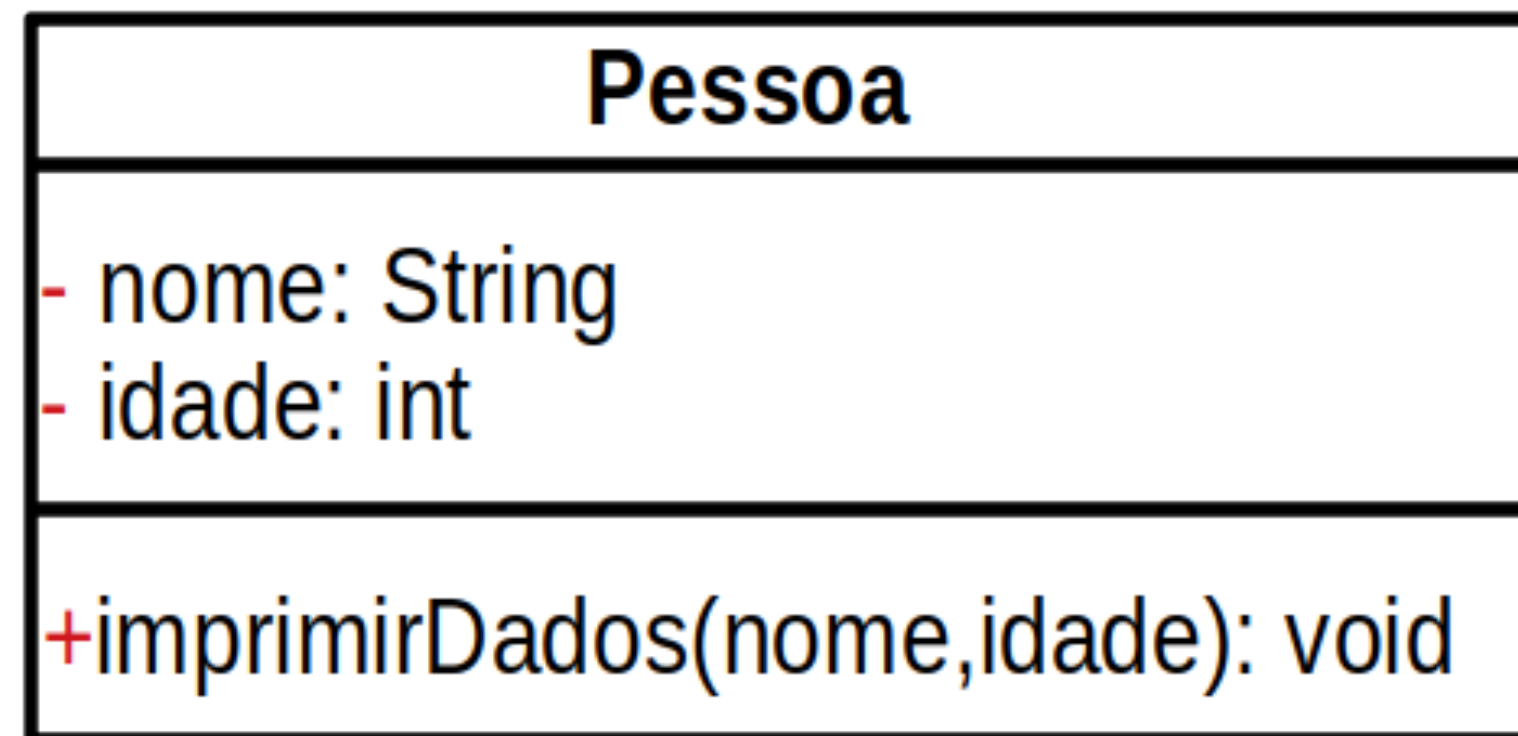


DIAGRAMA DE CLASSES

MULTIPLICIDADE

- Indicação de quantos objetos podem preencher a propriedade.

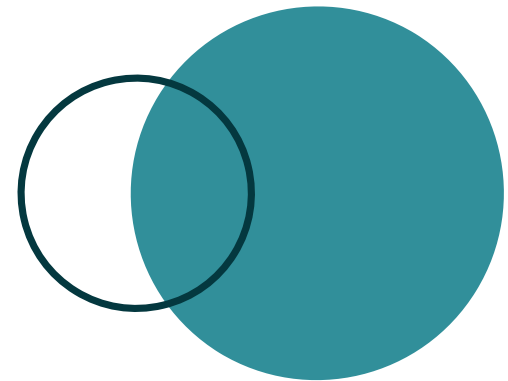
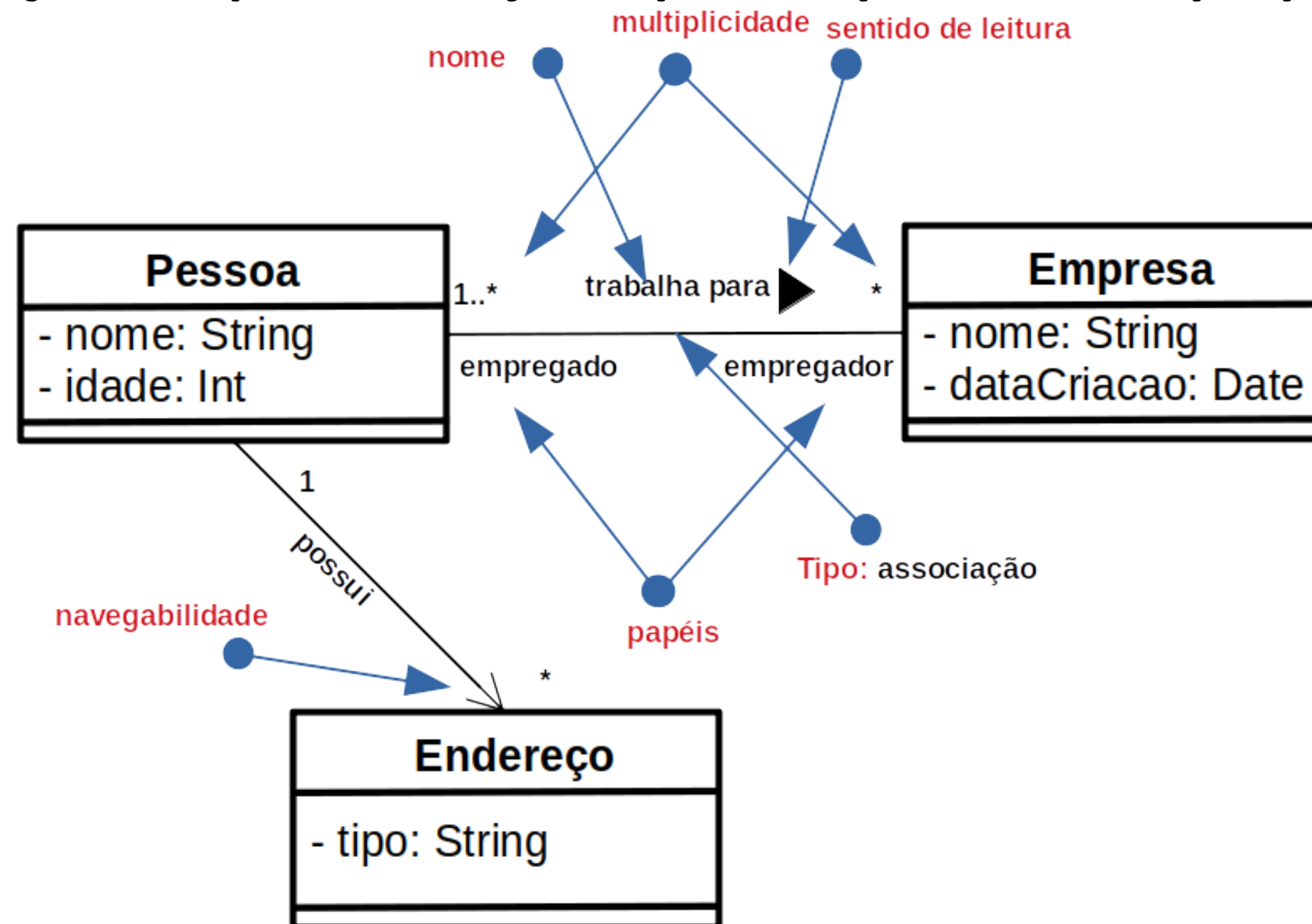


DIAGRAMA DE CLASSES

ASSOCIAÇÕES

- Representada por uma linha ligando duas classes.

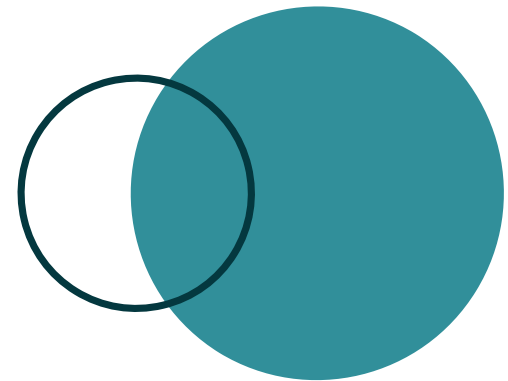
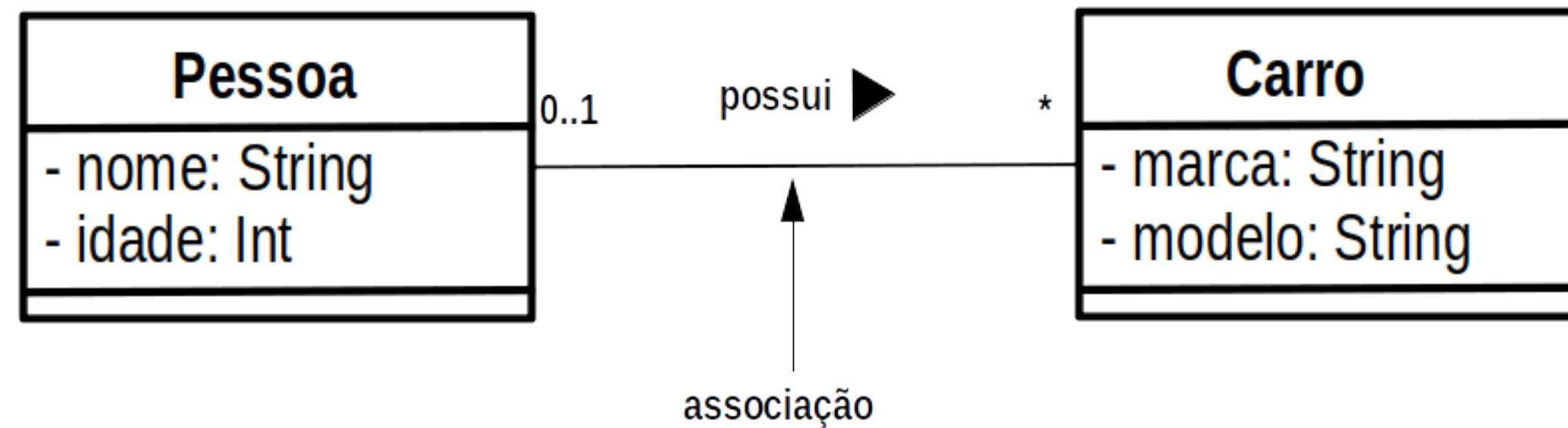


DIAGRAMA DE CLASSES

CONECTIVIDADE

- Tipo de associação entre duas classes.

| Conectividade | Multiplicidade de um extremo | Multiplicidade do outro extremo |
|--------------------|------------------------------|---------------------------------|
| Um para um | 0..1 ou 1 | 0..1 ou 1 |
| Um para muitos | 0..1 ou 1 | * ou 1..* ou 0..* |
| Muitos para muitos | * ou 1..* | * ou 1..* ou 0..* |

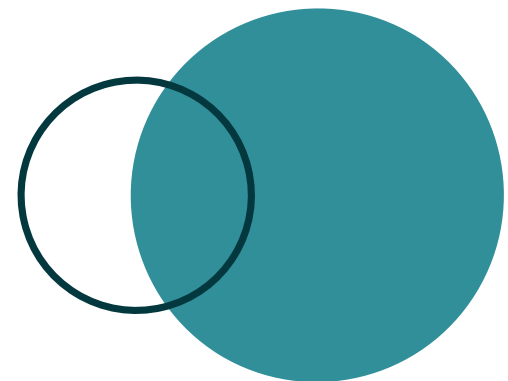
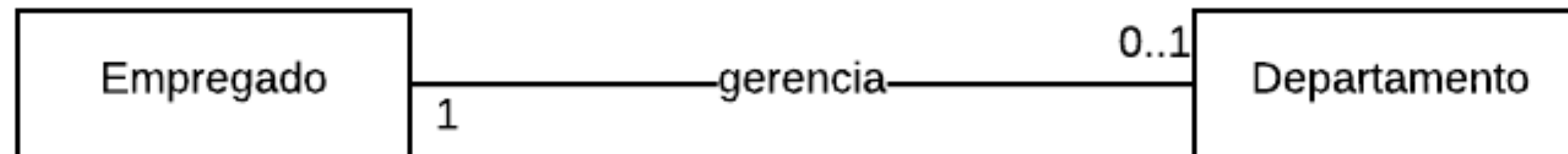


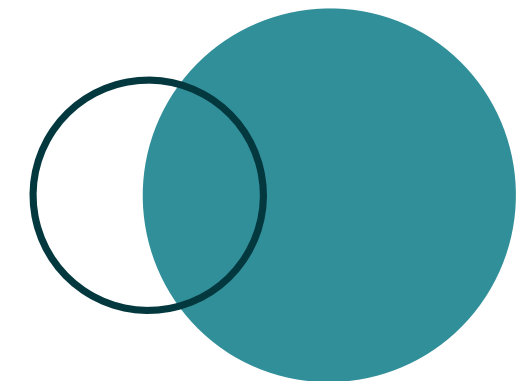
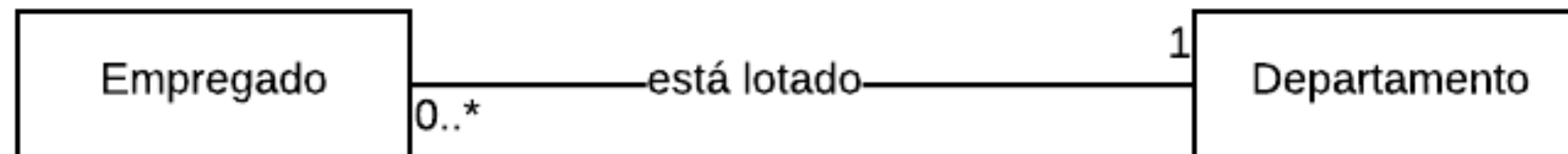
DIAGRAMA DE CLASSES

CONECTIVIDADE

Um para um



Um para muitos



Muitos para muitos



DIAGRAMA DE CLASSES

ASSOCIAÇÃO TERNÁRIA

- Quando uma associação entre duas classes gera uma nova classe

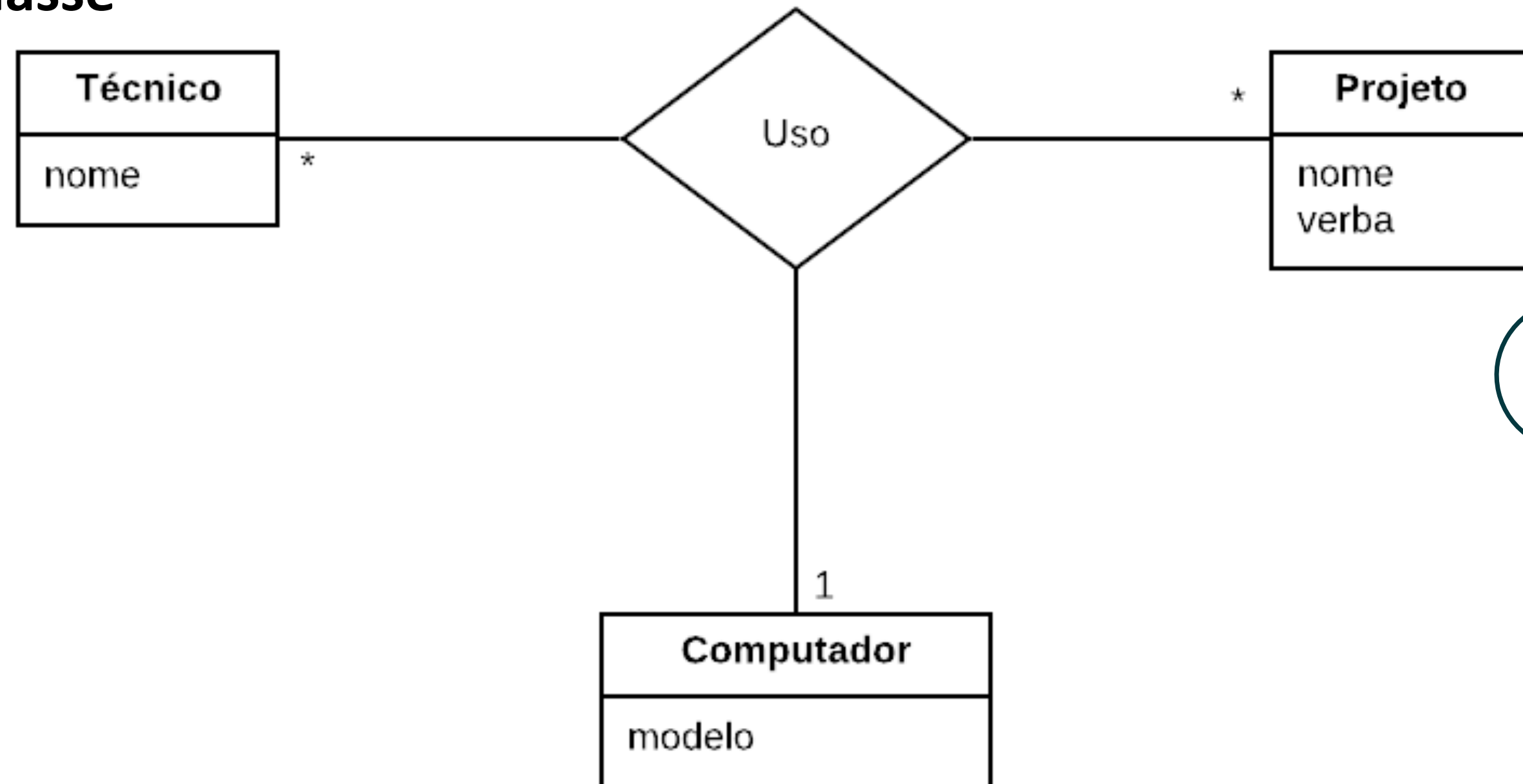


DIAGRAMA DE CLASSES

ASSOCIAÇÃO TERNÁRIA

- Associação reflexiva (auto associação) liga objetos da mesma classe.

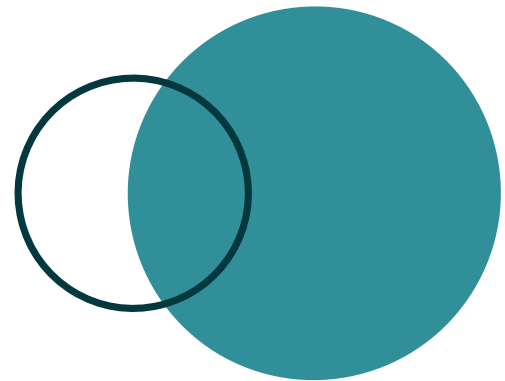
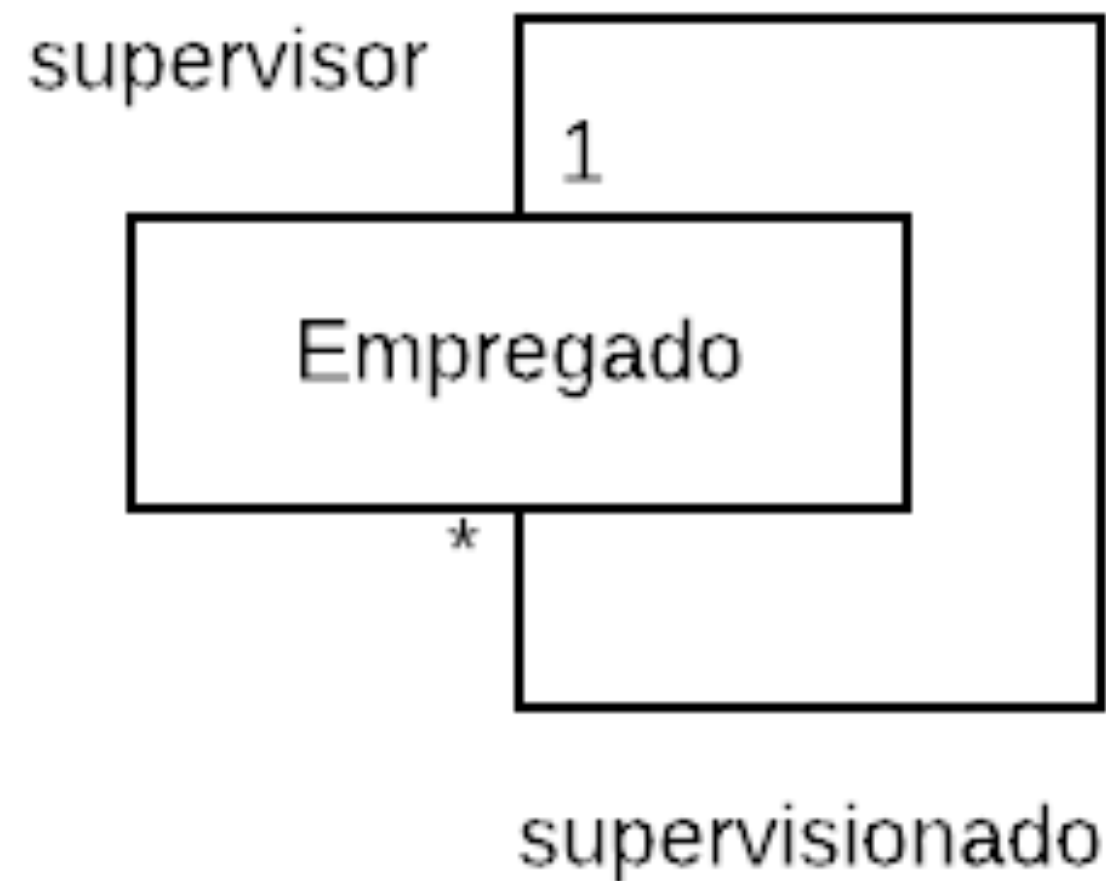


DIAGRAMA DE CLASSES

ASSOCIAÇÃO TODO-PARTE

1. Agregação:

- “Todo-parte” não-exclusivo.
- Uma classe faz **parte de** outra classe, mas de forma **não-exclusiva**.
- Se excluir a classe responsável pelo relacionamento, **não** precisa excluir a classe que ele possui relacionamento.

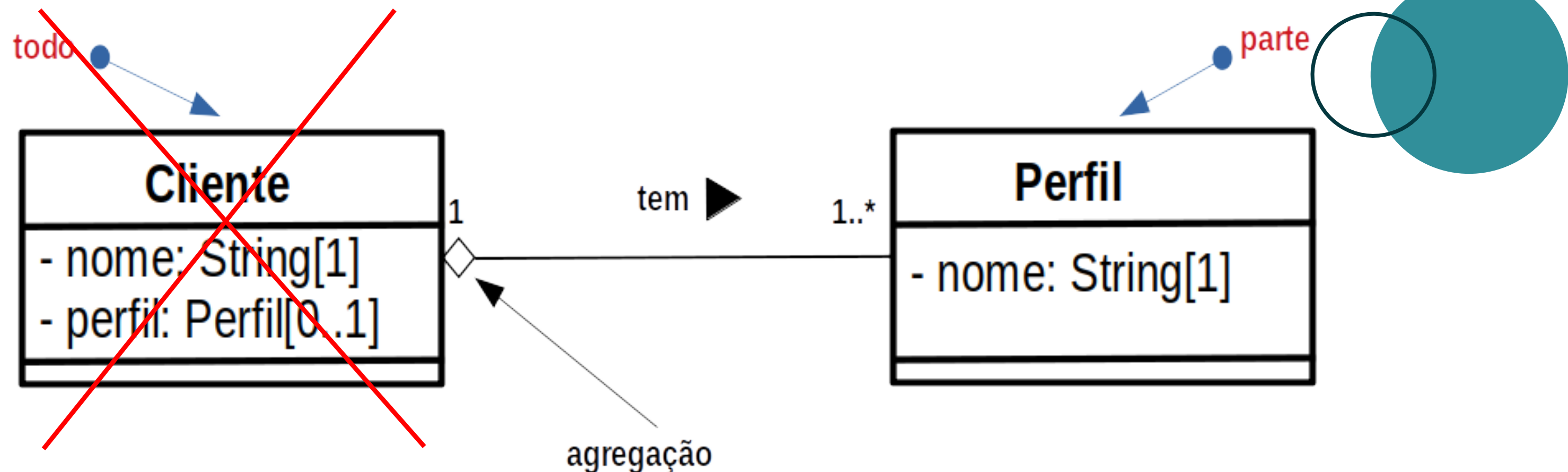


DIAGRAMA DE CLASSES

ASSOCIAÇÃO TODO-PARTE

2. Composição:

- “Todo-parte” exclusivo.
- Uma classe faz **parte de** outra classe de **forma exclusiva**.
- Se excluir a classe responsável pelo relacionamento, precisa **excluir** a classe que ele possui relacionamento.

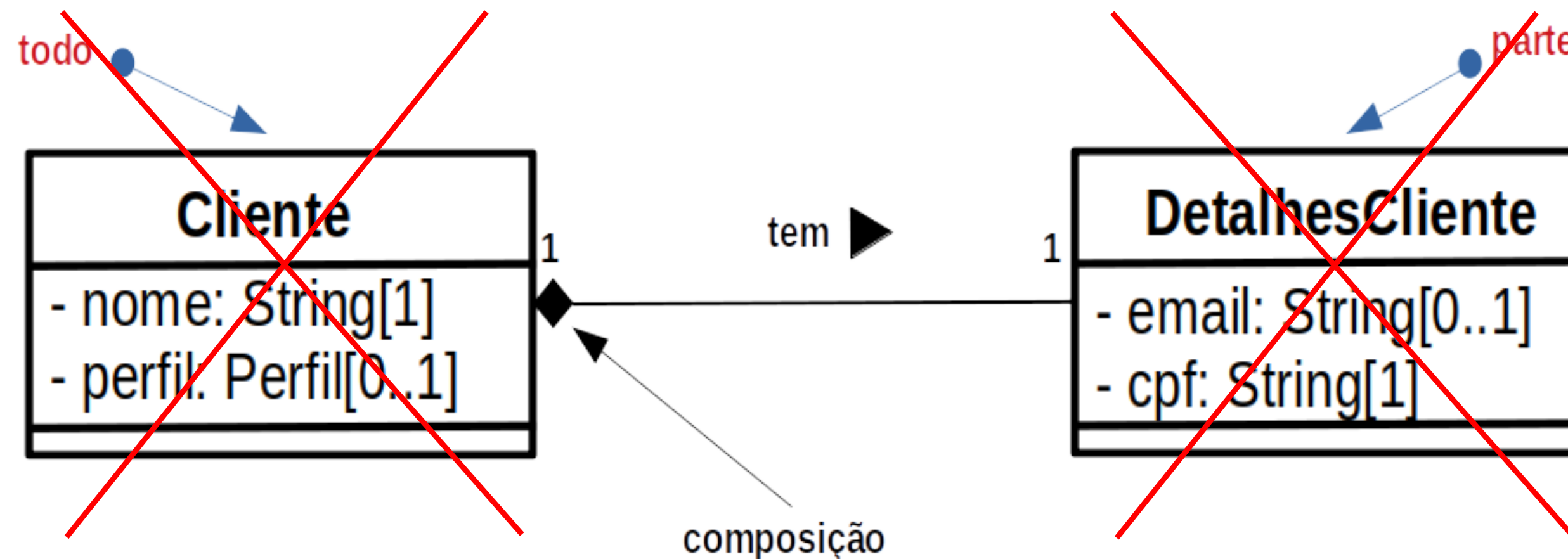


DIAGRAMA DE CLASSES

GENERALIZAÇÃO

- Relacionamento de **herança** entre um elemento geral (superclasse) e outro específico (subclasse).

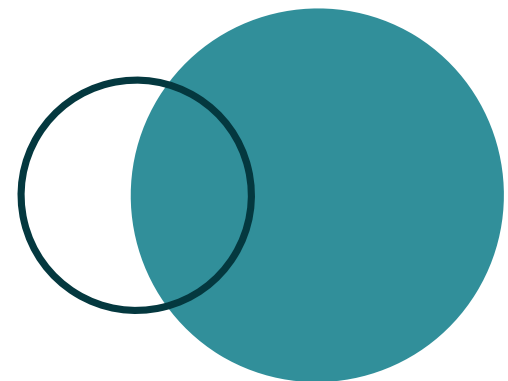
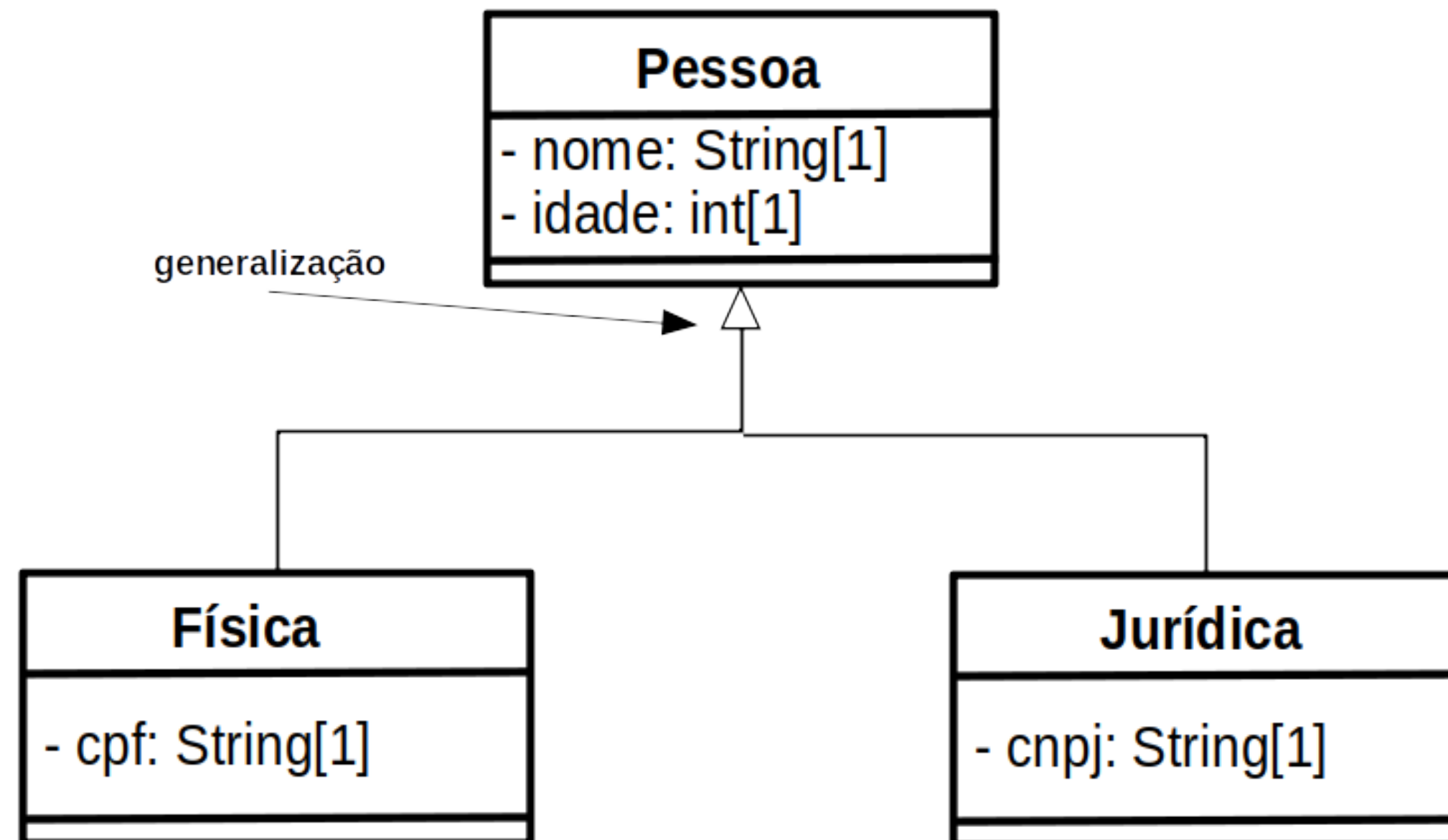


DIAGRAMA DE CLASSES

DEPENDÊNCIA

- Uma dependência entre dois elementos existe se mudanças na definição de um elemento (**fornecedor**) podem causar mudanças ao outro (**cliente**).
- Se uma classe muda a sua interface, qualquer mensagem enviada para essa classe pode não ser mais válida.

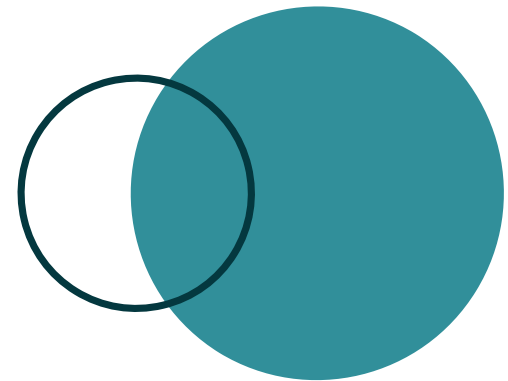
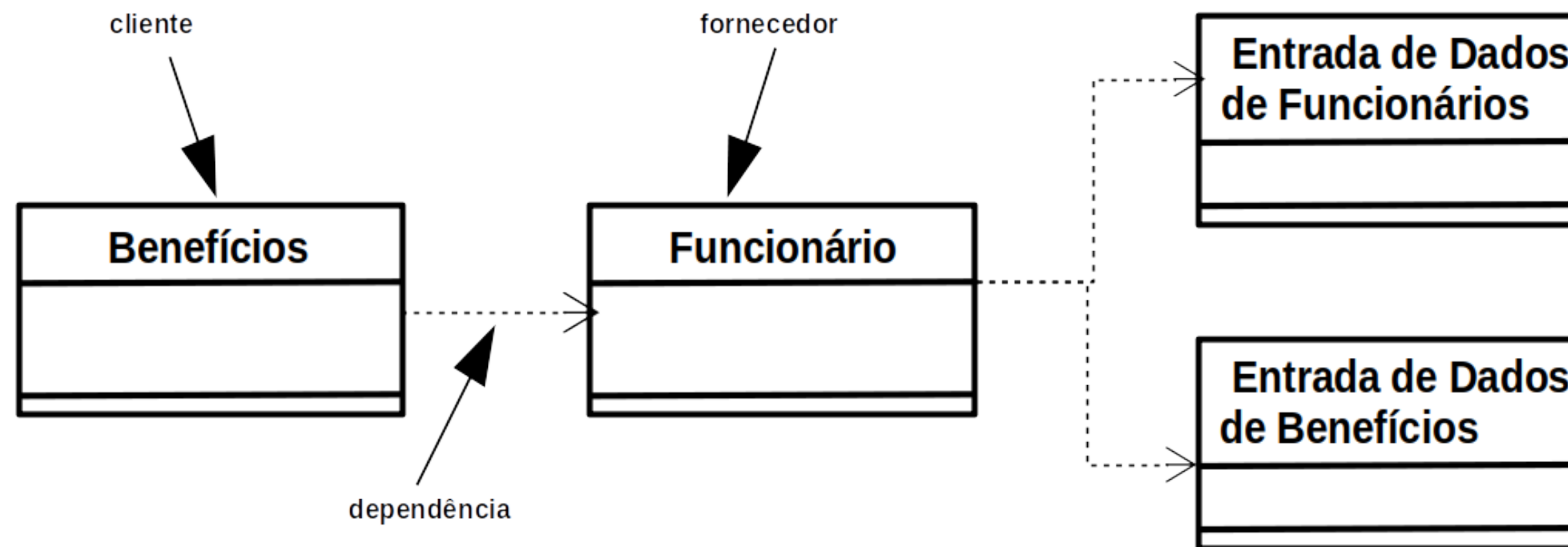


DIAGRAMA DE CLASSES

NOTAS E COMENTÁRIOS

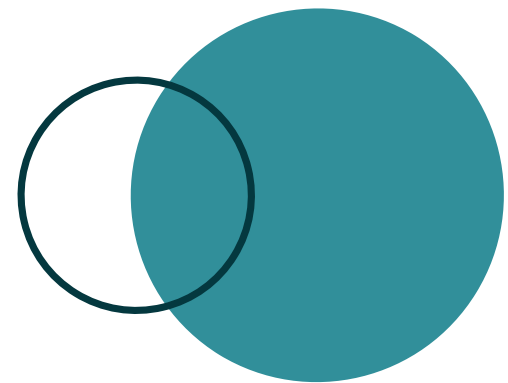
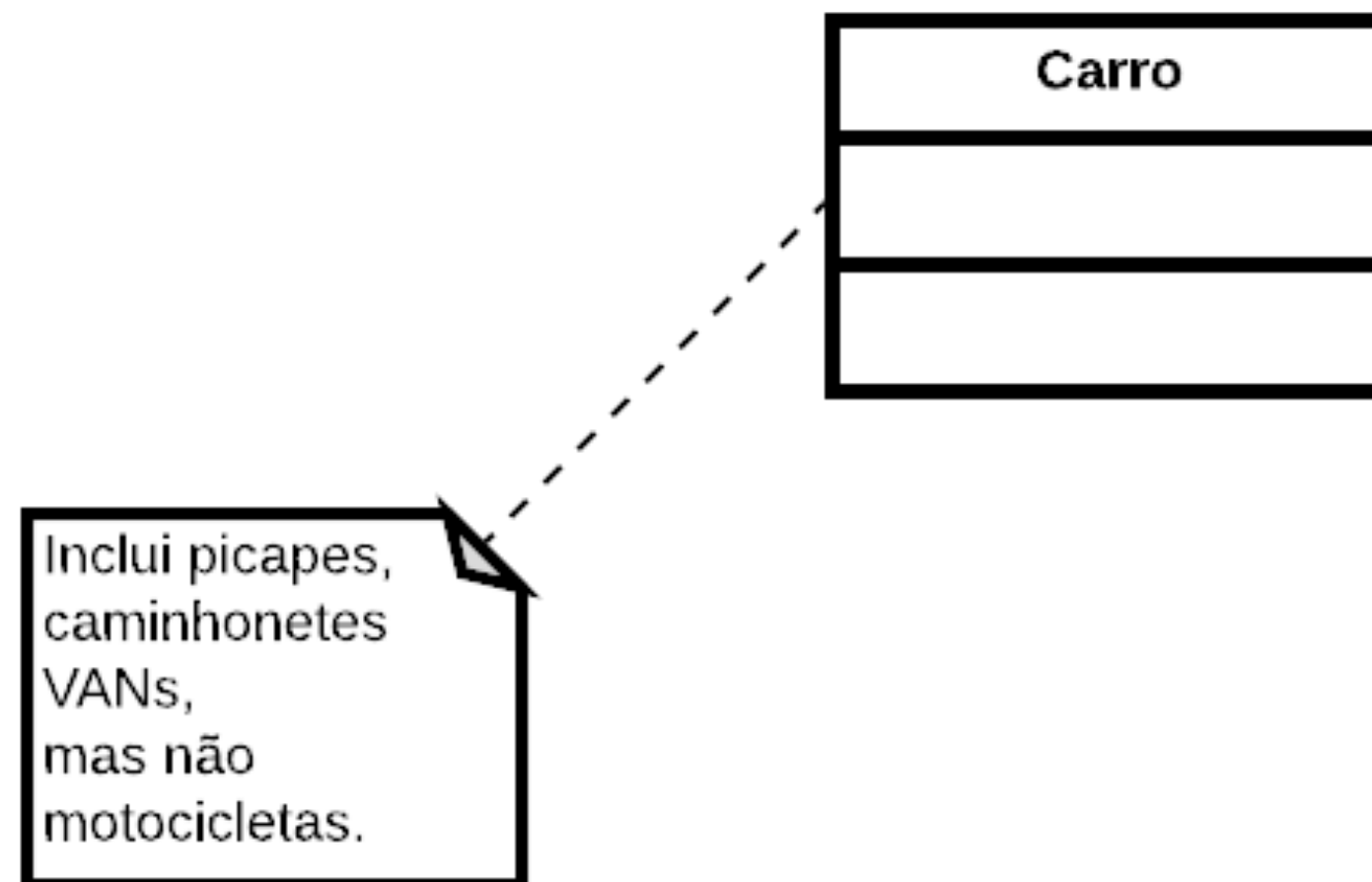


DIAGRAMA DE CLASSES

RESTRIÇÕES

- A UML permite que seja usada qualquer coisa para descrever restrições.

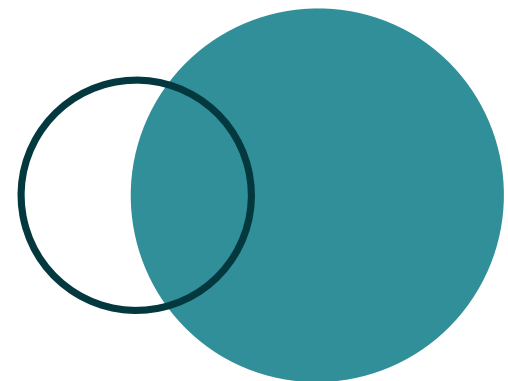
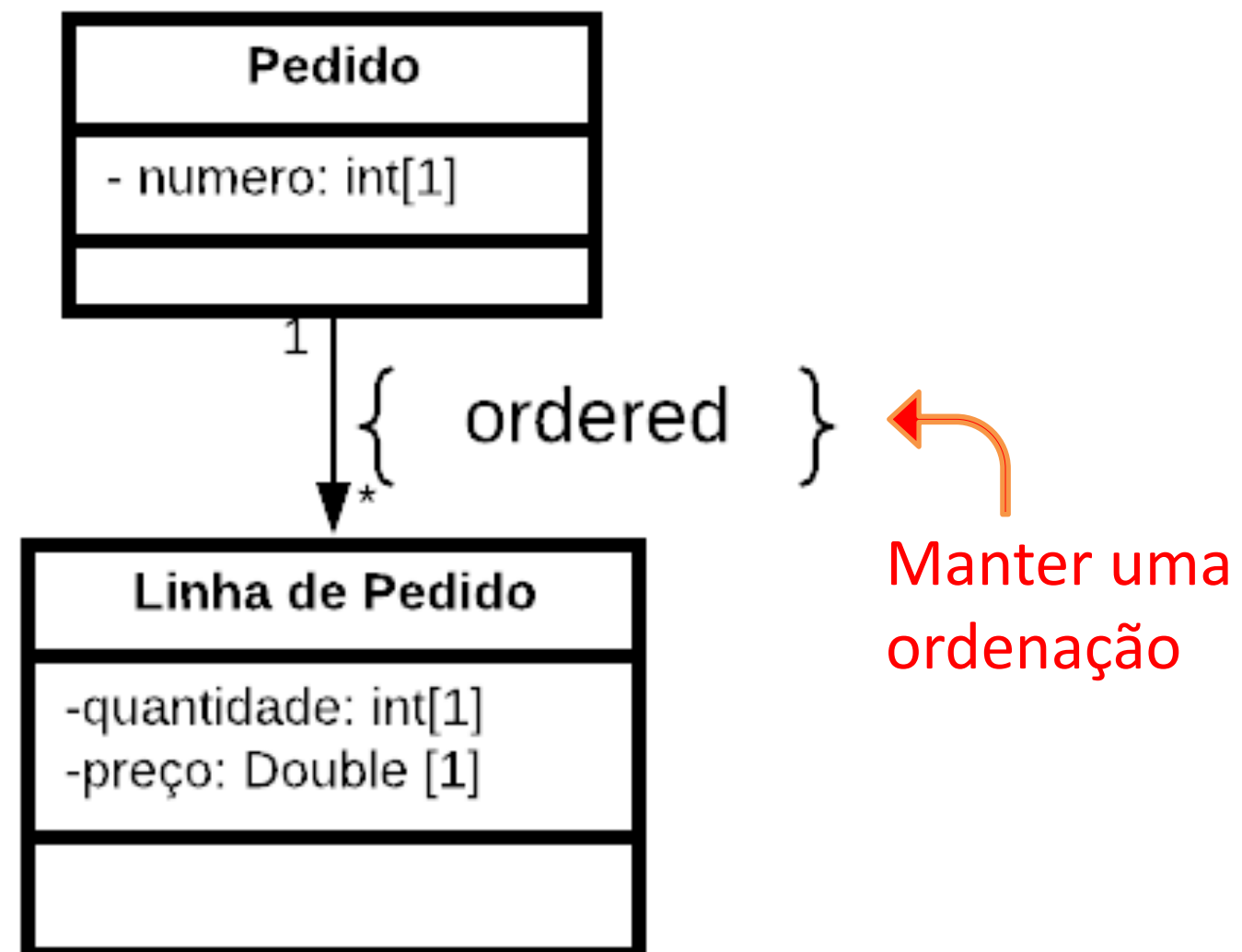


DIAGRAMA DE CLASSES

CLASSES ASSOCIATIVAS

- Classes ligadas à associações, em vez de estarem ligadas a outras classes.
- Esse tipo de classe normalmente aparece quando é necessário manter informações da associação de outras classes.

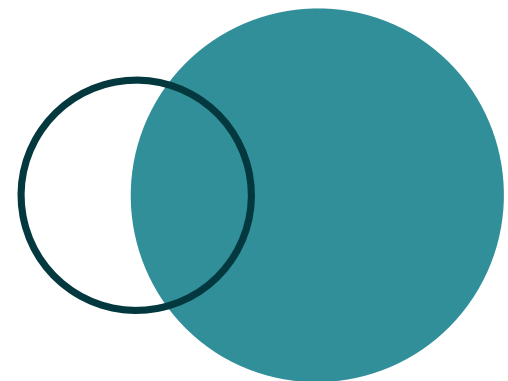
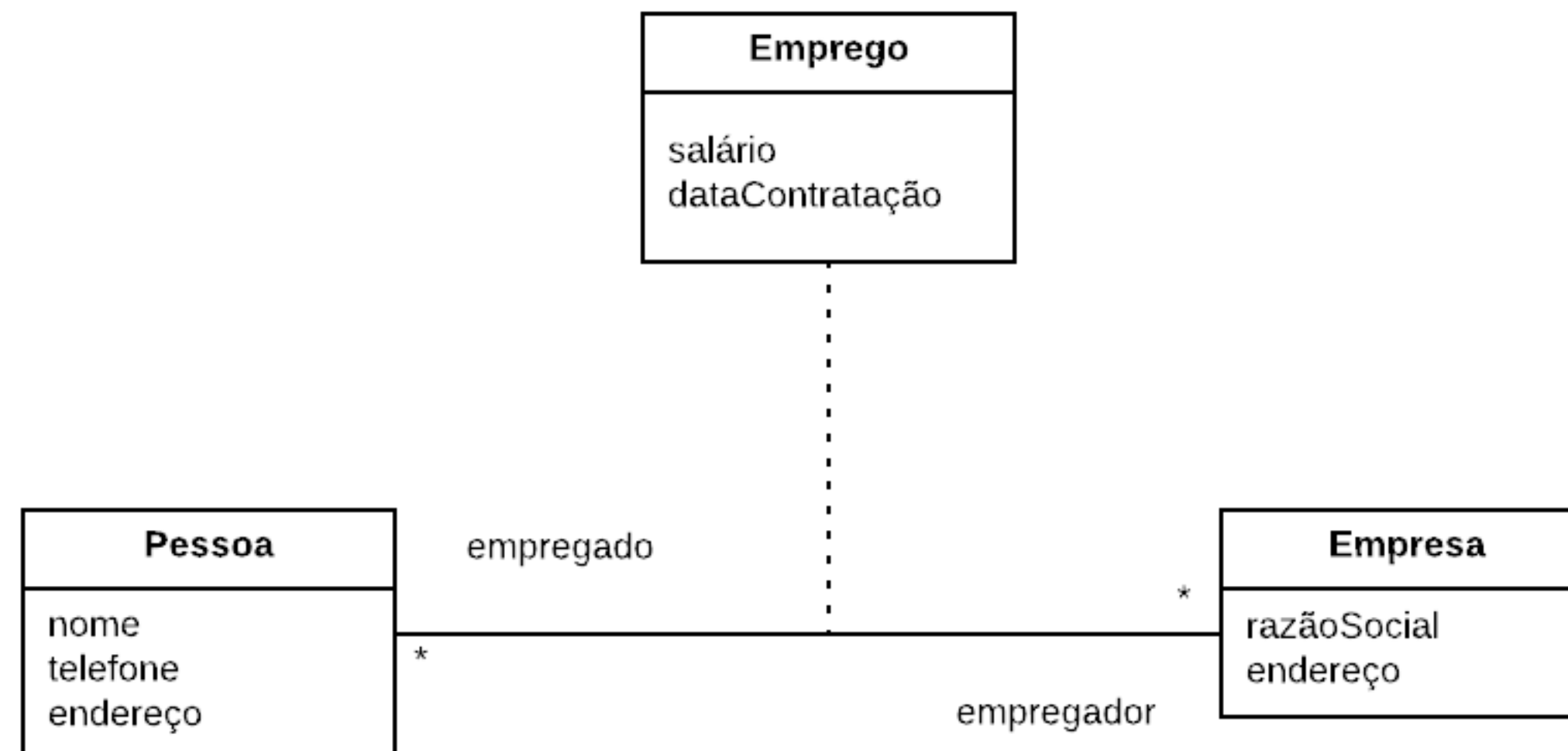


DIAGRAMA DE CLASSES

CLASSES ABSTRATAS

- Não pode ser instanciada diretamente.

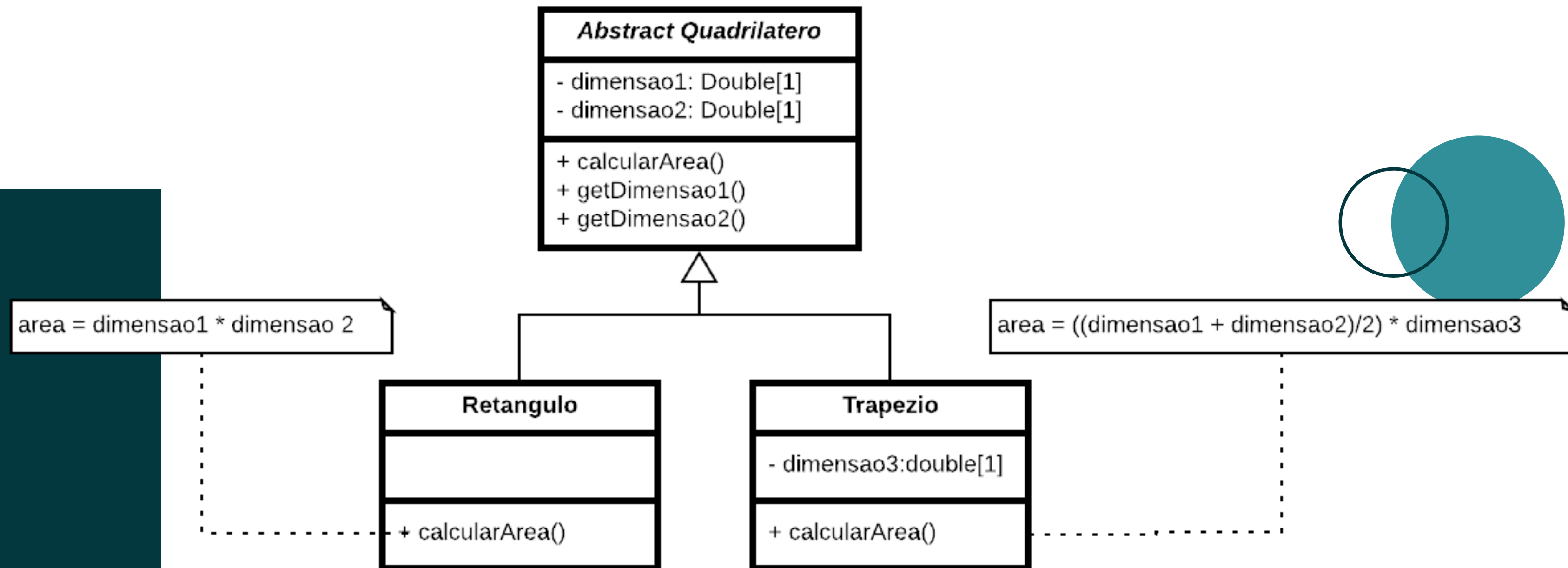


DIAGRAMA DE CLASSES

INTERFACE

- Não tem implementação (um contrato).
- Métodos declarados sem corpo (**abstratos**).
- Para implementar uma interface, a classe deve implementar o conjunto completo de métodos definidos.

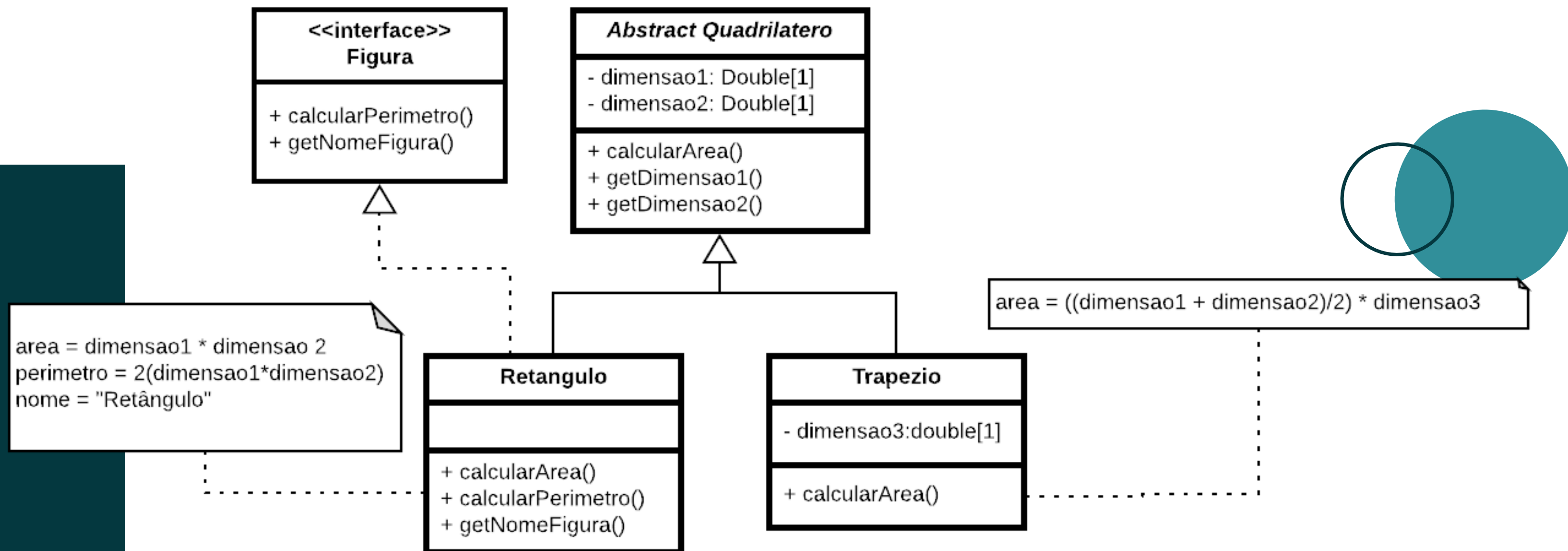


DIAGRAMA DE CLASSES

CLASSE ATIVA

- É uma classe cujos objetos possuem um ou mais processos ou threads e portanto podem iniciar atividade de controle.

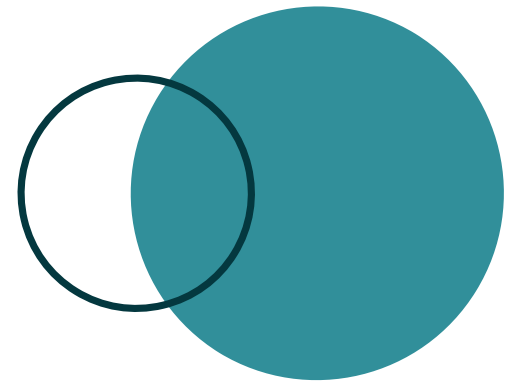


DIAGRAMA DE CLASSES

ENUMERAÇÃO

- Usadas para mostrar um conjunto fixo de valores que não possuem quaisquer propriedades além de seu valor simbólico.

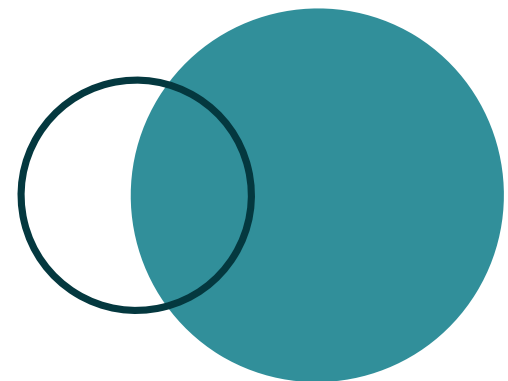
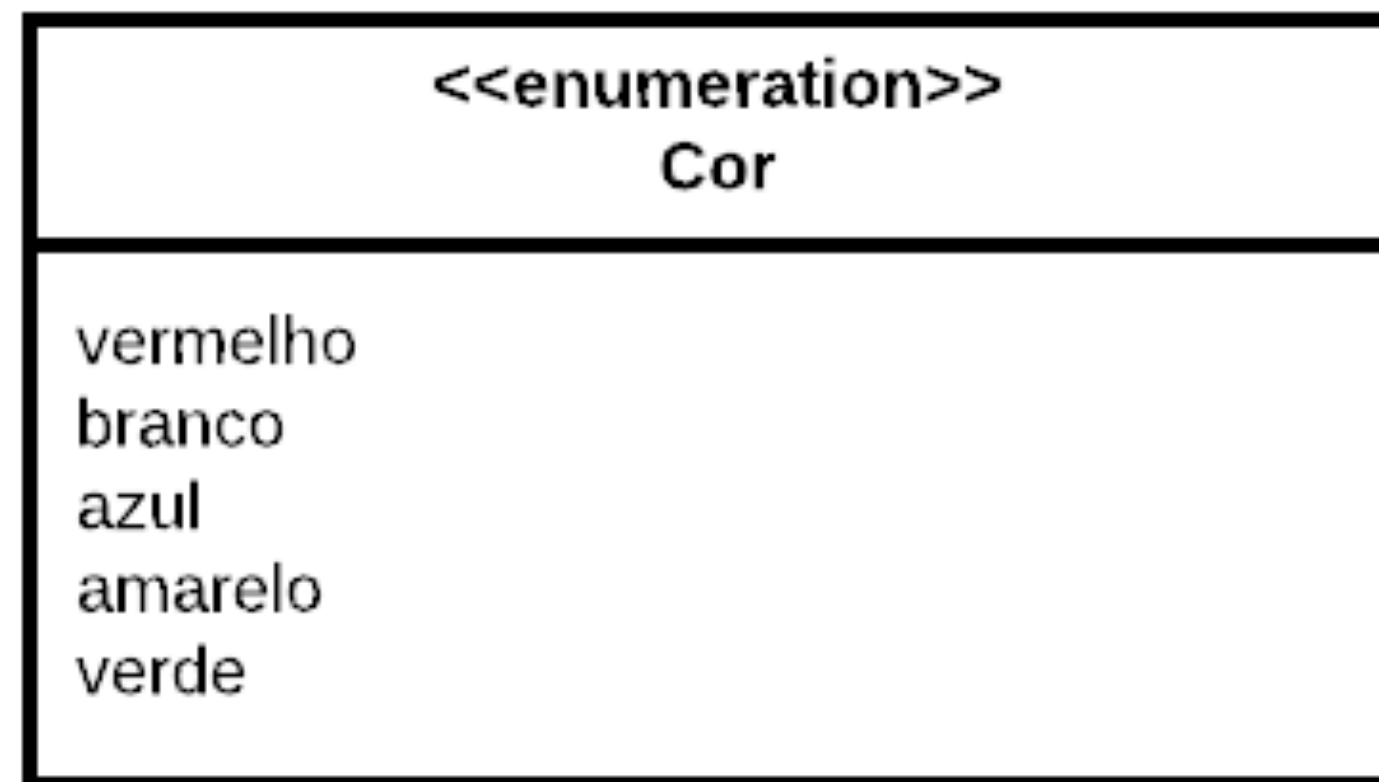
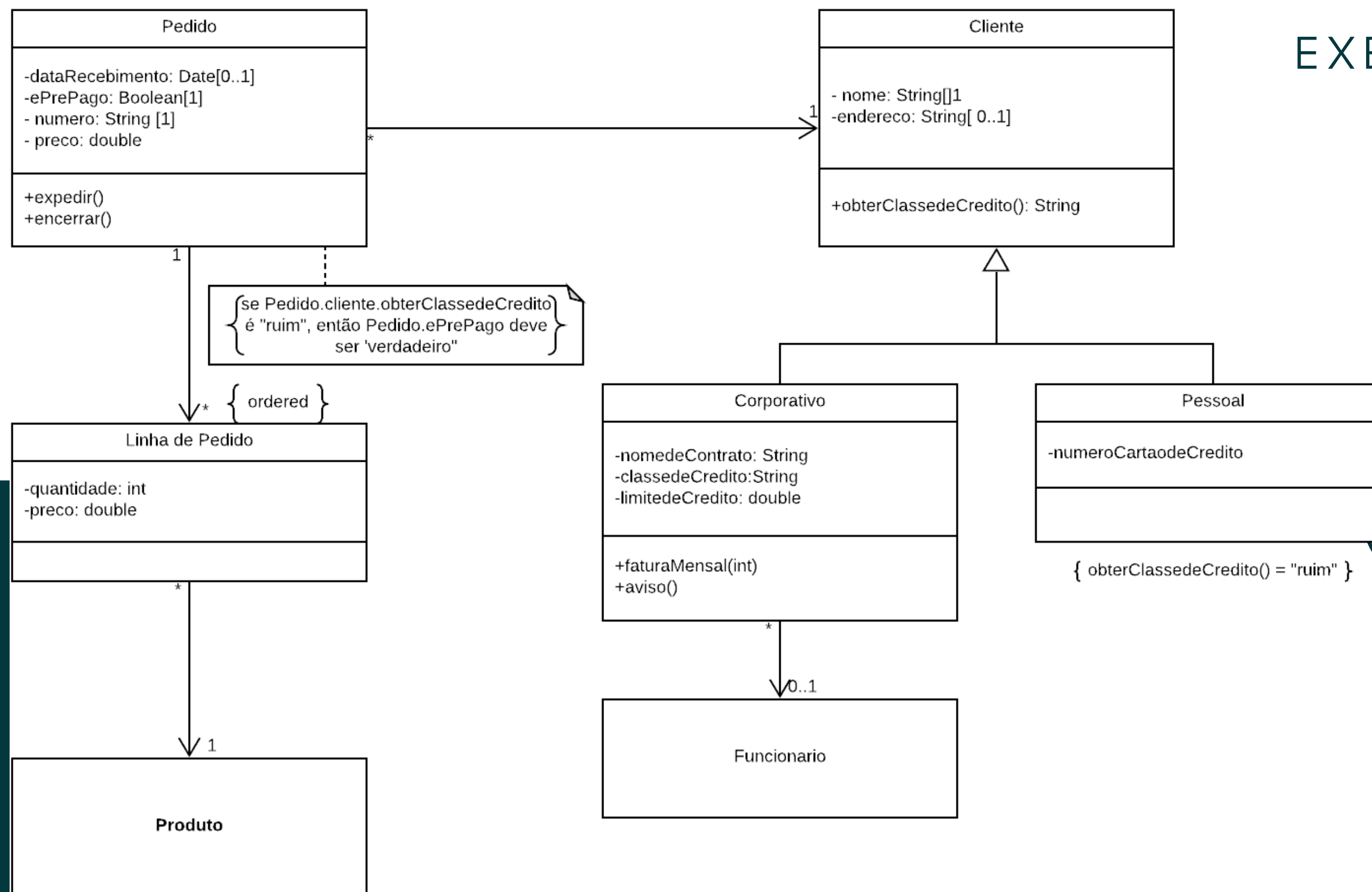


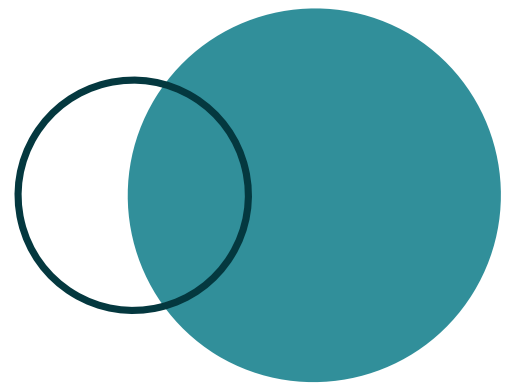
DIAGRAMA DE CLASSES

EXEMPLO



DÚVIDAS?

ENTRE EM CONTATO: *samara.leal@prof.unibh.br*



NÃO SE ESQUEÇA DE CONSULTAR O REFERENCIAL
BIBLIOGRÁFICO e BUSCA ATIVA!