



UML - DIAGRAMA DE CLASSES

- Diagrama mais utilizado da UML.
- Apresenta uma visão estática de como as classes estão organizadas a fim de definir sua estrutura lógica.
- Foi projetado para ser uma evolução (e não substituição) do Modelo Entidade-Relacionamento do Banco de Dados.
- Um diagrama de classes é uma representação da estrutura e relações das classes que servem de modelo para objetos.



Visualização



Construção



Comunicação



Especificação



Documentação

OBJETOS

- O mundo real é formado de “coisas” que são úteis para realização de determinadas tarefas.
- Na orientação a objetos, “coisas” do mundo real são denominadas objetos.



OBJETOS

- Os seres humanos costumam agrupar os objetos para entendê-los.
- A Orientação a Objetos utiliza-se de um componente chamado classe, que tem por objetivo agrupar objetos semelhantes necessários para realização das funcionalidades do sistema em estudo.

Classe dos equipamentos de Informática

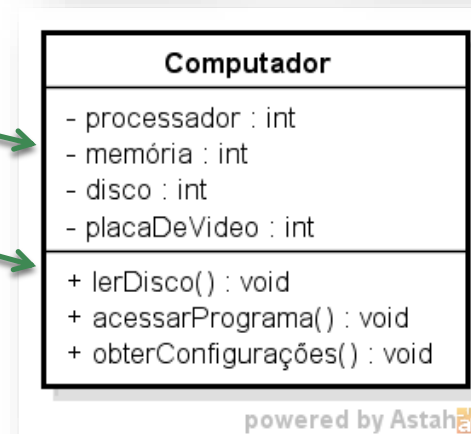


Classe dos equipamentos de Comunicação



CLASSE DE OBJETOS

- Juntar objetos em uma classe é o mesmo que classificá-los por possuírem a mesma estrutura de dados (atributos) e o mesmo comportamento (operações).



CLASSE

- Uma classe é representada por uma “caixa” com, no máximo, três compartimentos:
- Nome da classe: por convenção esse nome é apresentado no singular e com as palavras componentes começando por maiúsculas;
- Atributos: correspondem às informações que um objeto armazena;
- Operações: correspondem às ações que um objeto sabe realizar.

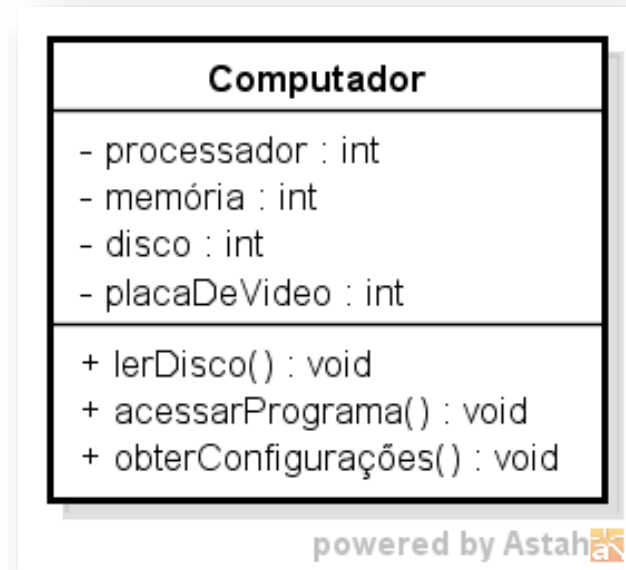
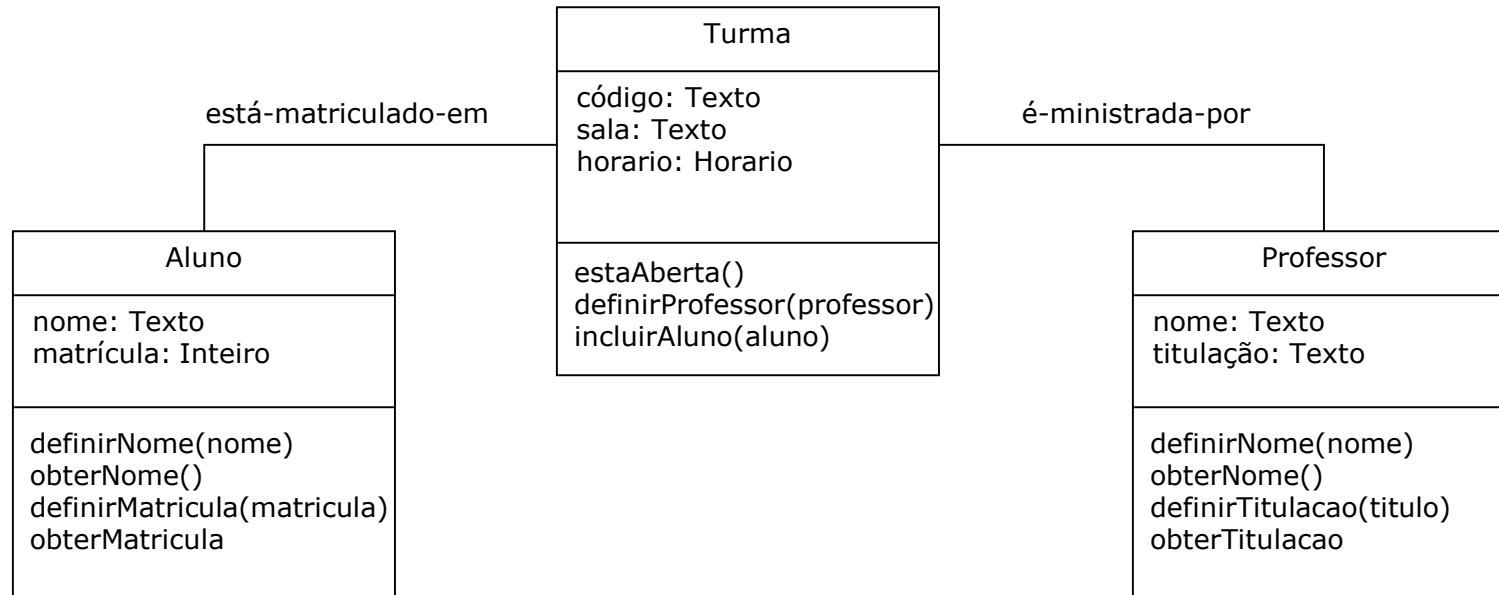


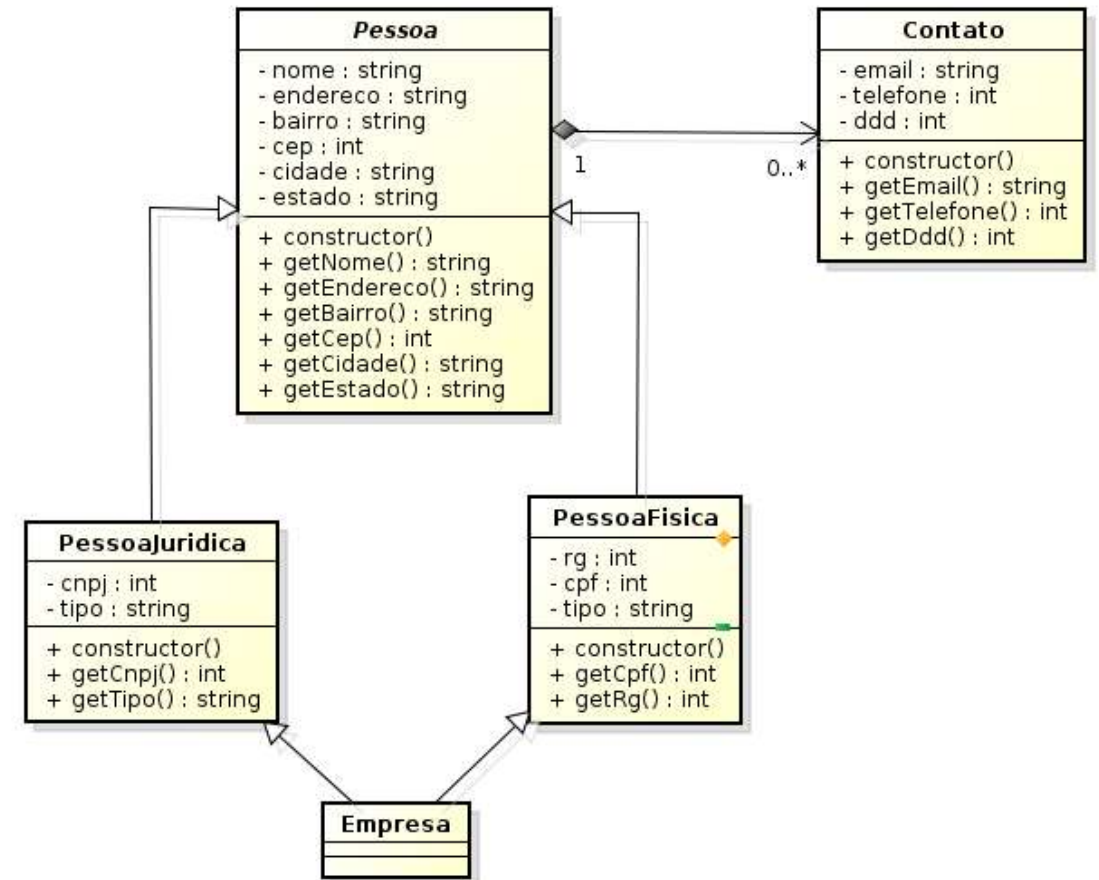
DIAGRAMA DE CLASSES

- Mostra um conjunto de classes e seus relacionamentos.
- É o diagrama central da modelagem orientada a objetos.



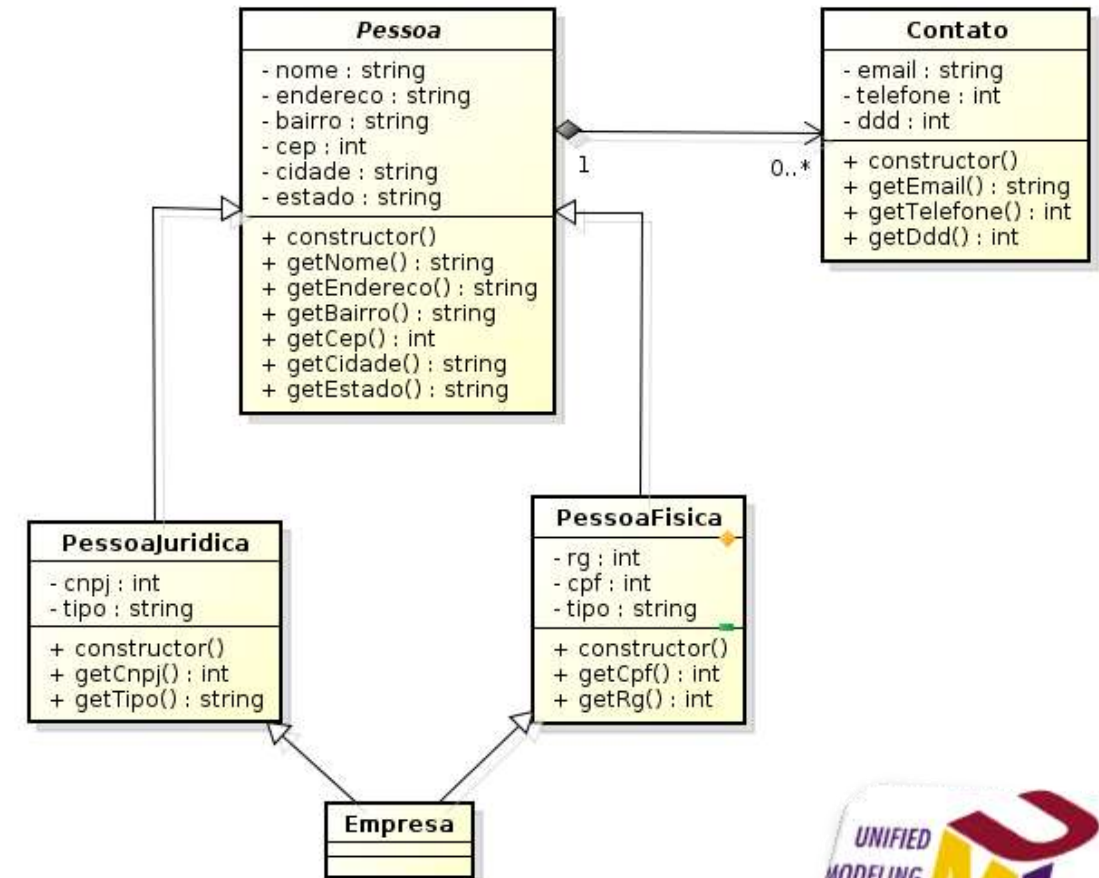
CLASSES

- Graficamente, as classes são representadas por retângulos incluindo nome, atributos e métodos.



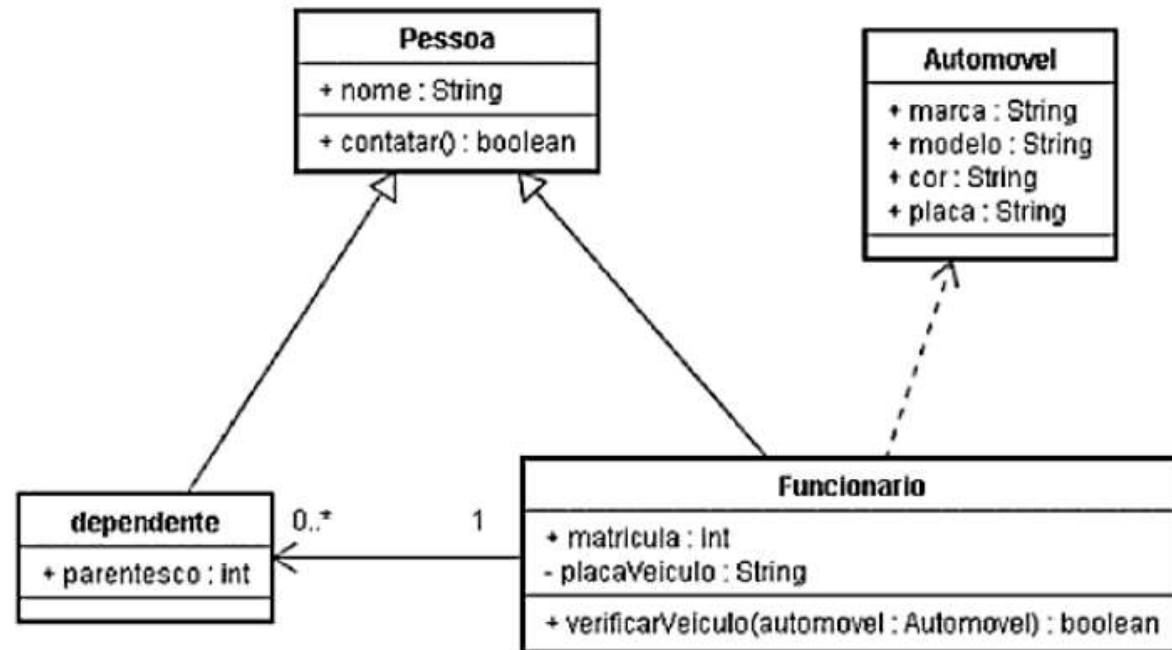
CLASSES

- Devem receber nomes de acordo com o vocabulário do domínio do problema.
- É comum adotar um padrão para nomeá-las
- Ex: todos os nomes de classes serão substantivos singulares com a primeira letra maiúscula



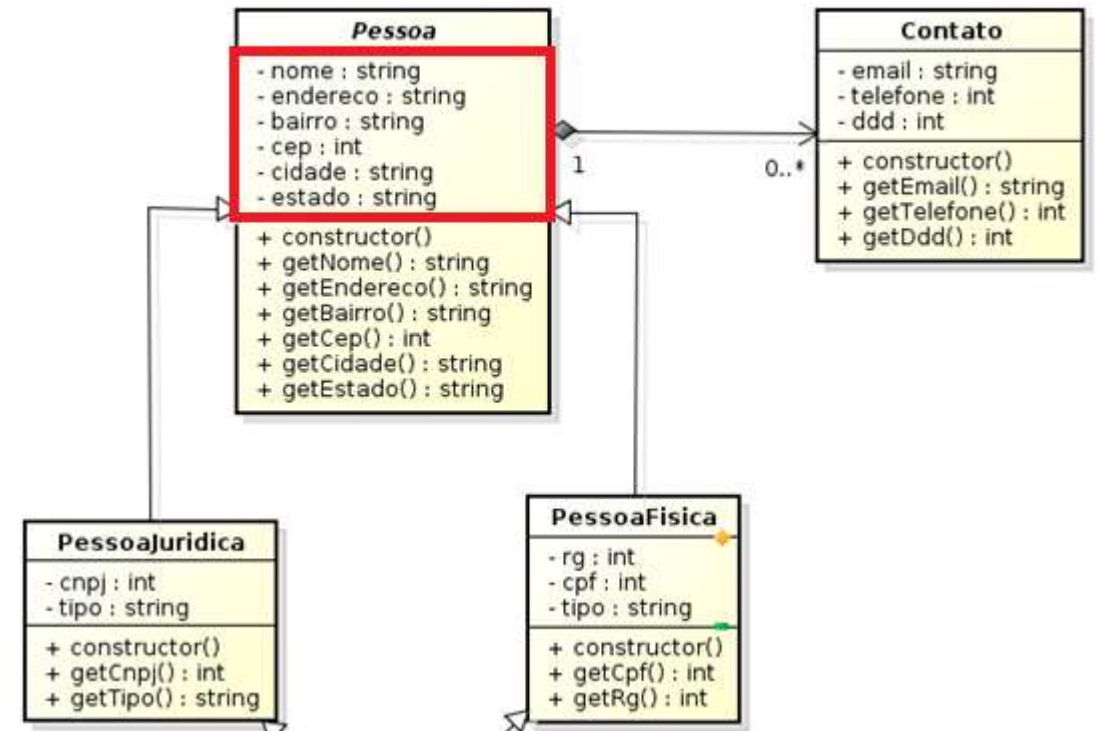
ESTRUTURA DE DE UMA CLASSE

- Relacionamentos
- Associação
- Agregação
- Composição
- Generalização
- Dependência



ATRIBUTOS

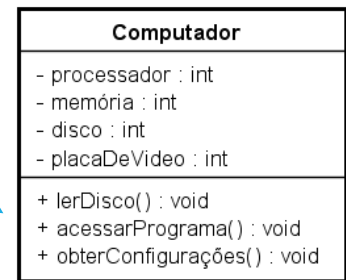
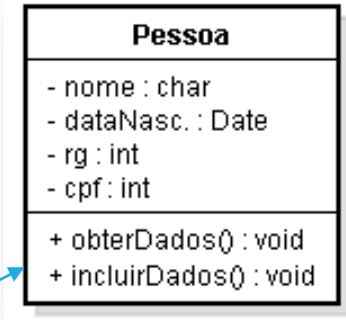
- Representam o conjunto de características (estado) dos objetos daquela classe
- Visibilidade:
 - + público: qualquer classe do sistema pode ver/manipular o atributo
 - # protegido: qualquer subclasse pode ver/manipular o atributo
 - - privado: somente a classe pode ver/ manipular o atributo
- Exemplo:
 - + nome : String



MÉTODOS

- Um método nada mais é que o equivalente a uma operação ou função, com a restrição que ele manipula apenas suas variáveis locais e os atributos que foram definidos para a classe.

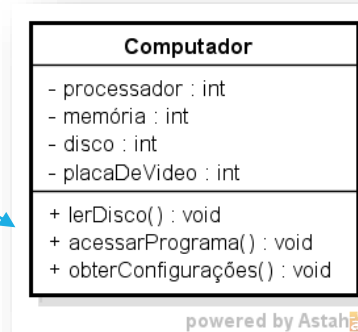
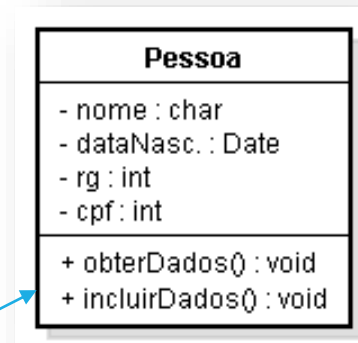
Métodos



MÉTODOS

- Representam o conjunto de operações (comportamento) que a classe fornece
- Visibilidade:
 - + público: visível para qualquer classe do sistema
 - # protegido: visível para qualquer subclasse
 - - privado: visível somente para classe
 - ~ pacote: visível somente para classes de um mesmo pacote
- Exemplo:
 - - getNome() : String

Métodos



Classe
Cachorro
Objetos da
classe
Cachorro

PINGO



Anda
Fala
Come
Dorme
PegaOsso

BIDU



Anda
Fala
Come
Dorme
PegaOsso

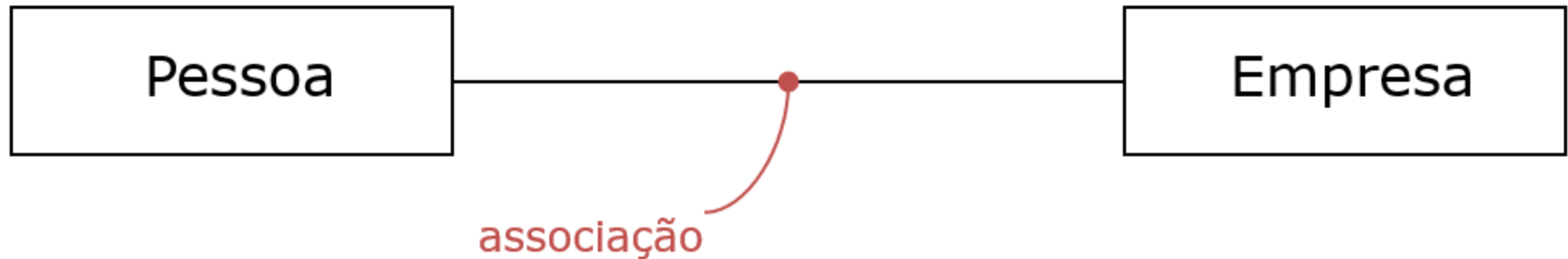


NINA

Anda
Fala
Come
Dorme
PegaOsso

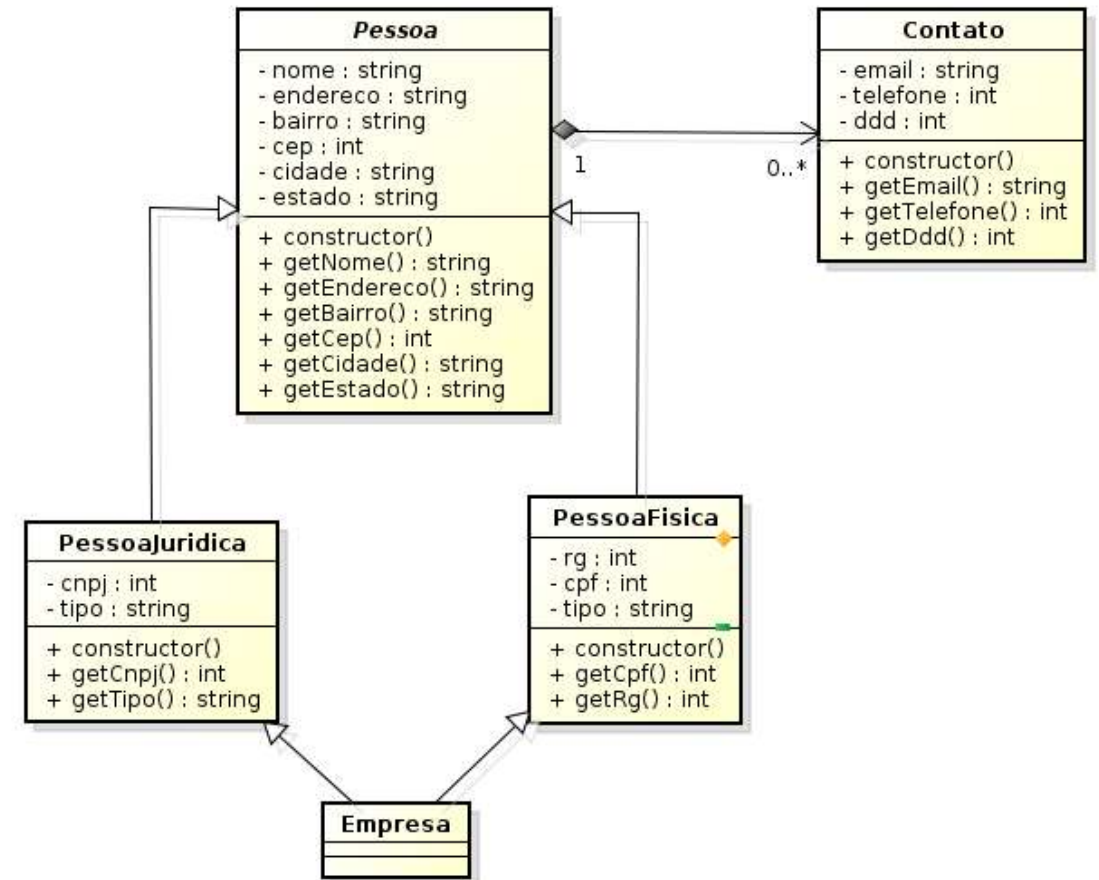
ASSOCIAÇÃO

- Uma associação é um relacionamento estrutural que indica que os objetos de uma classe estão vinculados a objetos de outra classe.
- Uma associação é representada por uma linha sólida conectando duas classes.

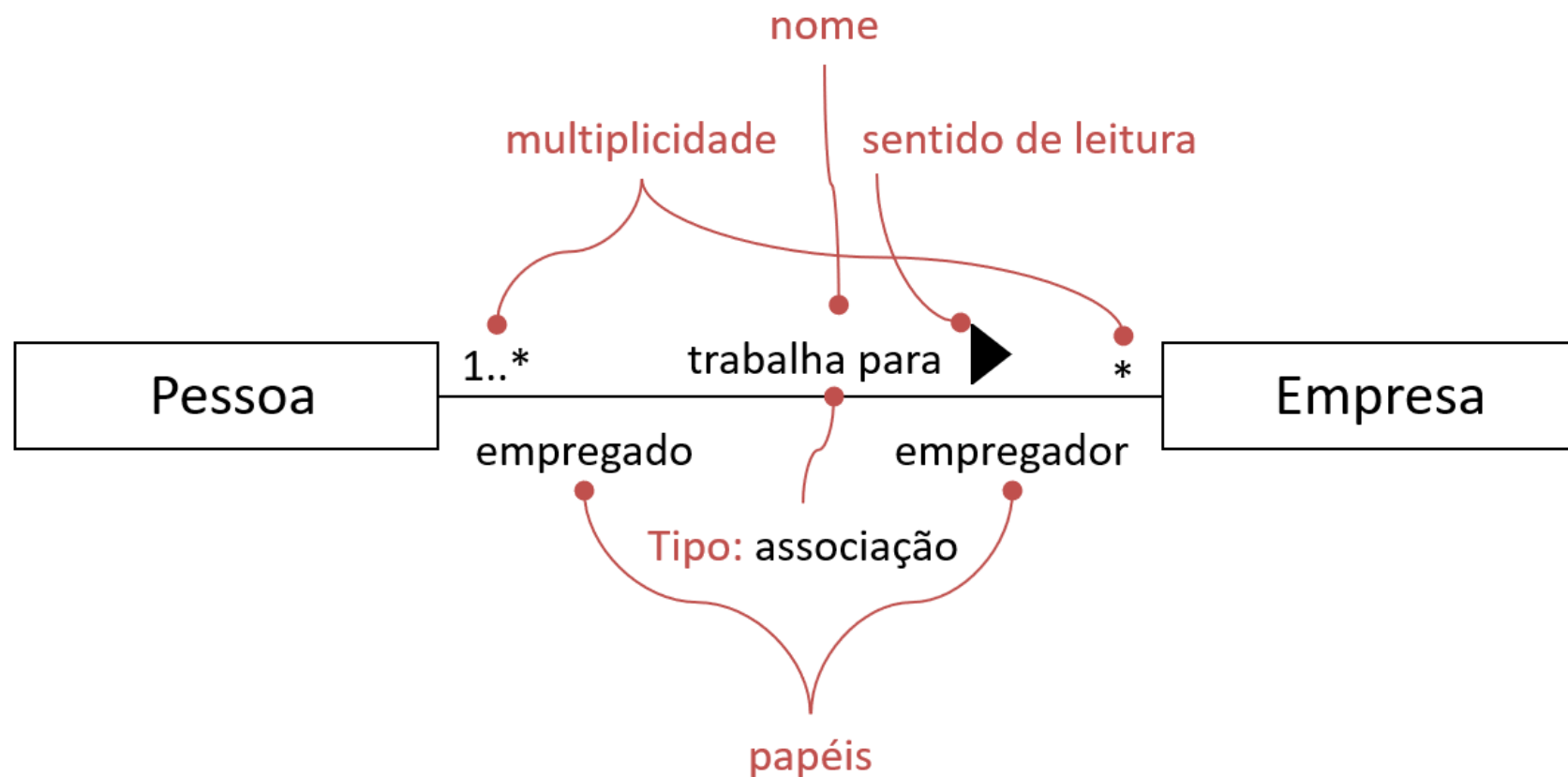


RELACIONAMENTOS

- Os relacionamentos possuem:
- Nome: descrição dada ao relacionamento (faz, tem, possui,...)
- Sentido de leitura
- Navegabilidade: indicada por uma seta no fim do relacionamento
- Multiplicidade: 0..1, 0..*, 1, 1..*, 2, 3..7
- Tipo: associação (agregação, composição), generalização e dependência
- Papéis: desempenhados por classes em um relacionamento



RELACIONAMENTOS



MULTIPLICIDADE

Multiplicidade	Significado
0..1	No mínimo zero e no máximo um. Os objetos não precisam estar relacionados, porém se houver relacionamento deve ser de no máximo 1.
1..1	Um e somente um
0..*	No mínimo nenhum e no máximo muitos.
*	Muitos
1..*	No mínimo um e no máximo muitos.
3..5	No mínimo 3 e no máximo 5.

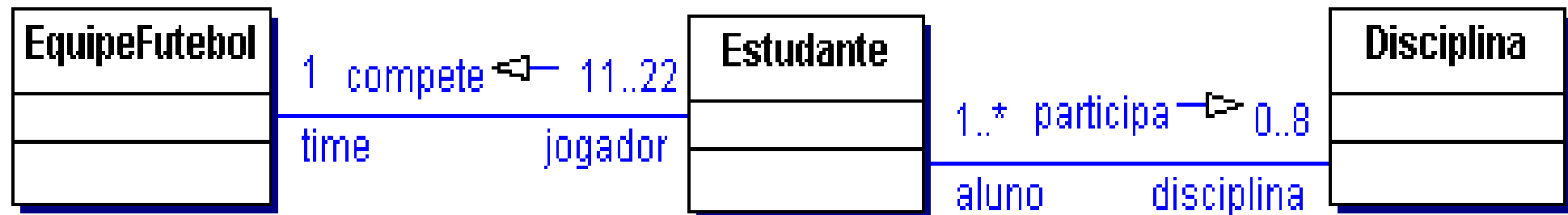
MULTIPLICIDADE

- Indicadores de multiplicidade:
- 1 Exatamente um
- 1..* Um ou mais
- 0..* Zero ou mais (muitos)
- * Zero ou mais (muitos)
- 0..1 Zero ou um
- m..n Faixa de valores (por exemplo: 4..7)



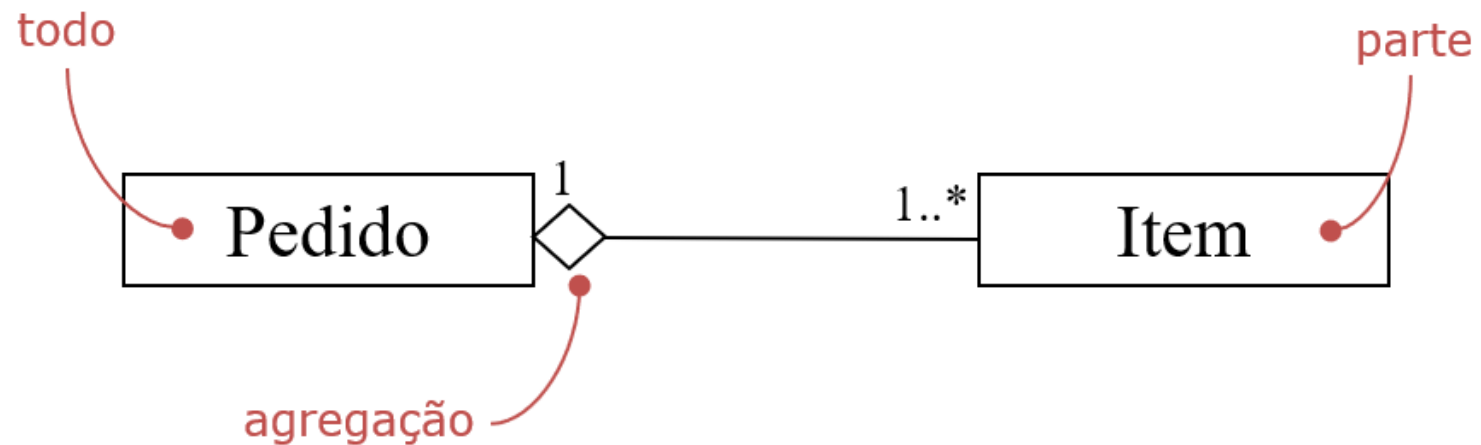
EXEMPLO

- Um Estudante pode ser um aluno de uma Disciplina e um jogador da Equipe de Futebol;
- Cada Disciplina deve ser cursada por no mínimo 1 aluno;
- Um aluno pode cursar de 0 até 8 disciplinas



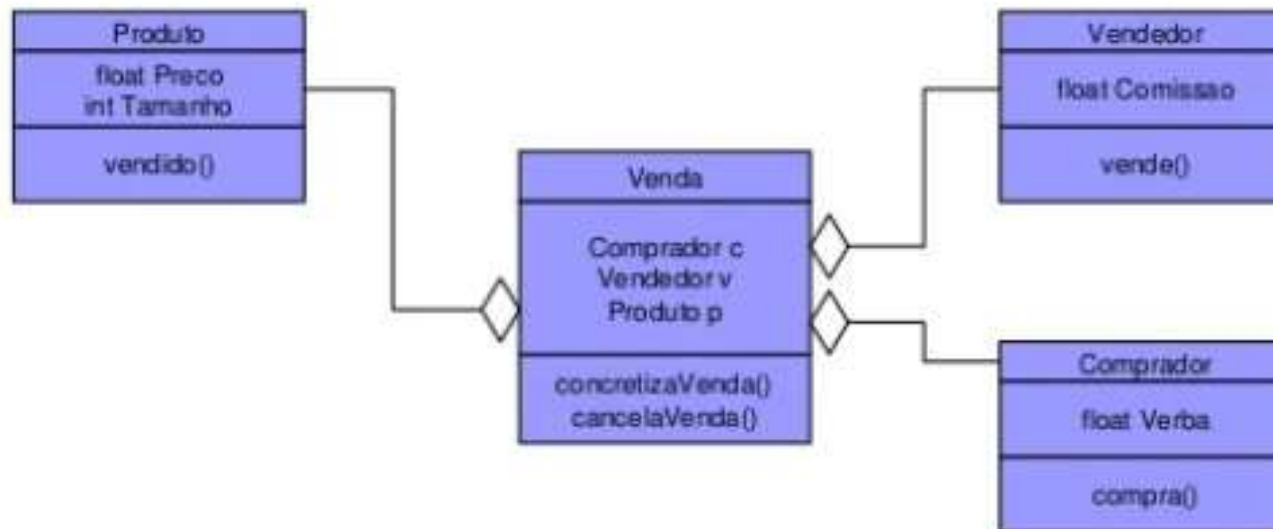
AGREGAÇÃO

- Um É um tipo especial de associação;
- Utilizada para indicar “todo-parte”



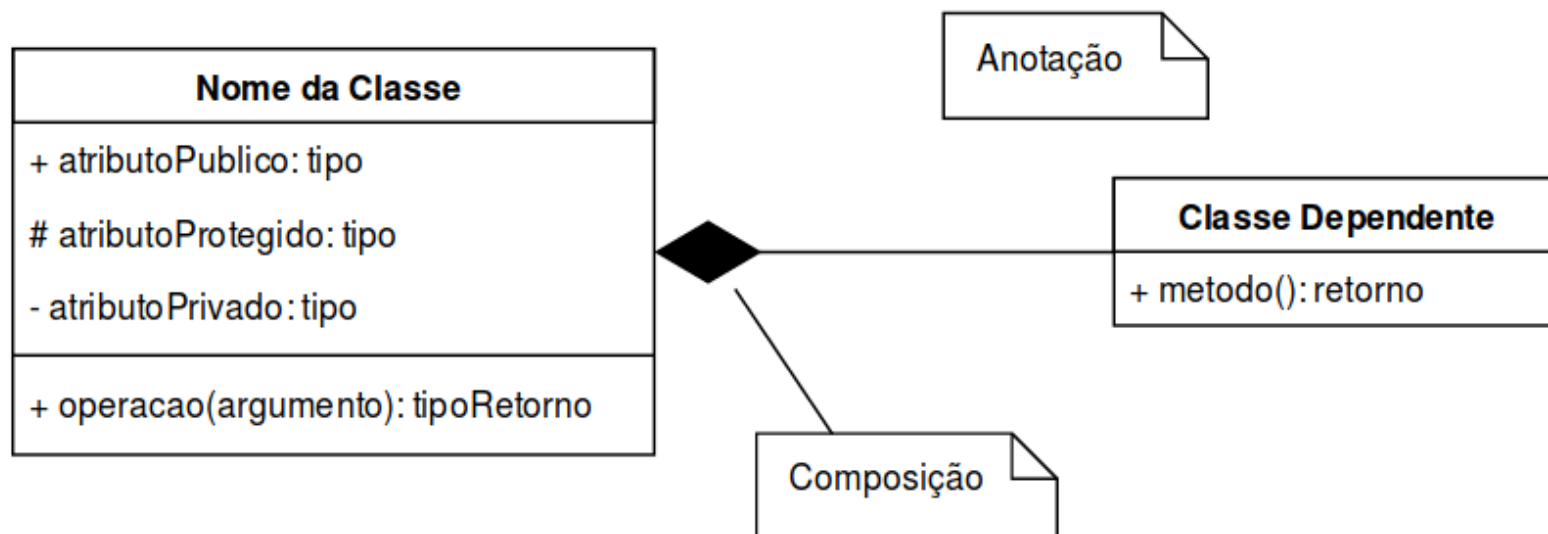
AGREGAÇÃO

- A venda é o objeto definido como sendo o todo. E este objeto somente pode existir caso os demais objetos que o compõem também existam.



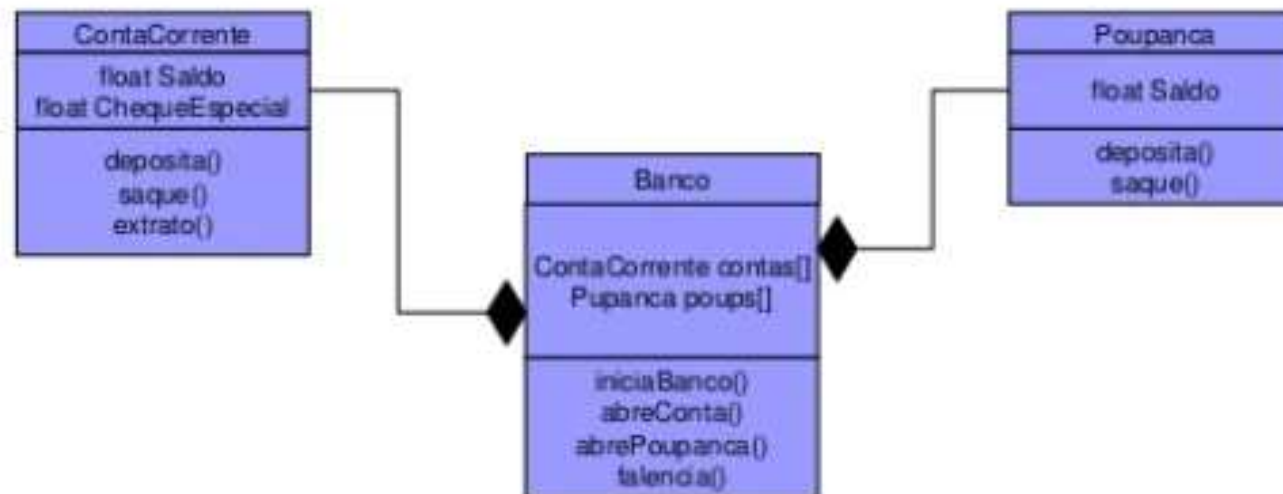
COMPOSIÇÃO

- É uma variante semanticamente mais “forte” da agregação
- Os objetos “parte” só podem pertencer a um único objeto “todo” e têm o seu tempo de vida coincidente com o dele

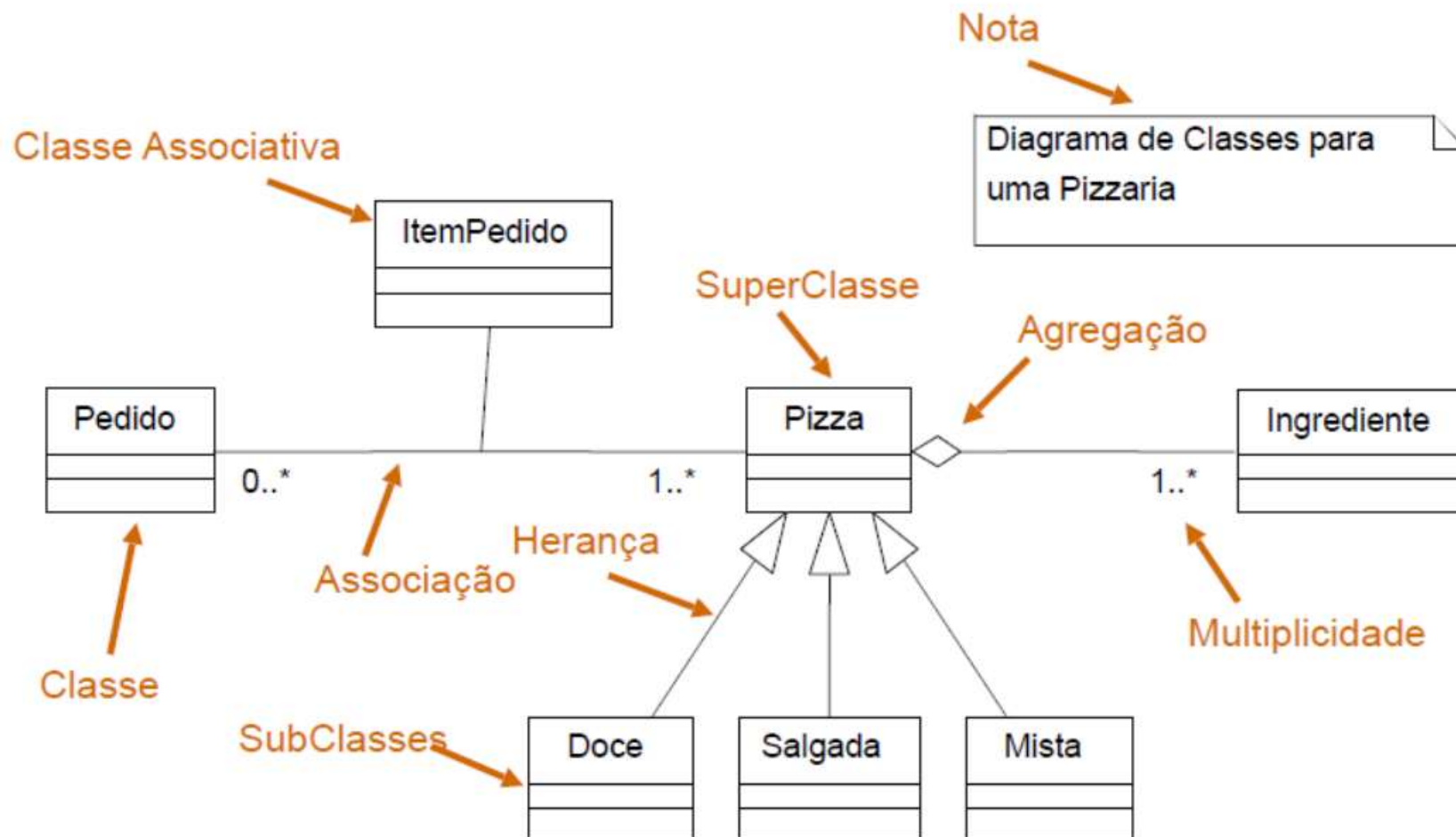


COMPOSIÇÃO

- No caso desta composição uma vez que o Objeto BANCO for destruído todos objetos **Poupança** e **Conta Corrente** deverão ser destruídos também.



ANÁLISE



ATIVIDADE - 1

- A Escola de Idiomas Phoenix deseja informatizar sua biblioteca. Pois deseja controlar os empréstimos dos livros para os alunos. Devem ser armazenados no sistema os dados dos livros, usuários, dos alunos, dos empréstimos, atendente e bibliotecária. OBS: Cada livro pertence a uma editora e deve ter no mínimo um autor. Defina as classes, seus principais atributos e relacionamentos.



ATIVIDADE - 2

- A escola de idiomas DeepLearning está com problemas durante as matrículas dos alunos no período das promoções, pois o cadastro é realizado com o preenchimento manual de formulários. Por este motivo deseja informatizar esse processo. Sabe-se que existem vários cursos em andamento e que cada curso possui várias turmas. Uma turma, por sua vez pode ter diversos estudantes matriculados, no entanto uma matrícula refere-se exclusivamente a uma determinada turma. Cada turma tem um número mínimo de matrículas para iniciar o curso.
- O dono da DeepLearning quer te contratar para analisar o problema e criar o diagrama de classes para este projeto. Qual seria a solução?



BIBLIOGRAFIA

MEDEIROS, Erani.
**Desenvolvendo
Software com UML.**
Editora Pearson,
2006.

