

# Diagrama de Classes

Disciplina: **Tópicos em Sistemas de Informação**

Prof. Me. Fernando Roberto Proença

## Diagrama de Classes

2

- Diagrama mais utilizado da UML
- Composto por
  - ▣ Classes – Cada classe com seus atributos e métodos
  - ▣ Associações – Relacionamento entre as classes
- Serve de apoio para a maioria dos outros diagramas
- “Ideia” do diagrama Entidade-Relacionamento(DER)
- Reflete a estrutura do código
  - ▣ O diagrama de classes é o diagrama mais próximo da implementação.

## Diagrama de Classes – Finalidade

3

- Modelar os elementos de um programa orientado a objetos em tempo de desenvolvimento
  - ▣ Classes com seus atributos e métodos
- Modelar os relacionamentos entre classes, de forma mais explícita que aquela do código
  - ▣ Associação
  - ▣ Agregação (e composição)
  - ▣ Herança
  - ▣ Dependência

## Notação de uma Classe

4

- Representada através de uma “caixa” com no máximo três divisões.
- Notação utilizada depende do nível de abstração desejado.

Nome da Classe
----------------

Nome da Classe
lista de atributos

Nome da Classe
lista de operações

Nome da Classe
lista de atributos
lista de operações

## Notação de uma Classe

5

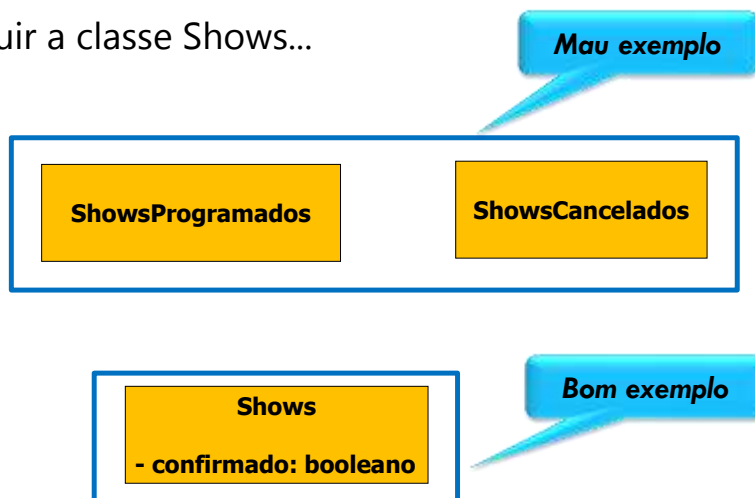
- Exemplo (classe ContaBancária)



## Bons e maus exemplos de Classe

6

- Construir a classe Shows...



## Classes – Atributos

7

- São as características / propriedades da classe.
- Na UML, atributos são mostrados com pelo menos seu nome
- Podem também mostrar seu tipo, valor inicial e outras propriedades.

## Classes – Métodos

8

- São comportamentos (operações) de uma classe.
- Métodos são exibidos com pelo menos seu nome
- Podem também mostrar seus parâmetros e valores de retorno.

## Classes – Visibilidade

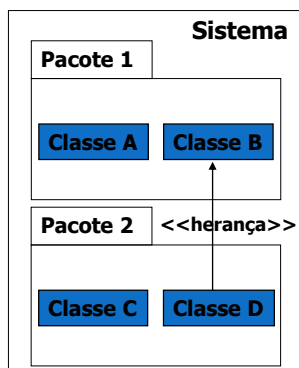
9

- Public (+) – Público
- Protected (#) – Protegido
- Private (-) – Privado
- Package (~) – Pacote

## Classes – Visibilidade

10

- Public (+)
  - Visível para qualquer elemento que possa ver a classe.



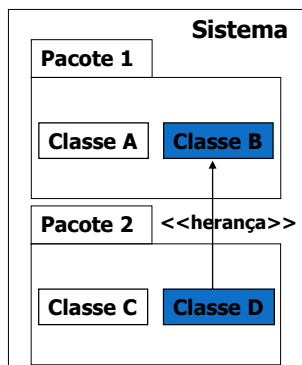
### Atributos e Métodos Públicos:

- Podem ser acessados por qualquer outra classe do sistema.

## Classes – Visibilidade

11

- Protected (#)
  - ▣ Visível a outros elementos dentro da classe e de subclasses (herança).



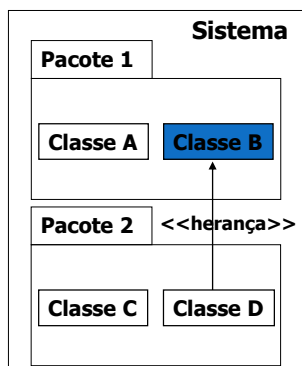
### Atributos e Métodos Protegidos:

- Só podem ser acessados por uma classe que tenha uma relação de generalização / especialização, independente do pacote.

## Classes – Visibilidade

12

- Private (-)
  - ▣ Visível a outros elementos que estão dentro da classe.



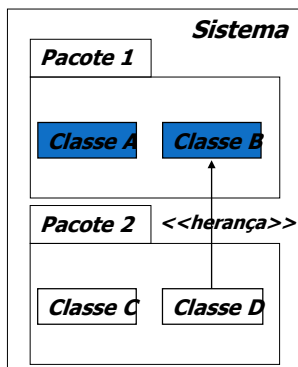
### Atributos e Métodos Privados:

- Só são acessíveis dentro do mesmo objeto.
- Normalmente este é o estado inicial dos atributos, depois se necessário, pode ser modificado.

## Classes – Visibilidade

13

- Package (~)
  - ▣ Visível a elementos do mesmo pacote.



### Atributos e Métodos Privados:

- Só são acessíveis dentro do mesmo objeto.
- Normalmente este é o estado inicial dos atributos, depois se necessário, pode ser modificado.

## Recomendação para estabelecimento de visibilidade

14

- Atributos → privados ou protegidos
  - ▣ Princípio da ocultação de informação do paradigma de orientação a objetos
  - ▣ Possibilita que atributos herdados ou definidos na classe sejam tratados de maneira uniforme.
- Atributos públicos → **JAMAIS**

## Recomendação para estabelecimento de visibilidade

15

- Métodos → públicos
  - ▣ O meio (objeto) externo acessa uma classe através de seus métodos.
- Necessidades específicas podem justificar o aumento de restrição de visibilidade.

## Diagrama de Classes - Relacionamentos

16

- Relacionamento entre classes
  - ▣ É a maneira na qual as classes conectam entre si com o intuito de compartilhar informações e colaborarem umas com as outras para permitir a execução de processos.
- Em OO há três tipos de relacionamentos:
  - ▣ Especializações/Generalizações (Heranças)
  - ▣ Associações
  - ▣ Dependências



## Diagrama de Classes - Relacionamentos

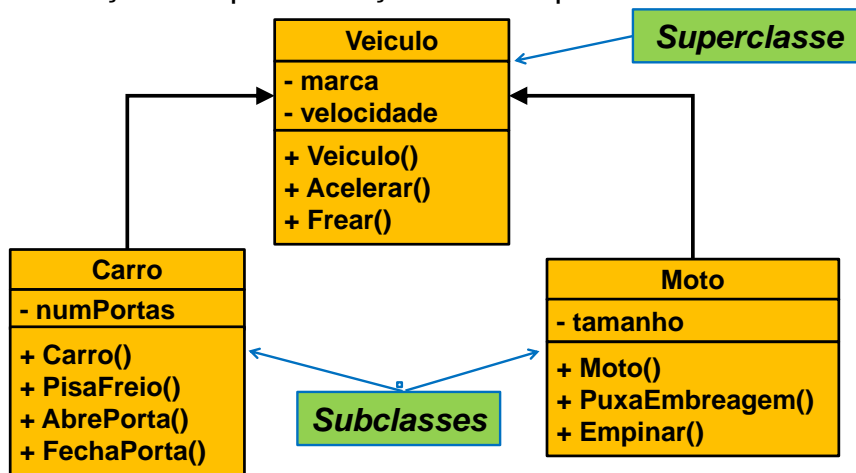
17

- Generalização / Especialização
  - Relacionamento **entre classes mais gerais** a outras **classes mais específicas** onde o elemento mais específico **herda** as propriedades e métodos do elemento mais geral.
  - Conhecido como relacionamento de **superclasse/subclasse**
    - **Subclasse** (classe derivada ou filha): consiste na classe que herda todos os métodos e atributos de uma classe existente;
    - **Superclasse** (classe base ou classe mãe): é a classe existente que é herdada por uma outra classe.
  - É implementada como **herança** em Programação OO.

## Diagrama de Classes - Relacionamentos

18

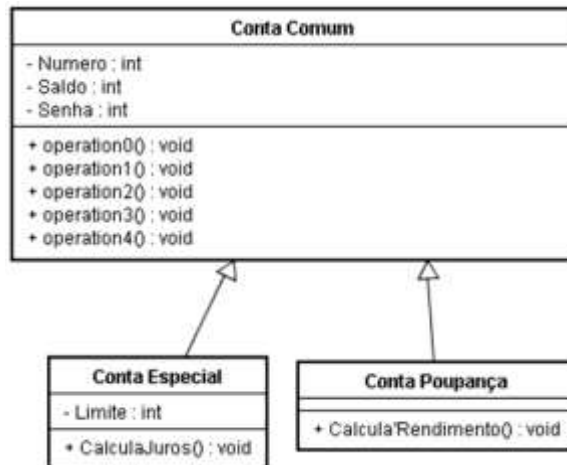
- Generalização / Especialização – Exemplo



## Diagrama de Classes - Relacionamentos

19

### Generalização / Especialização – Exemplo



## Diagrama de Classes - Relacionamentos

20

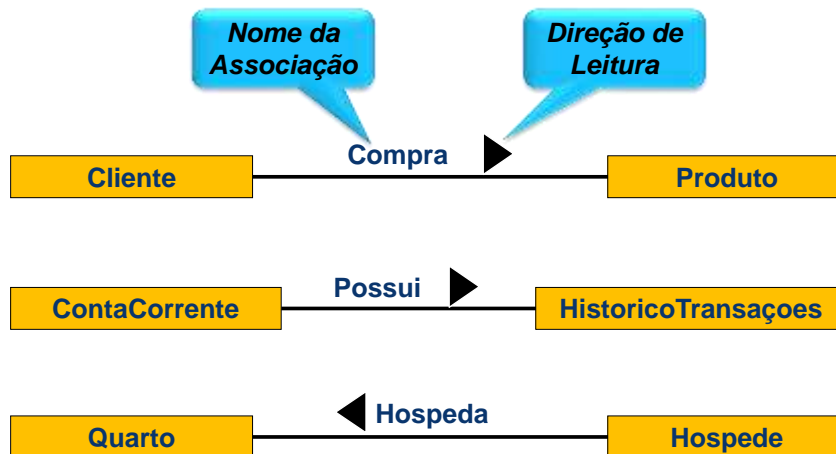
### Associação

- É uma conexão entre classes.
- Relacionamentos estruturais entre instâncias e especificam que objetos de uma classe estão ligados a objetos de outras classes, podendo haver troca de informações e compartilhamento de métodos.
- Ocorre normalmente entre duas classes (binária), entre uma classe com ela mesma (unária) e entre várias classes (ternária/N-ária).
- "Equivale" aos relacionamentos E-R.

## Diagrama de Classes - Relacionamentos

21

### □ Associação – Exemplos



## Diagrama de Classes - Relacionamentos

22

### □ Associação

■ **Participação:** consiste no **papel** (pode ser uma função ou cargo) de uma determinada classe em um relacionamento com outra classe.

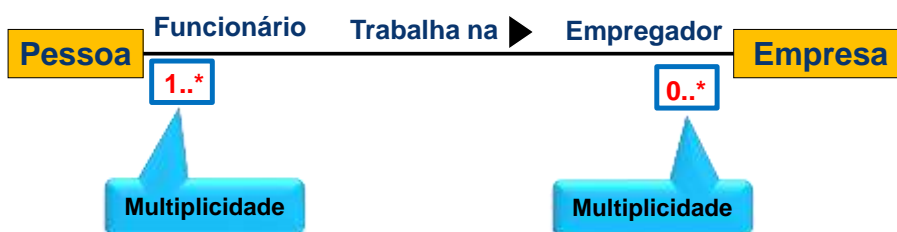
■ Exemplo:



## Diagrama de Classes - Relacionamentos

23

- Associação
  - ▣ **Multiplicidade:** é a quantidade mínima e máxima de objetos que podem ser conectados pela instancia de uma associação.
  - ▣ Exemplo:



## Diagrama de Classes – Multiplicidade

24

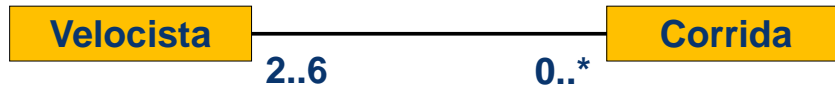
- Simbologia e Significado

Nome	Simbologia
Apenas um	1
Zero ou muitos	0..*
Um ou muitos	1..*
Zero, um ou muitos	*
Zero ou um	0..1
Intervalo específico	ex.: 2..8

## Diagrama de Classes – Multiplicidade

25

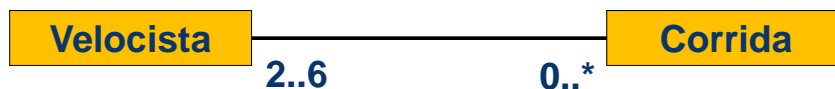
- Exemplos de Multiplicidades



## Diagrama de Classes – Multiplicidade

26

- Exemplos de Multiplicidades

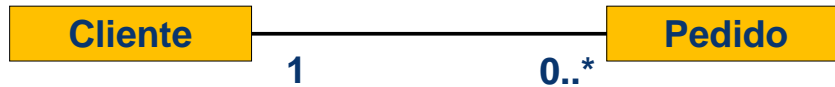


- Uma **corrida** está associada a, no mínimo, **dois velocistas**
- Uma **corrida** está associada a, no máximo, **seis velocistas**.
- Um **velocista** pode estar associado a **nenhuma corrida**.
- Um **velocista** pode estar associado a **várias corridas**.

## Diagrama de Classes – Multiplicidade

27

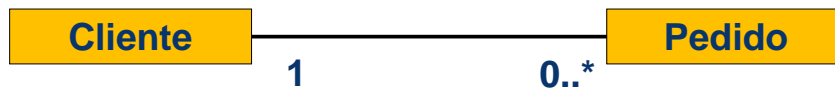
- Exemplos de Multiplicidades



## Diagrama de Classes – Multiplicidade

28

- Exemplos de Multiplicidades



- Um **cliente** pode estar associado a **nenhum pedido**.
- Um **cliente** pode estar associado a **vários pedidos**.
- Um **pedido** está associado a **um, e somente um, cliente**.

## Diagrama de Classes – Conectividade

29

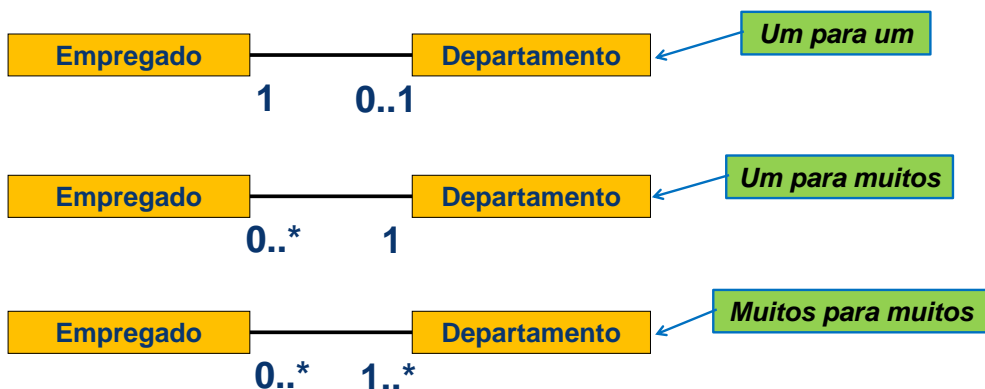
- Corresponde ao tipo de associação entre classes:
  - ▣ “muitos para muitos”, “um para muitos” e “um para um”.

Conectividade	Em um Extremo	No outro Extremo
Um para um	0..1 1	0..1 1
Um para muitos	0..1 1	* 1..* 0..*
Muitos para muitos	* 1..* 0..*	* 1..* 0..*

## Diagrama de Classes – Conectividade

30

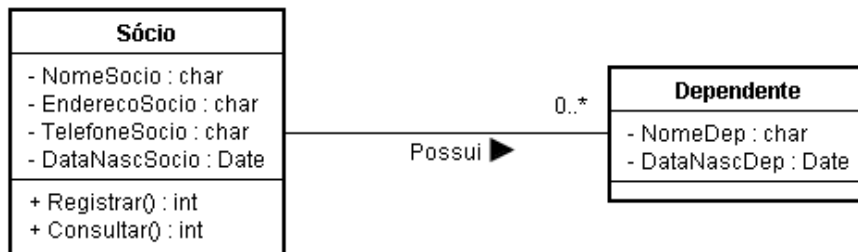
- Exemplos de Conectividades



## Diagrama de Classes - Relacionamentos

31

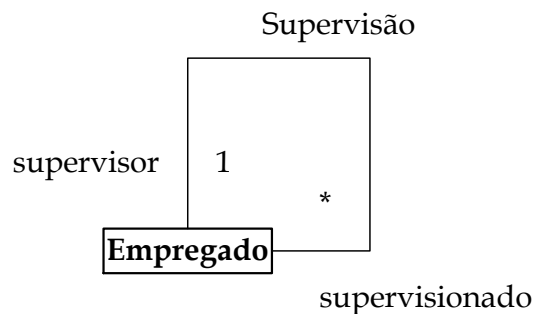
- Associação Binária
  - ▣ Associações entre duas classes
  - ▣ Mais comum
  - ▣ Exemplo:



## Diagrama de Classes - Relacionamentos

32

- Associação Unária (ou Reflexiva)
  - ▣ Ocorre quando uma classe relaciona com si mesmo.
  - ▣ Exemplo:





## Diagrama de Classes - Relacionamentos

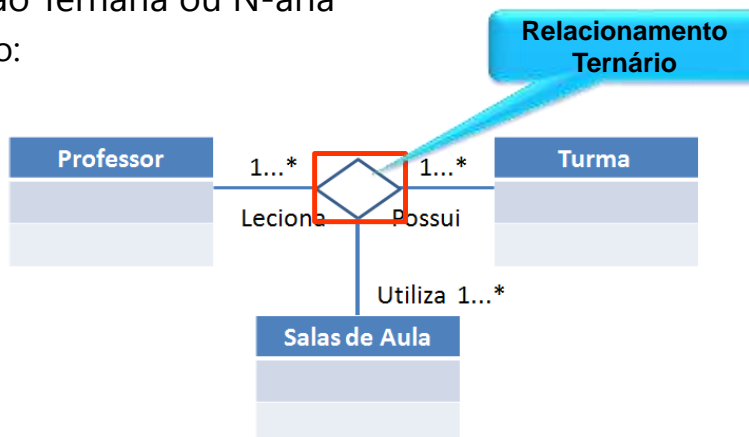
33

- Associação Ternária ou N-ária
  - ▣ Associações que conectam mais de duas classes
  - ▣ Úteis para demonstrar associações complexas
  - ▣ Devem ser evitadas – difíceis de interpretar
  - ▣ São representadas por um **losango** para onde convergem todas as ligações de associação

## Diagrama de Classes - Relacionamentos

34

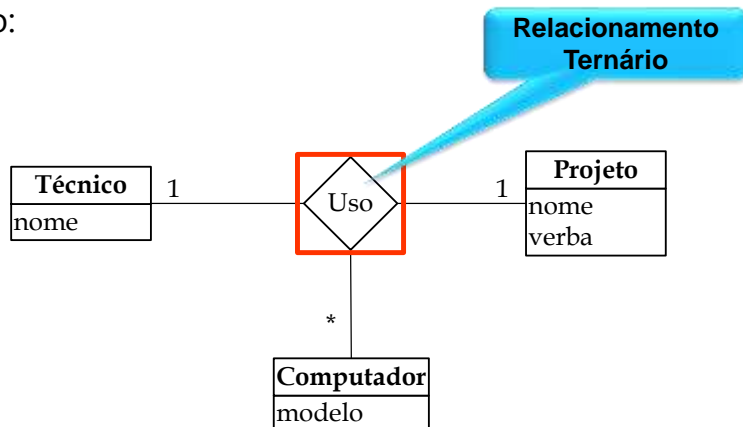
- Associação Ternária ou N-ária
  - ▣ Exemplo:



## Diagrama de Classes - Relacionamentos

35

- Associação Ternária ou N-ária
  - ▣ Exemplo:



## Diagrama de Classes - Relacionamentos

36

- Agregação
  - ▣ Qual o significado da palavra Agregação?
  - ▣ Reunião de partes homogêneas formando um todo;
  - ▣ Sinônimos: junção, acumulação, coesão, aglomeração, anexo, conjunto...
  - ▣ Ex.: Em física: Porção de moléculas agrupadas.

## Diagrama de Classes - Relacionamentos

37

- Agregação
  - É um tipo especial de associação onde tenta-se demonstrar que as informações de um objeto (objeto-todo) precisam ser complementadas pelas informações contidas em um ou mais objetos de outra classe (objeto-parte)
  - Em uma agregação, um objeto **está contido** no outro
  - Temos a Relação Todo-Parte

## Diagrama de Classes - Relacionamentos

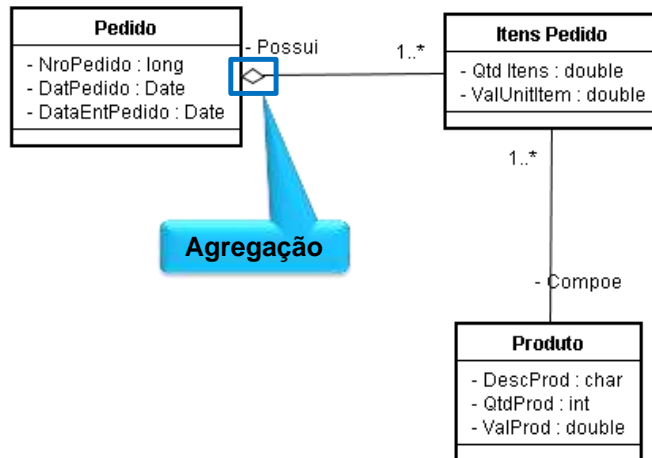
38

- Quando utilizar a Agregação?
  - Deseja-se modelar um relacionamento "todo/parte", em que uma classe representa uma entidade completa (todo), composta de outras entidades (partes).
- Notação da Agregação:
  - Representada como uma linha conectando as classes relacionadas, com um diamante **(losango) branco** perto da classe que representa o todo.

## Diagrama de Classes - Relacionamentos

39

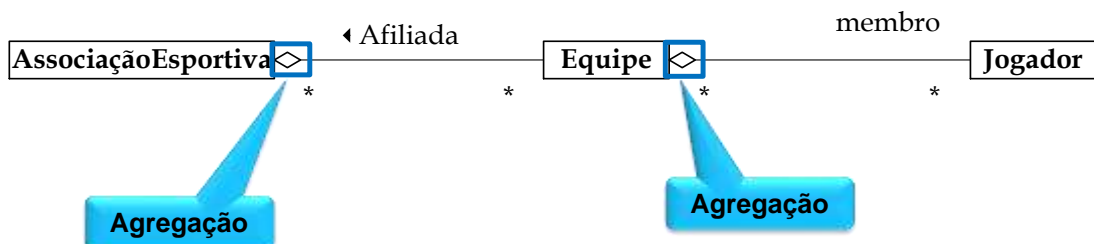
### □ Agregação – Exemplo



## Diagrama de Classes - Relacionamentos

40

### □ Agregação – Exemplo



## Diagrama de Classes - Relacionamentos

41

- Composição
  - Qual o significado da palavra Composição?
  - O que constitui ou forma algo. Ação de compor um todo juntando as partes.
  - Sinônimos: Arranjo, disposição, associação, combinação, constituição, organização, estrutura...
  - Ex.: Composição do sangue, da orquestra, de uma palavra; Exercício de redação escolar.

## Diagrama de Classes - Relacionamentos

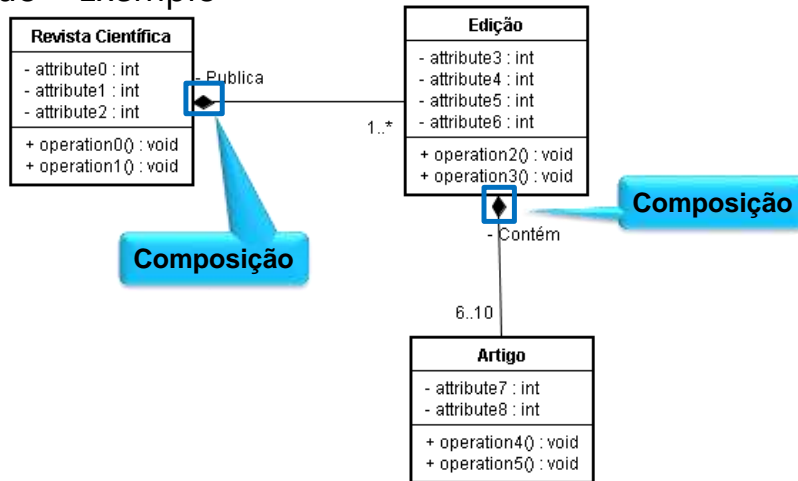
42

- Composição
  - É uma forma de agregação onde há
    - Vínculo **mais forte** entre Objetos-Todo e Objetos-Parte
    - Objetos-Parte têm de pertencer **exclusivamente** a um Objeto-Todo
    - As partes **não podem existir** sem o todo
  - Ou seja, o relacionamento entre um elemento (o todo) e outros elementos (as partes), onde as parte só podem pertencer ao todo e são criadas e destruídas com ele.
  - Representada como uma linha conectando as classes relacionadas, com um diamante (**losango**) pintado de **preto** perto da classe que representa o todo.

## Diagrama de Classes - Relacionamentos

43

### □ Composição – Exemplo



## Diagrama de Classes - Relacionamentos

44

### □ Associação com propriedades

- Em uma associação entre classes, a própria associação pode ter propriedades.
- O conjunto destas propriedades formam uma **Classe Associativa**.

## Diagrama de Classes - Relacionamentos

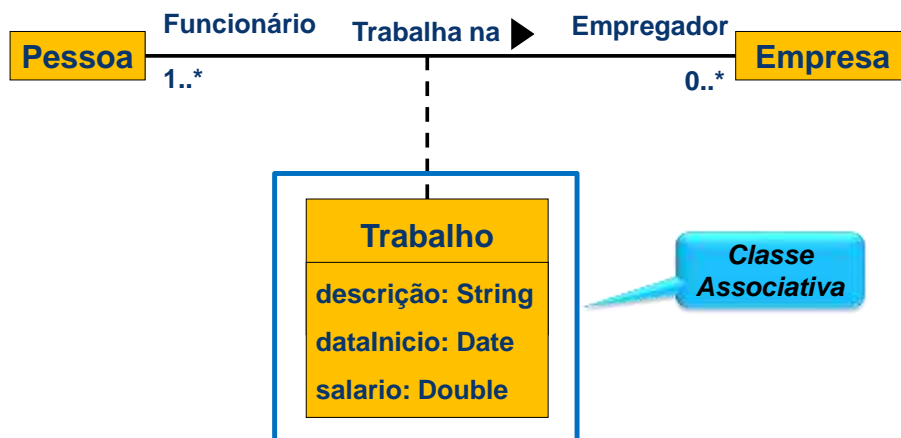
45

- Associação com propriedades
  - ▣ Em uma associação entre classes, a própria associação pode ter propriedades.
  - ▣ O conjunto destas propriedades formam uma **Classe Associativa**.
- Classe Associativa
  - ▣ Produzida quando da ocorrência de associações que possuem multiplicidade muitos (\*) em todas as suas extremidades.
  - ▣ Classe para armazenar os atributos transmitidos pela associação.

## Diagrama de Classes - Relacionamentos

46

- Classe Associativa – Exemplo



## Diagrama de Classes - Relacionamentos

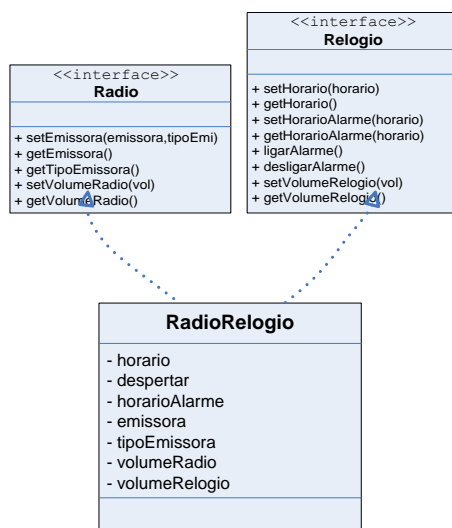
47

- Interfaces
  - **Interfaces** são classes abstratas que **contêm apenas métodos públicos abstratos**.
  - Uma **Interface** é formada por declarações de métodos desprovidos de implementação.
    - Ou seja, **apenas assinatura** e **sem o corpo** dos métodos.

## Diagrama de Classes - Relacionamentos

48

- Interfaces – notação

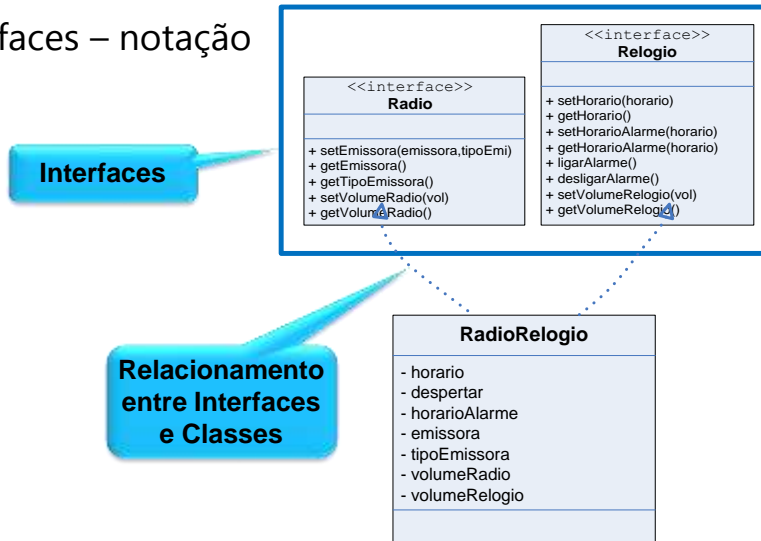




## Diagrama de Classes - Relacionamentos

49

### □ Interfaces – notação



## Diagrama de Classes - Relacionamentos

50

### □ Dependência

- Demonstra certo grau de dependência de uma classe a outra
- A dependência entre classes indica que os objetos de uma classe **usam operações, atributos, variáveis** ou **argumentos** dos objetos de outra classe.
- Mudança numa classe deverá refletir na outra.

## Dúvidas?

51

**Prof. Me. Fernando Roberto Proença**

***[fernando.proenca@uemg.br](mailto:fernando.proenca@uemg.br)***

