

MODELAGEM DE SOFTWARE - APOO



Junio Figueiredo

Professor

junioinf@gmail.com

Aula 03

Diagrama de Classes e
Associação

MODELAGEM DE SOFTWARE - APOO



Associações

Engenharia de Software - APOO

Cada ocorrência dos meus conceitos recebe o nome de INSTÂNCIA ou OBJETO

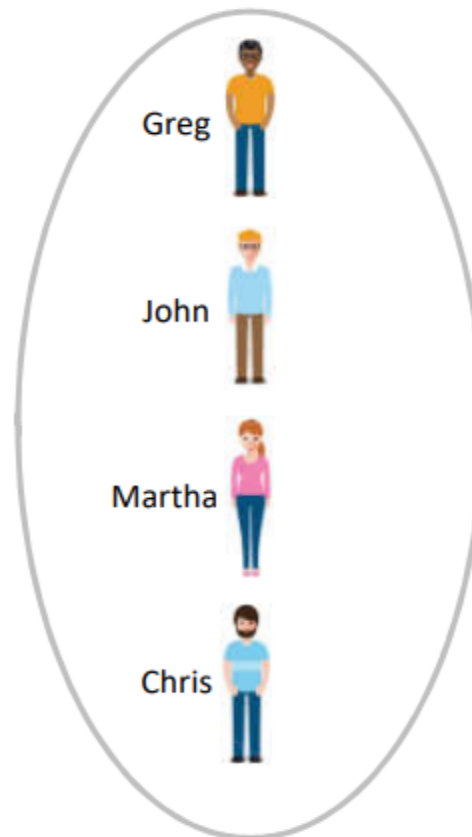
Instâncias

Exemplo:

Desejo criar um sistema para armazenar informações de pessoas e carros.

Conceitos:

- Pessoa
- Carro



Engenharia de Software - APOO



O que são associações

Associação é um relacionamento estático entre dois conceitos.

Exemplo:

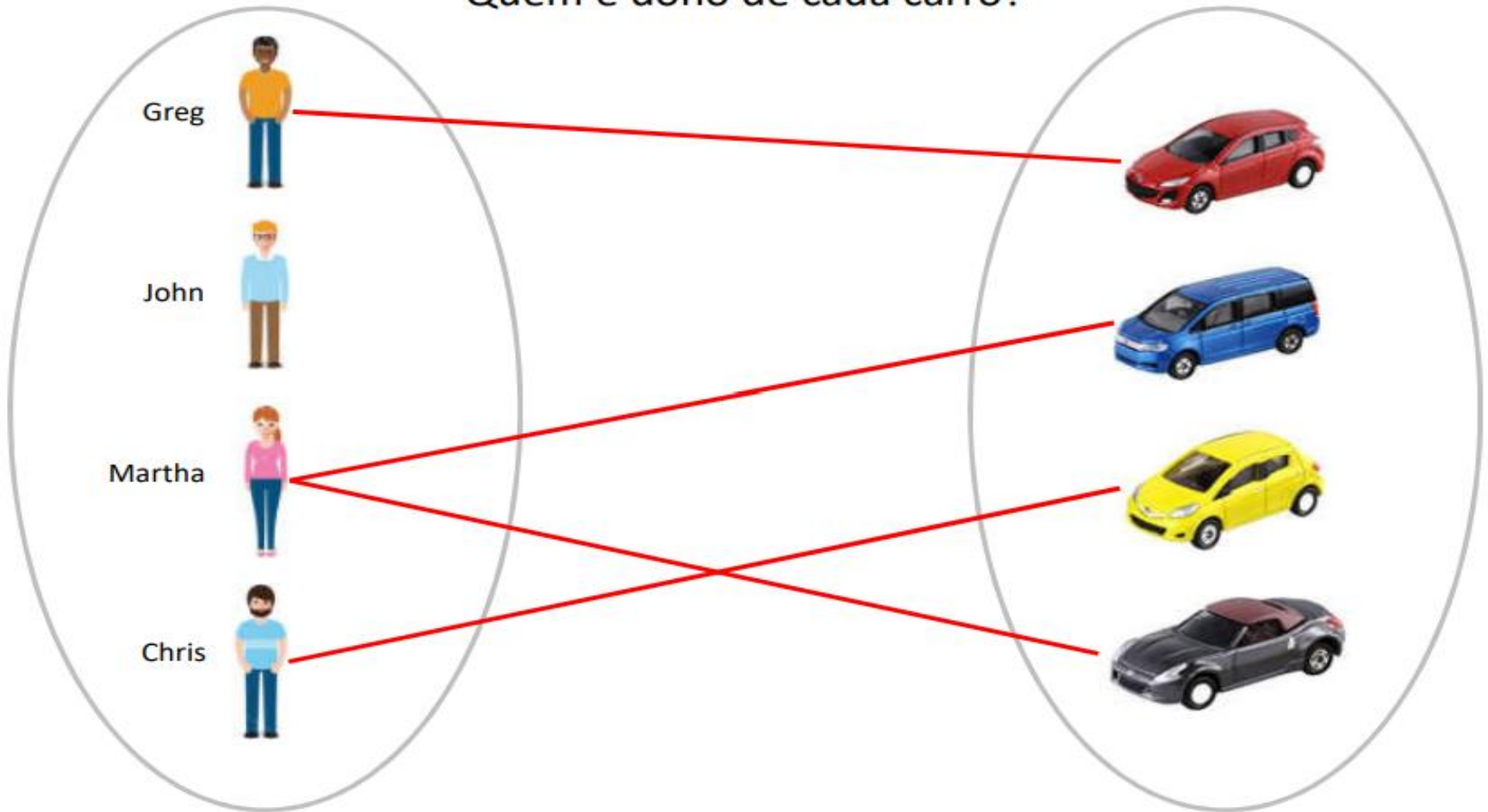
Desejo criar um sistema para armazenar informações de pessoas e carros.

Mas eu não tenho simplesmente a necessidade de saber quais são as pessoas e quais são os carros:

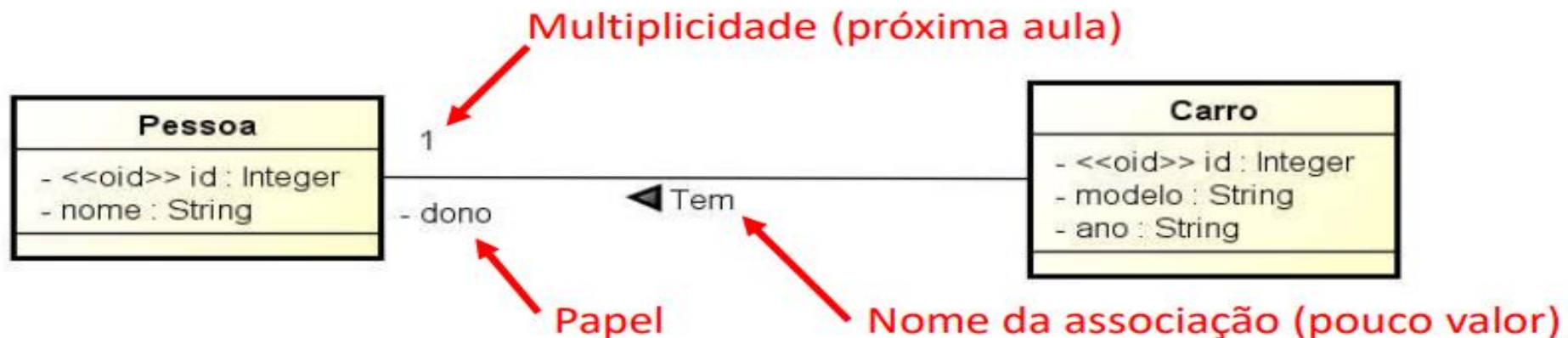
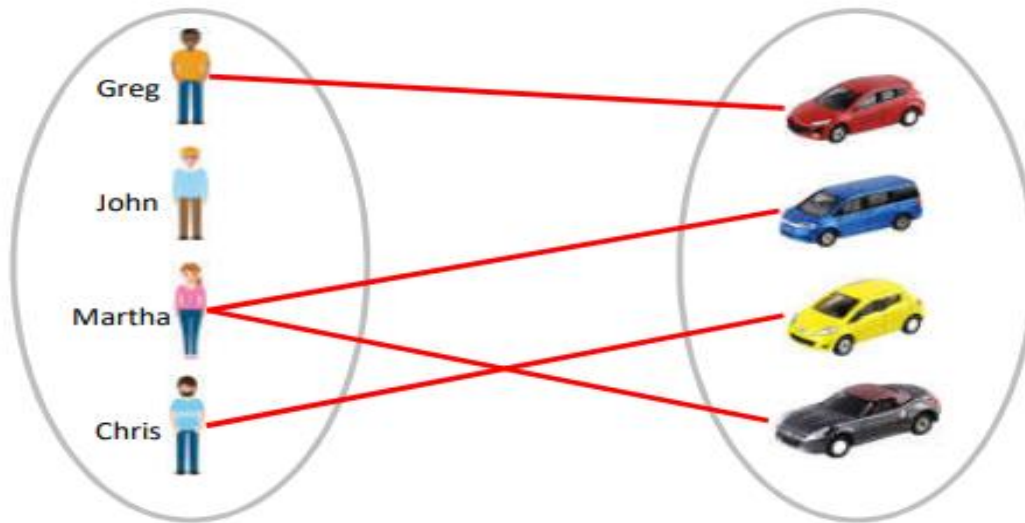
Também desejo saber quem é o dono de cada carro!

Engenharia de Software - AP00

Quem é dono de cada carro?

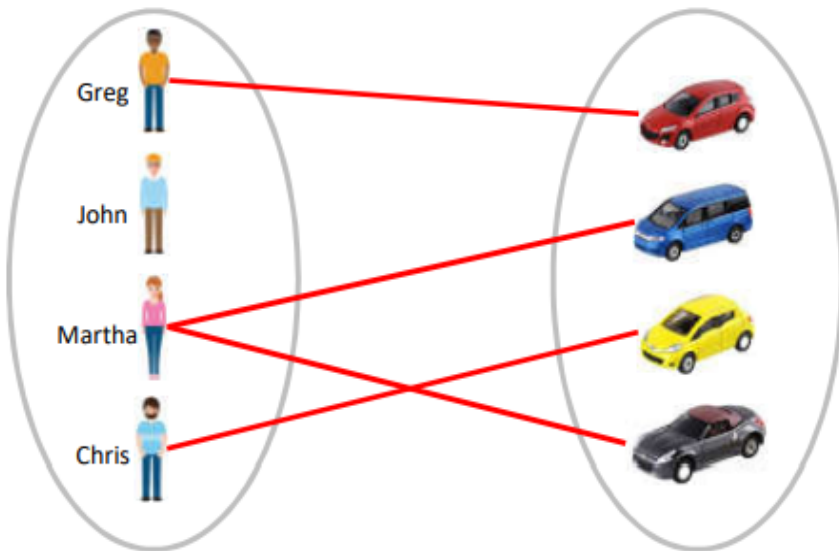


Engenharia de Software - AP00



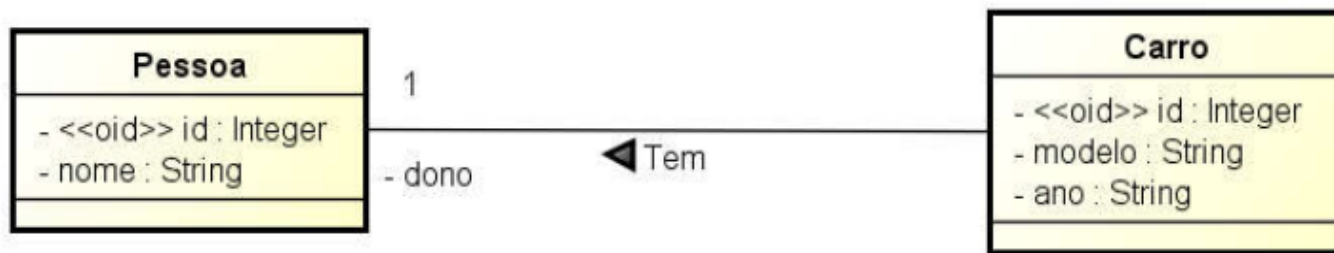
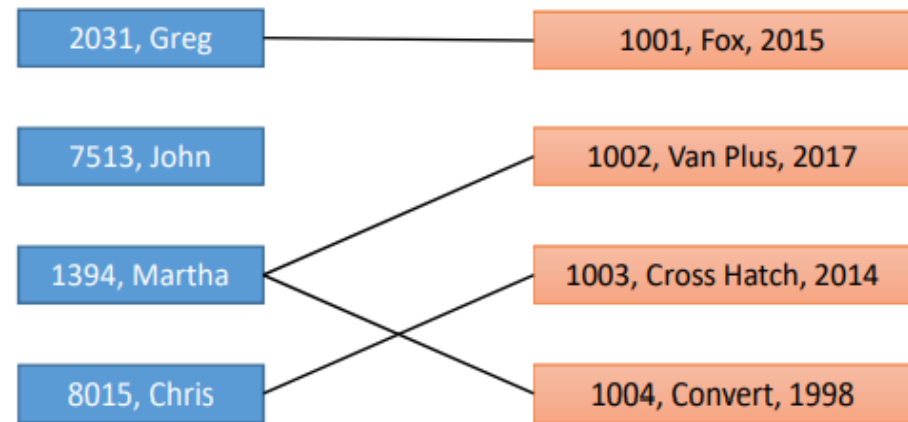
Engenharia de Software - APOO

Mundo real



Objetos

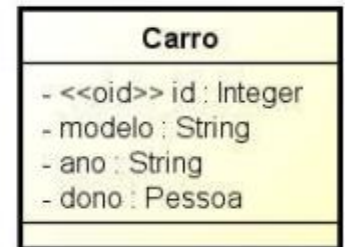
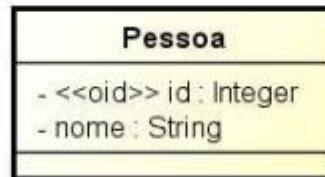
Como os objetos são representados dentro do SI



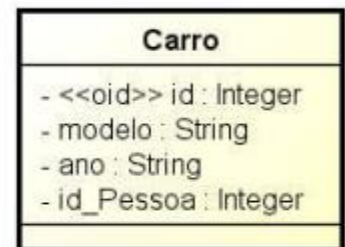
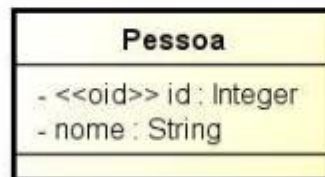
Engenharia de Software - APOO

Atenção!

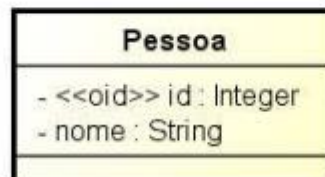
ERRADO



MUITO ERRADO

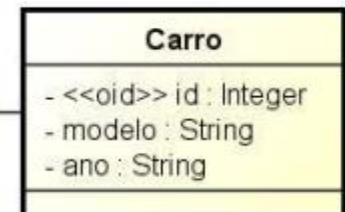


CORRETO



1

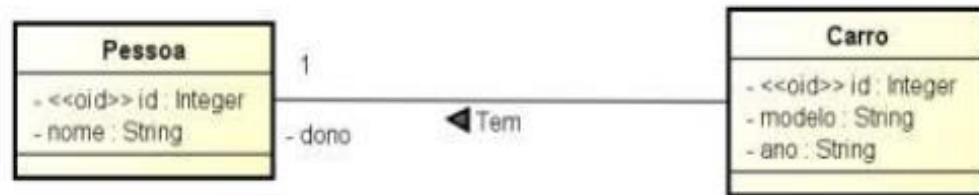
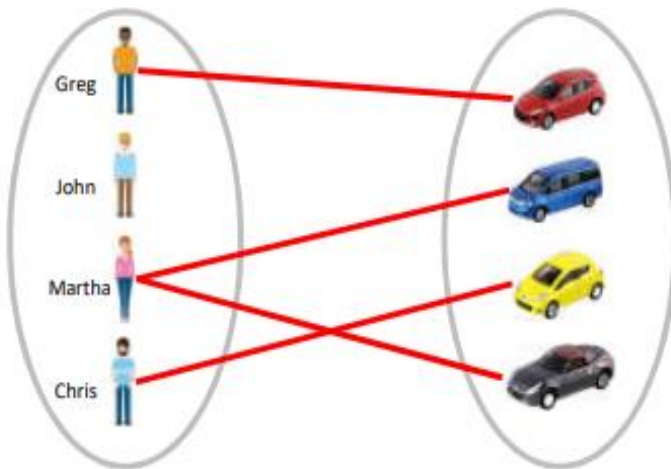
- dono



Engenharia de Software - APOO

Resumo da aula

- Associação é um relacionamento estático entre dois conceitos



- Nome da associação (pouco valor)
- Nome de papel
- Multiplicidade (próxima aula)

- Não confunda com Modelo Relacional

Engenharia de Software - APOO



Multiplicidade

Engenharia de Software - APOO

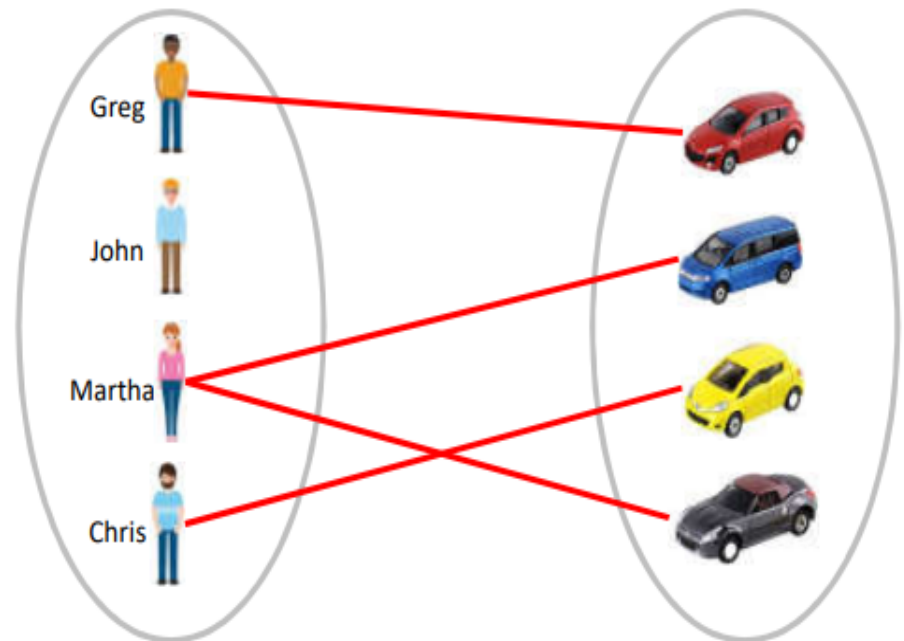
O que é multiplicidade?

É a quantidade mínima e máxima de objetos que uma associação permite em cada um de seus papéis.

Exemplo: um carro pode ter quantos donos?

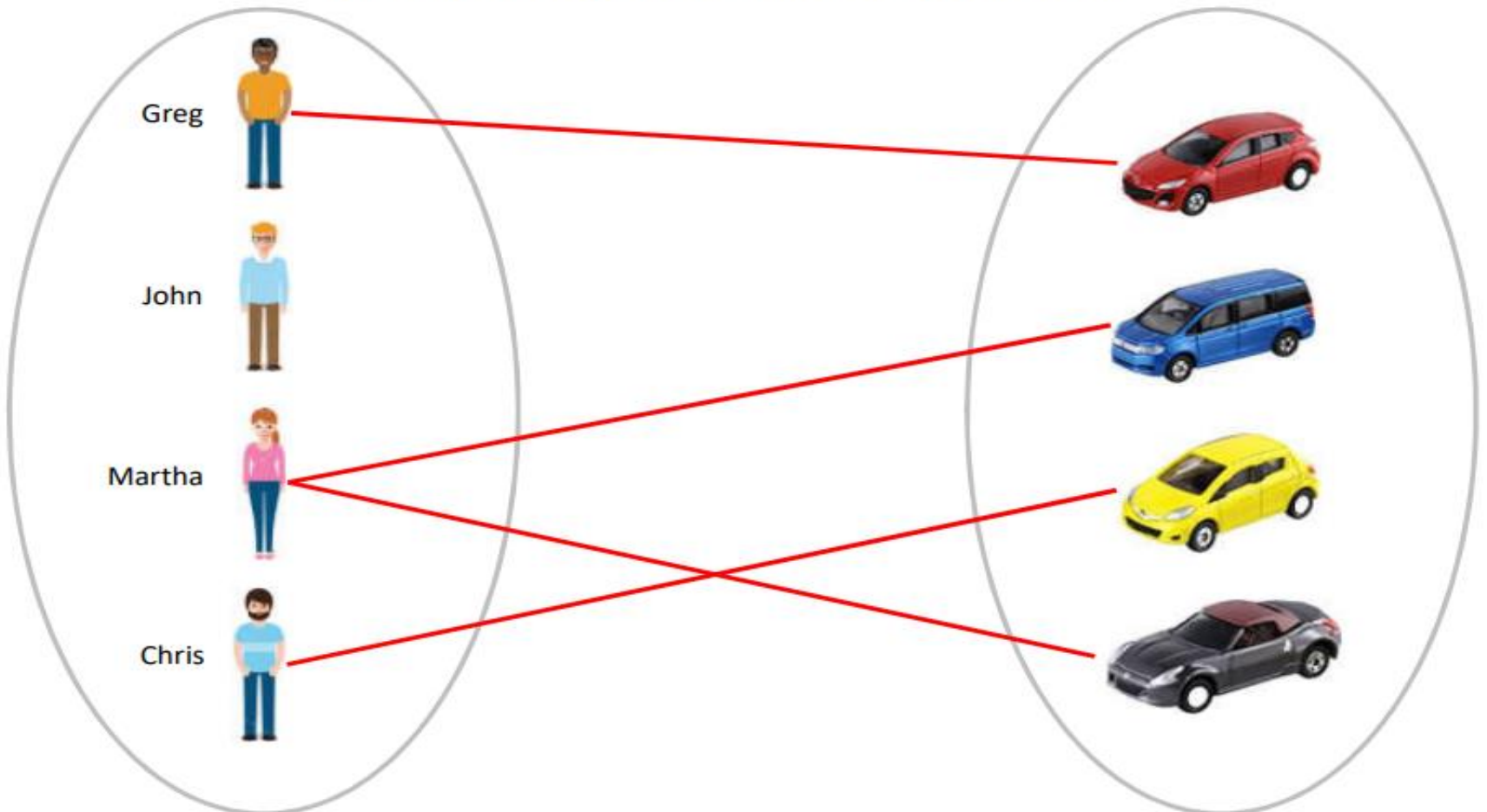
Mínimo: 1

Máximo: 1



Engenharia de Software - AP00

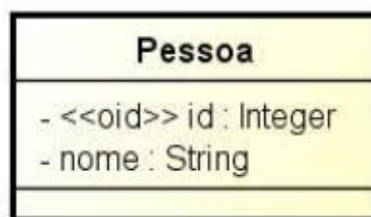
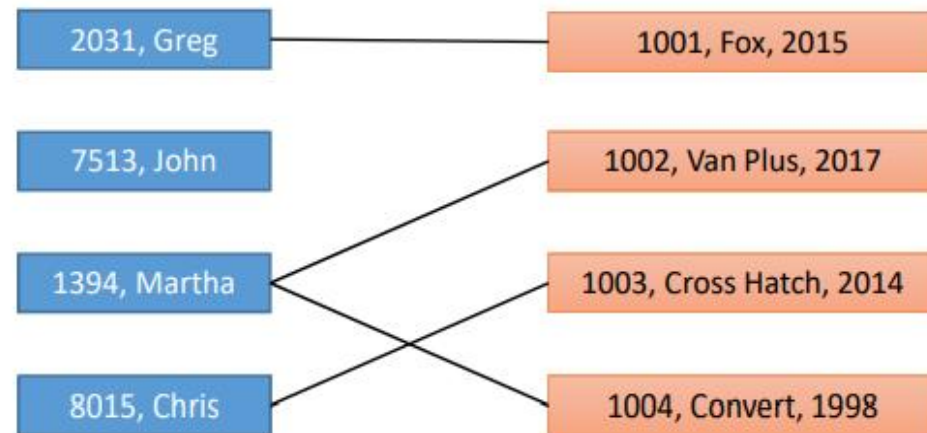
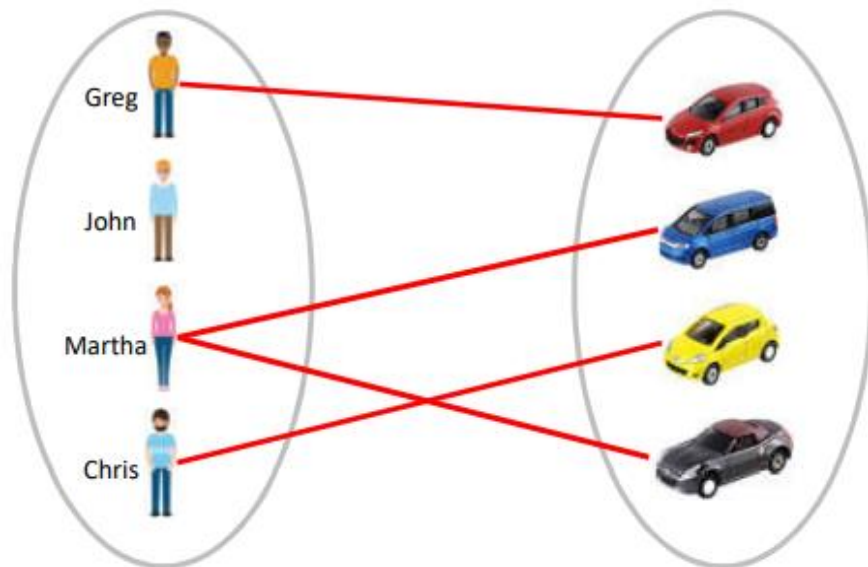
Recordando: quem é dono de qual carro?



Engenharia de Software - APOO

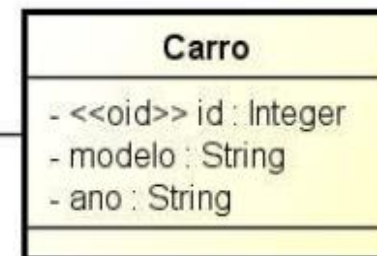
Mundo real

Objetos



1

- dono

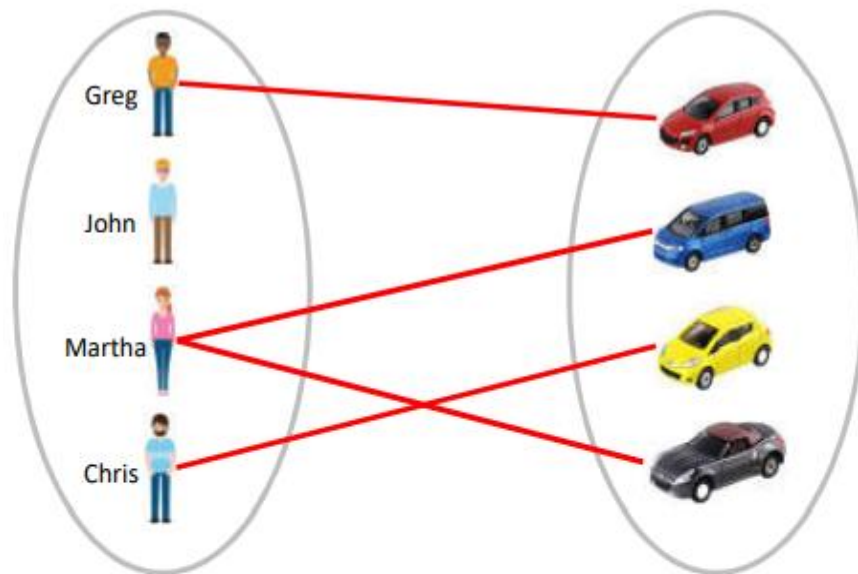


Engenharia de Software - APOO



Como encontrar as
multiplicidades?

Engenharia de Software - APOO



Pergunte para os dois lados:

1 <conceito> pode ter quantos <papel>?

1 carro pode ter quantos donos?

1 pessoa pode ter quantos carros?

Atenção:
sempre inicie
a pergunta
com "1"



Engenharia de Software - AP00



Multiplicidades possíveis

"," significa "ou"

".." significa "a" ou até

"*" significa "vários" (sem limite específico)

a)	1	exatamente um
b)	2	exatamente dois
c)	0..1	zero a um
d)	0..*	zero ou mais
e)	*	zero ou mais
f)	1..*	um ou mais
g)	2..*	dois ou mais
h)	2..5	de dois a cinco
i)	2,5	dois ou cinco
j)	2,5..8	dois ou cinco a oito

Engenharia de Software - APOO

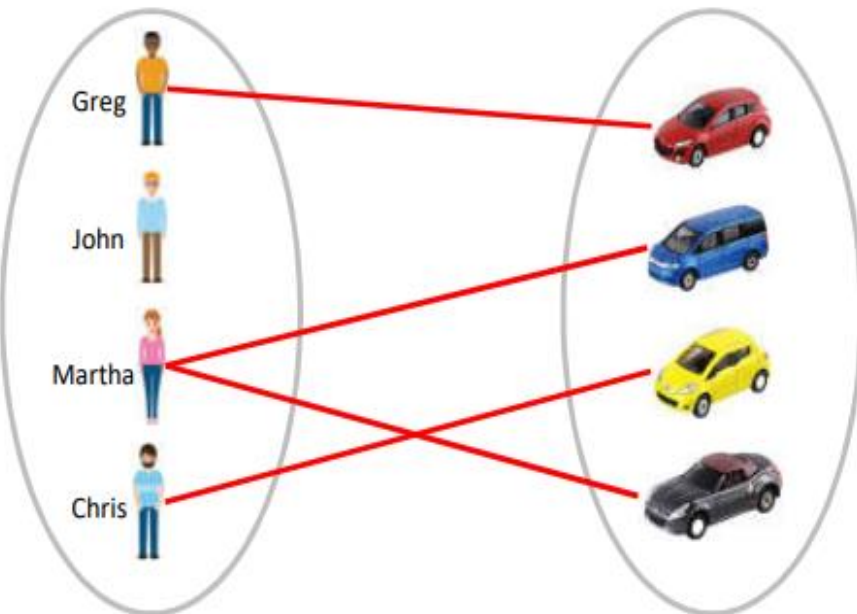


Associações comuns

Engenharia de Software - APOO

Um para muitos

Exemplo: quem é dono de cada carro?

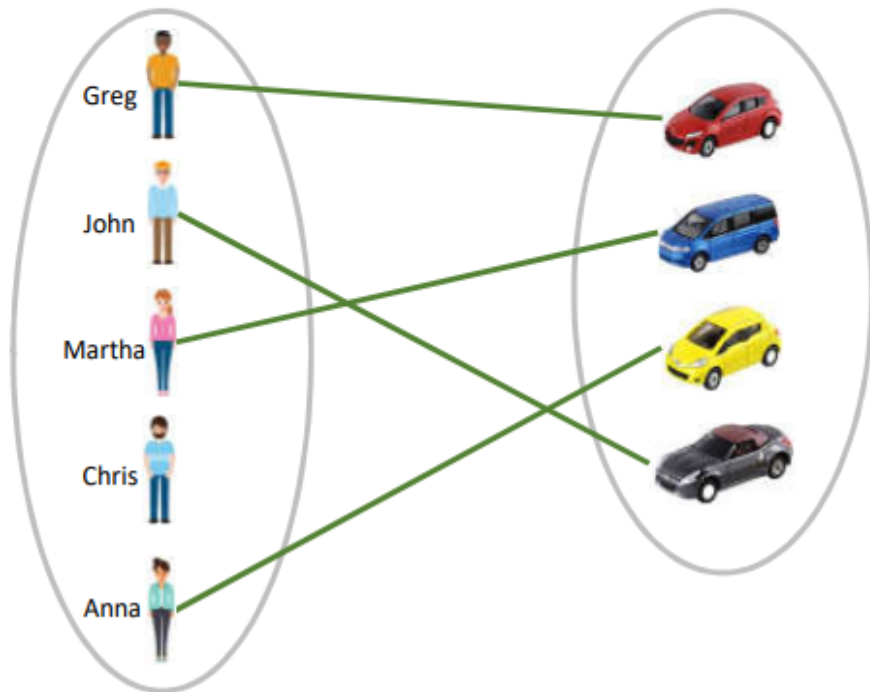


- Em um dos lados o máximo é 1
- No outro lado o máximo é "vários"

Engenharia de Software - APOO

Um para um

Exemplo: quem é o responsável por cada carro?



1 carro pode ter quantos responsáveis?

1 pessoa pode ser responsável por quantos carros?

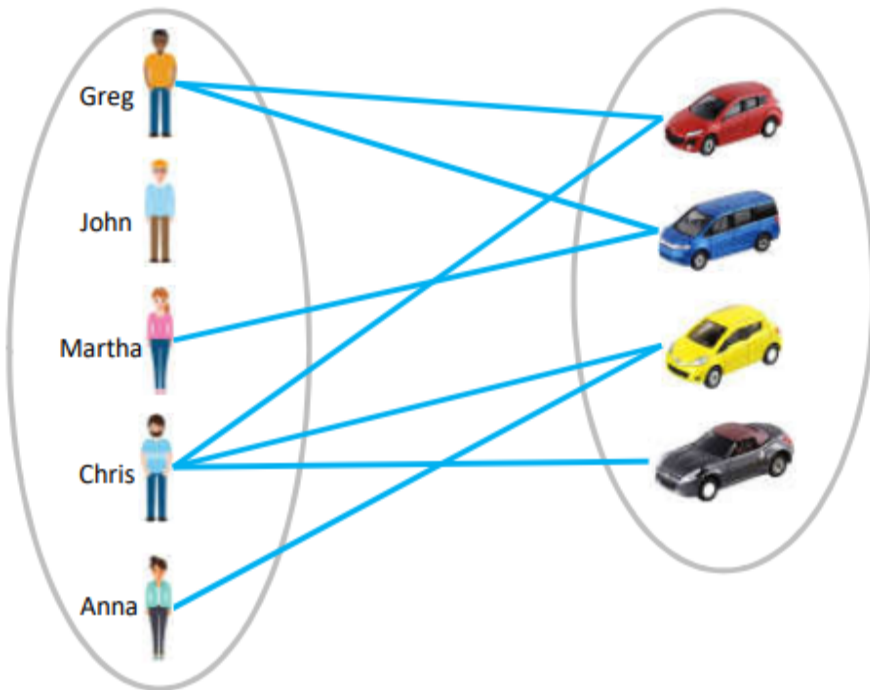


- Em ambos os lados o máximo é 1

Engenharia de Software - APOO

Muitos para muitos

Exemplo: quem dirige cada carro?



1 carro pode ter quantos motoristas?

1 pessoa pode dirigir quantos carros?



- Em ambos os lados o máximo é "vários"

Engenharia de Software - APOO



Resumo da aula

- Multiplicidade é a quantidade **mínima** e **máxima** de objetos que uma associação permite em cada um de seus papéis.
- Como encontrar as multiplicidades?
 - Pergunte para os dois lados: **1** <conceito> pode ter quantos <papel>?
- Multiplicidades possíveis
- Associações comuns
 - Um para muitos
 - Um para um
 - Muitos para muitos

Engenharia de Software - AP00



**Conceito dependente
Associações obrigatórias,
múltiplas e autoassociações**

Engenharia de Software - APOO

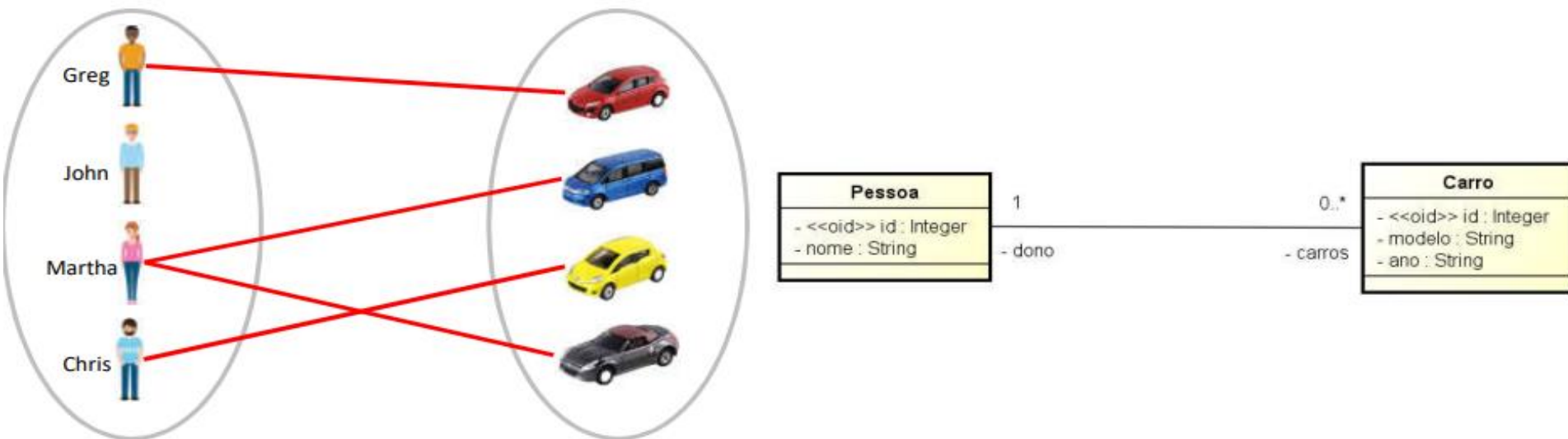


Associação obrigatória

Engenharia de Software - APOO

Definição

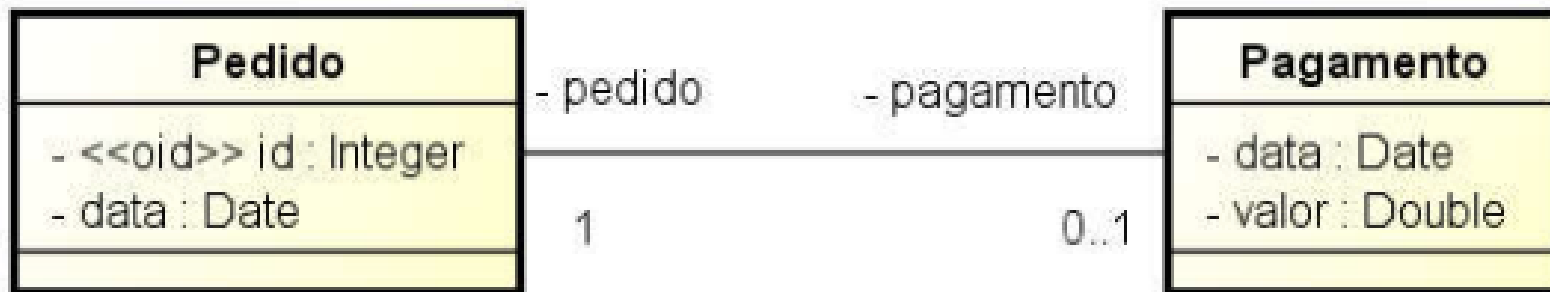
Uma associação é obrigatória se o conceito associado desempenha um papel de multiplicidade mínima maior que zero



- A associação de uma pessoa com carros não é obrigatória.
- A associação de um carro com dono é obrigatória.

Engenharia de Software - APOO

Atenção



Embora o pagamento seja "obrigatório" em um pedido, um pedido pode existir temporariamente sem um pagamento.

Não confunda este caso com o conceito de associação obrigatória que acabamos de aprender.

Engenharia de Software - APOO

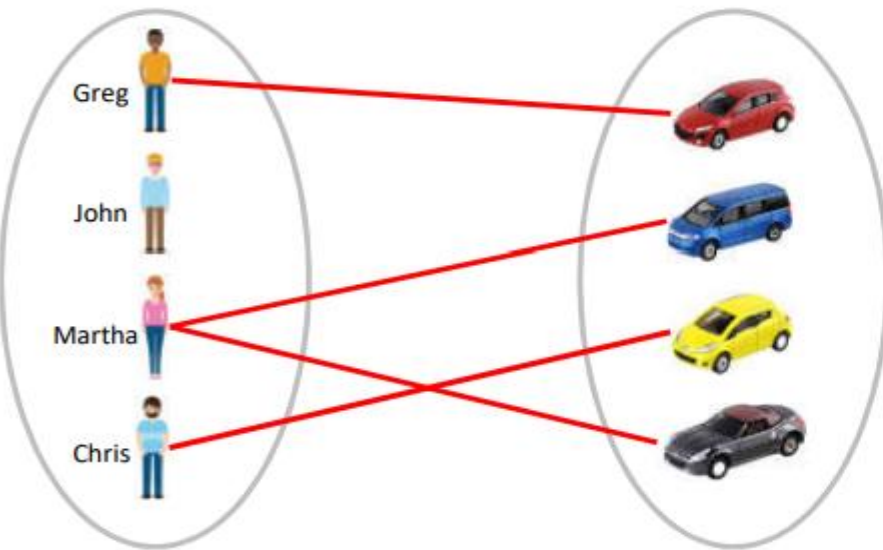


Conceito dependente

Engenharia de Software - APOO

Definição

Um conceito é dependente se ele possuir pelo menos uma associação obrigatória.



Para cadastrar um carro no sistema é obrigatoriamente deve de estar associado uma pessoa



- Pessoa é um conceito independente
- Carro é um conceito dependente

Só pode existir se existir uma pessoa dona dele

Se o dono deixar de existir, o carro também deixa de existir

Engenharia de Software - APOO

Nota

A UML tem um símbolo que denota dependência de um modo geral, mas que não acrescenta valor prático à modelagem conceitual:



"Carro depende de pessoa"

Engenharia de Software - APOO

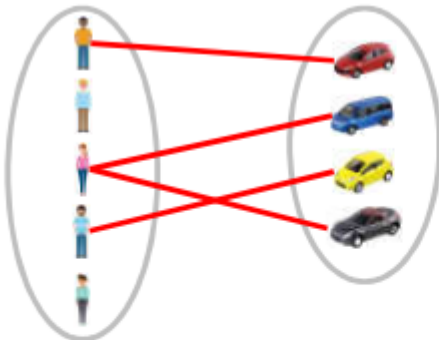


Associações múltiplas

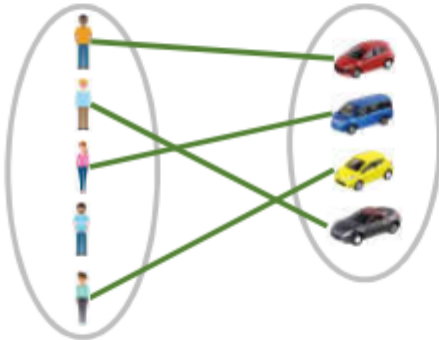
Pode haver mais de uma associação entre dois conceitos?

Engenharia de Software - APOO

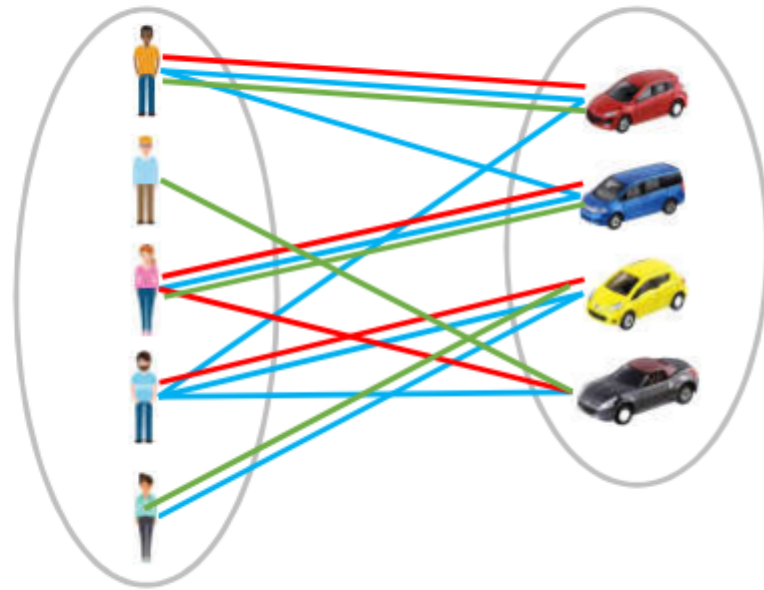
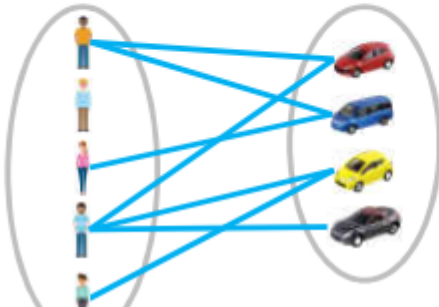
Quem é dono de cada carro?



Quem é responsável por cada carro?



Quem dirige cada carro?



É possível um modelo de conceitual com todas as três informações ao mesmo tempo?

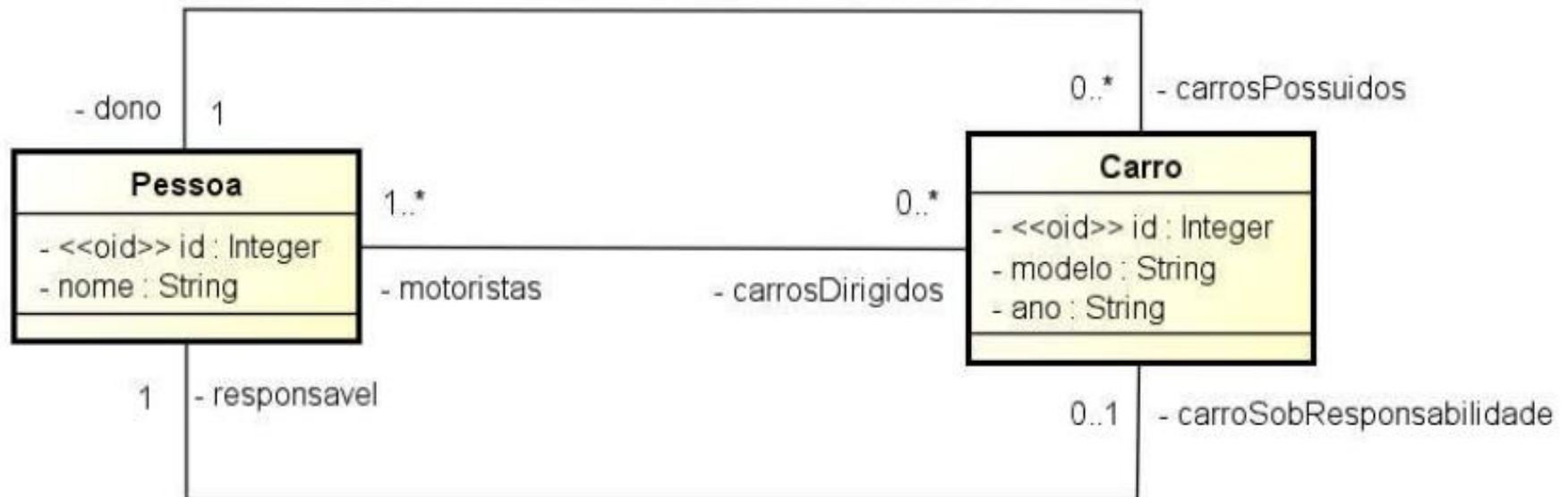
Engenharia de Software - APOO

Sim. Mas os nomes de papel devem ser únicos.

Quem é dono de cada carro?

Quem é responsável por cada carro?

Quem dirige cada carro?



Engenharia de Software - APOO

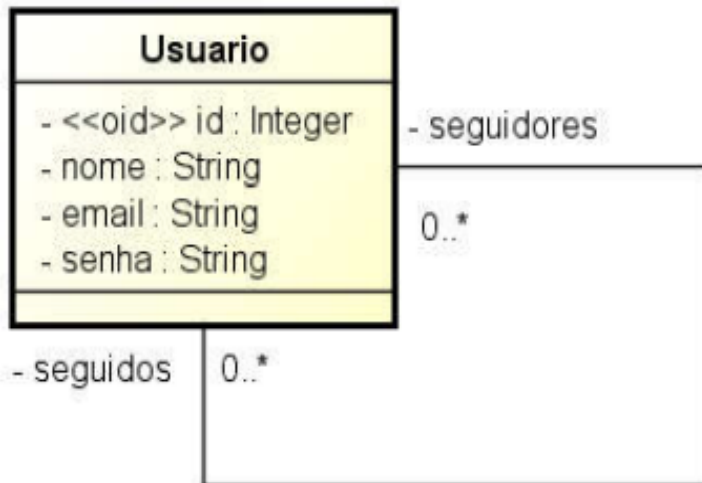


Autoassociações

Engenharia de Software - APOO

Autoassociações

Quando um conceito é associado com ele próprio.



1 usuário pode ter quantos seguidores?

1 usuário pode ter quantos seguidos?

Engenharia de Software - APOO



**Desenhando instâncias com
diagrama de objetos da UML**

Engenharia de Software - APOO

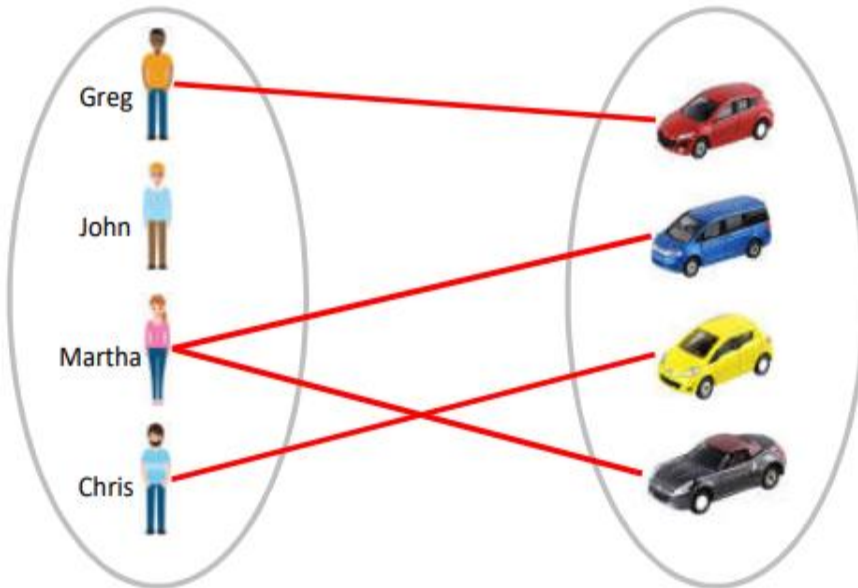


Relembrando:

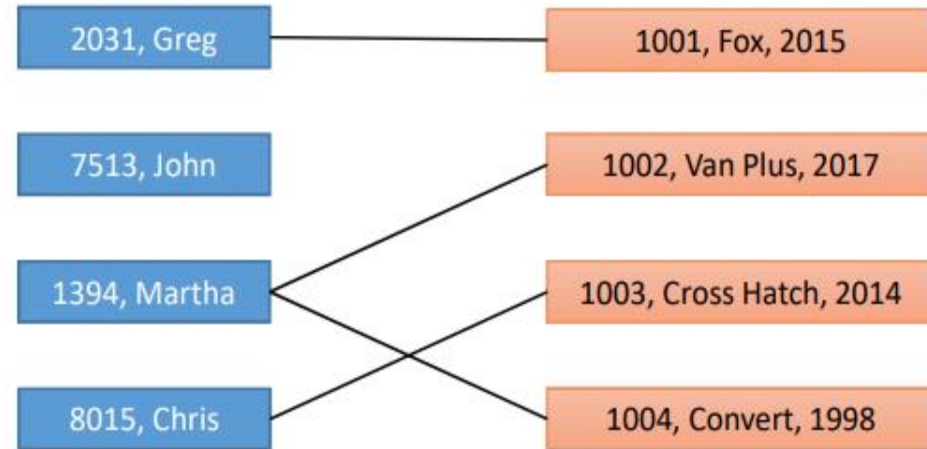
- O Modelo Conceitual representa a estrutura dos dados
 - Conceitos/atributos e como eles se inter-relacionam entre si
- Cada ocorrência de um conceito é chamada de instância ou objeto

Engenharia de Software - APOO

Mundo real



Objetos



Engenharia de Software - APOO

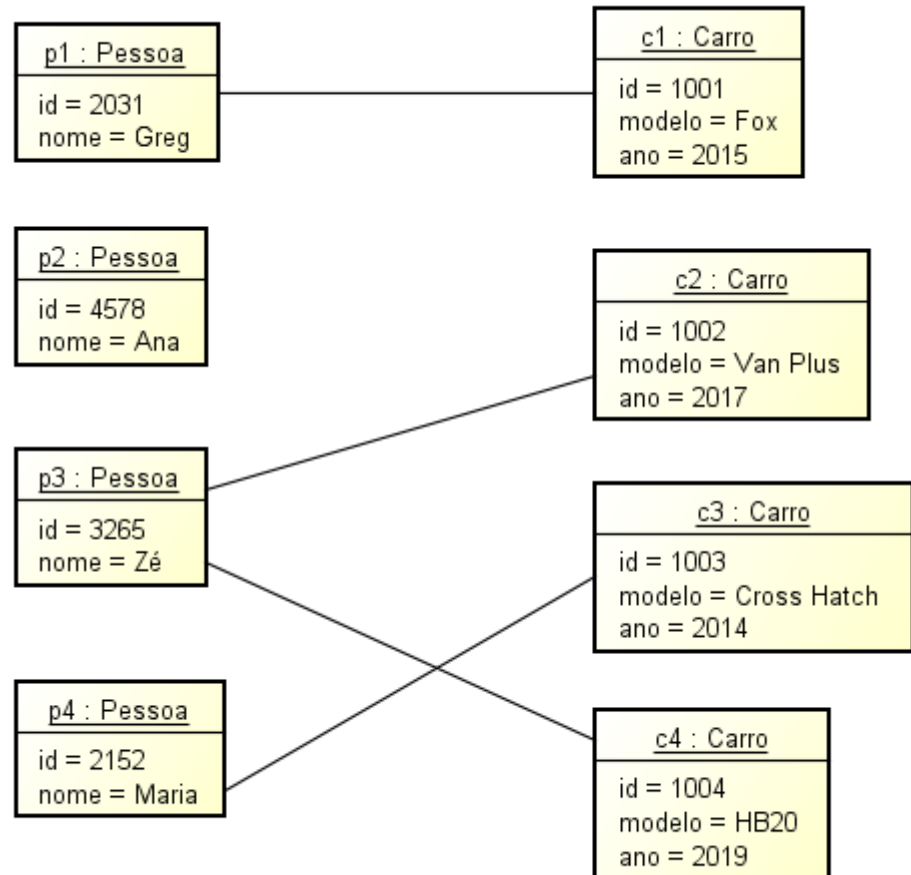


Pra quê visualizar as instâncias (ou objetos)?

- Ajuda a compreender
- Ajuda a descobrir problemas
- Ferramenta UML: diagrama de objetos

Engenharia de Software - APOO

Diagrama de Objetos:



Engenharia de Software - APOO

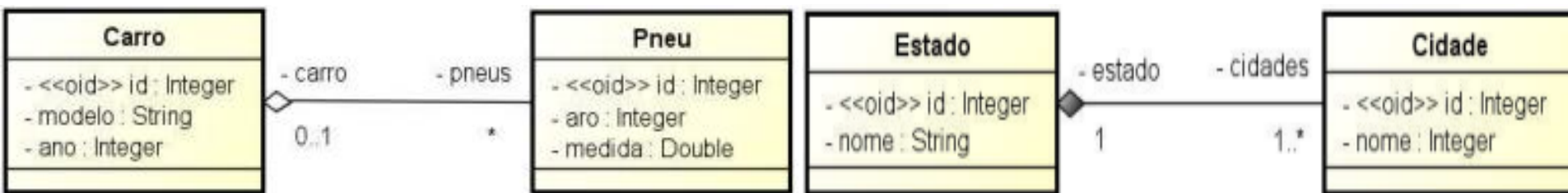


Associações todo-parte:
agregação e composição

Engenharia de Software - APOO

Associações todo-parte:

Quando um conceito é parte de outro que representa um todo, desenhamos um diamante no lado do todo.



Diamante branco: "Agregação"

O conceito parte não é exclusivo

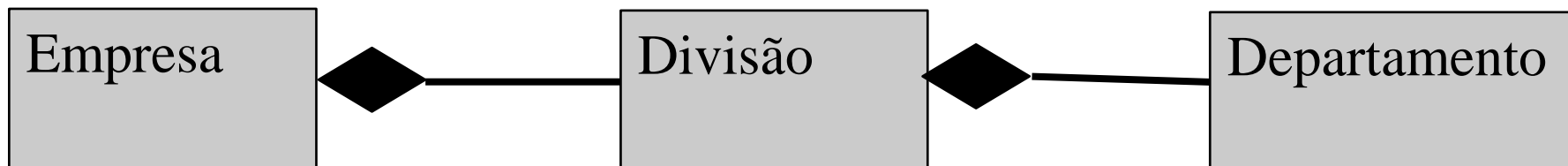
Diamante preto: "Composição"

O conceito parte é **exclusivo**

Engenharia de Software - APOO

Composição:

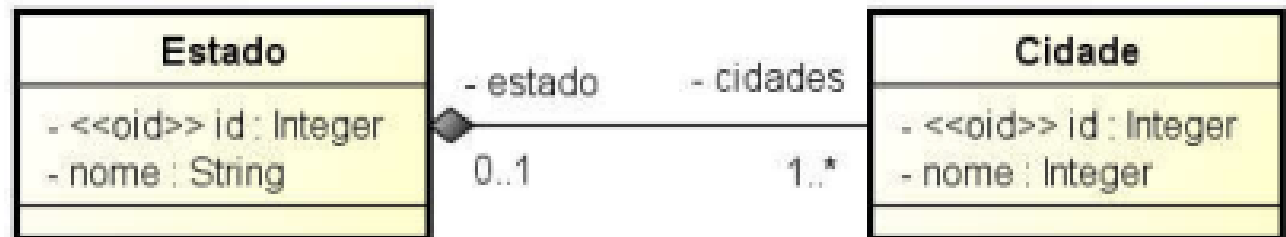
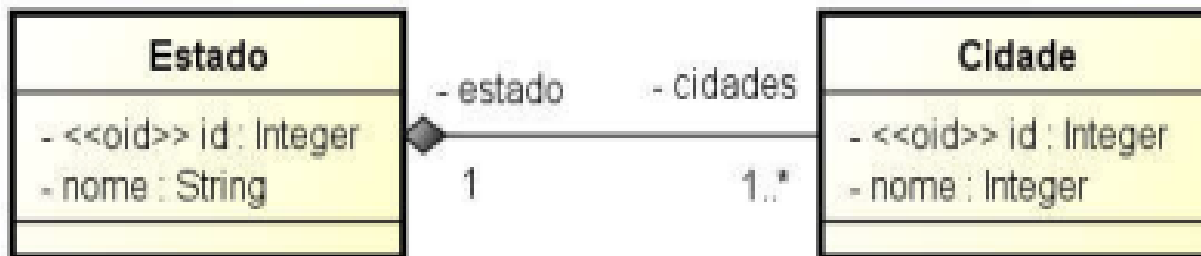
- A classe **Todo Empresa** contém do seu ponto de vista uma classe **Parte Divisão**, que por sua vez, do ponto de vista da classe **Parte Departamento**, passa a ser uma classe **Todo**.



Engenharia de Software - APOO

Exclusividade: 1 ou 0..1

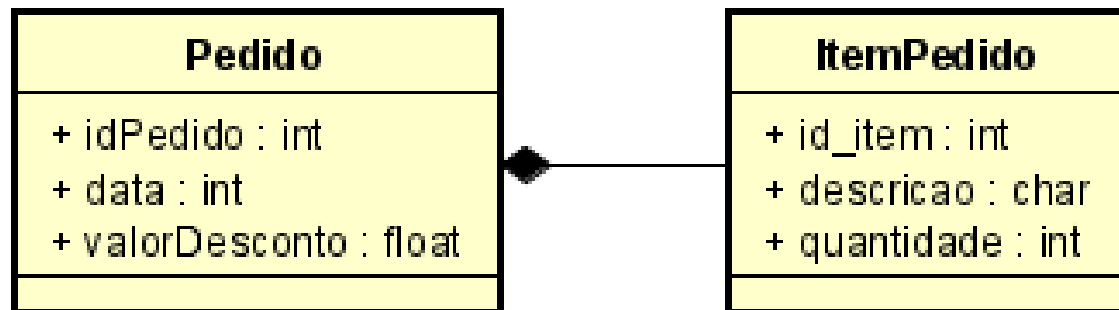
Como a composição (diamante preto) é uma relação exclusiva, a multiplicidade no lado do diamante sempre será 1 ou 0..1



Engenharia de Software - APOO

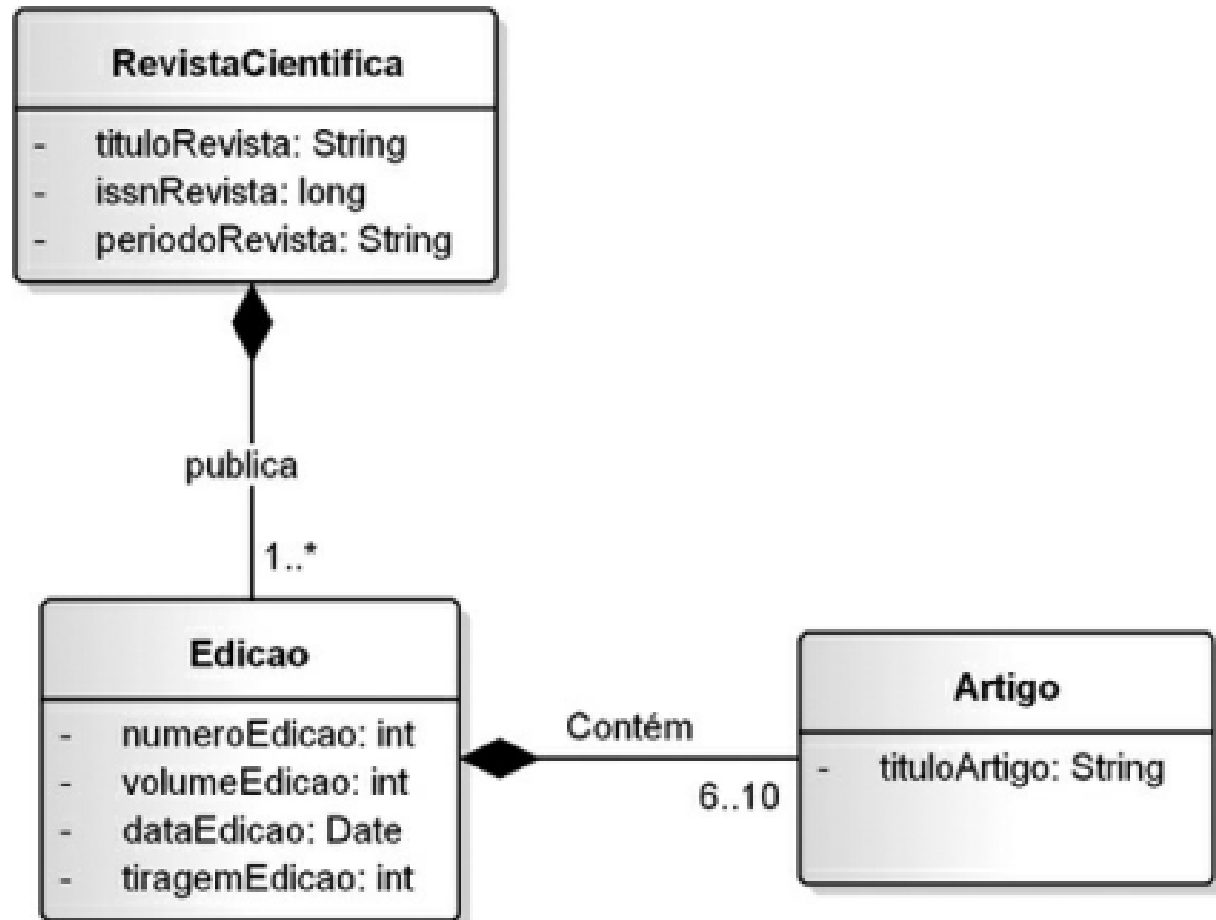
Composição:

- O objeto-parte não pode existir sem o objeto-todo.
- Se o objeto-todo for destruído, o objeto-parte também será.



Engenharia de Software - APOO

Exemplo de Composição



Engenharia de Software - APOO



Agregação:

- É que uma das classes participantes do relacionamento é **uma parte** ou **está contida** em outra classe.
- Estas são as **palavras-chave** mais comuns para **identificar um agregação**:
 - ✓ Consiste em
 - ✓ Contém
 - ✓ É parte de
- Uma agregação descreve como a classe que possui a regra do todo é composta (tem) de outras classes, que possuem a regra das Partes.

Engenharia de Software - APOO

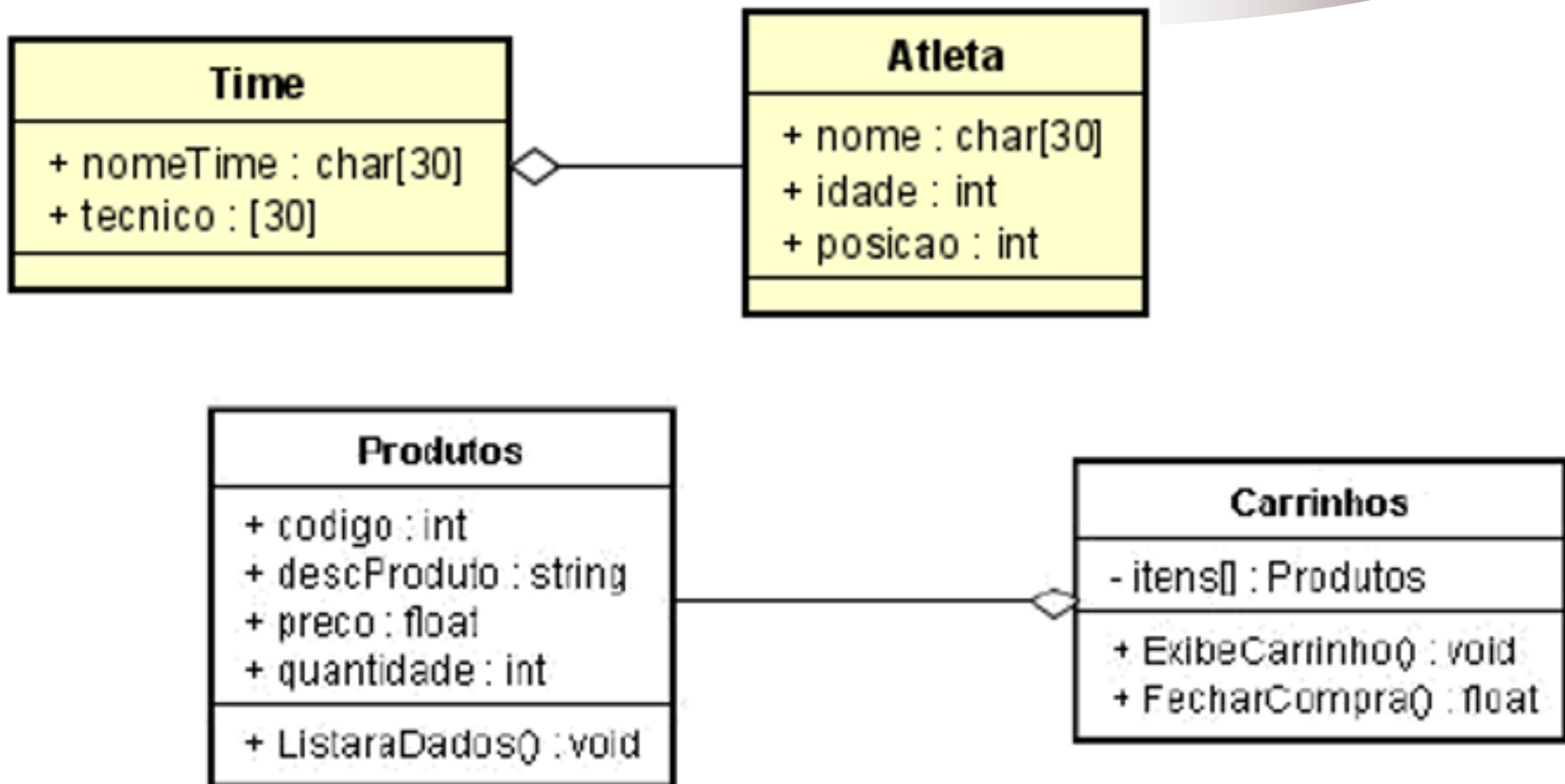


Agregação:

- É representado por um segmento de reta com uma diamante (losango sem preenchimento) próxima a classe Todo.
- A associação de agregação pode, em muitos casos, ser substituída por uma associação binária simples, dependendo da visão de quem faz a modelagem.

Engenharia de Software - APOO

Exemplo de Agregação



Engenharia de Software - APOO

Exemplo de Agregação



Engenharia de Software - APOO

Ressalva

O diamante (seja branco, seja preto) deve ser usado **somente em casos em que realmente se trata de uma relação todo-parte**



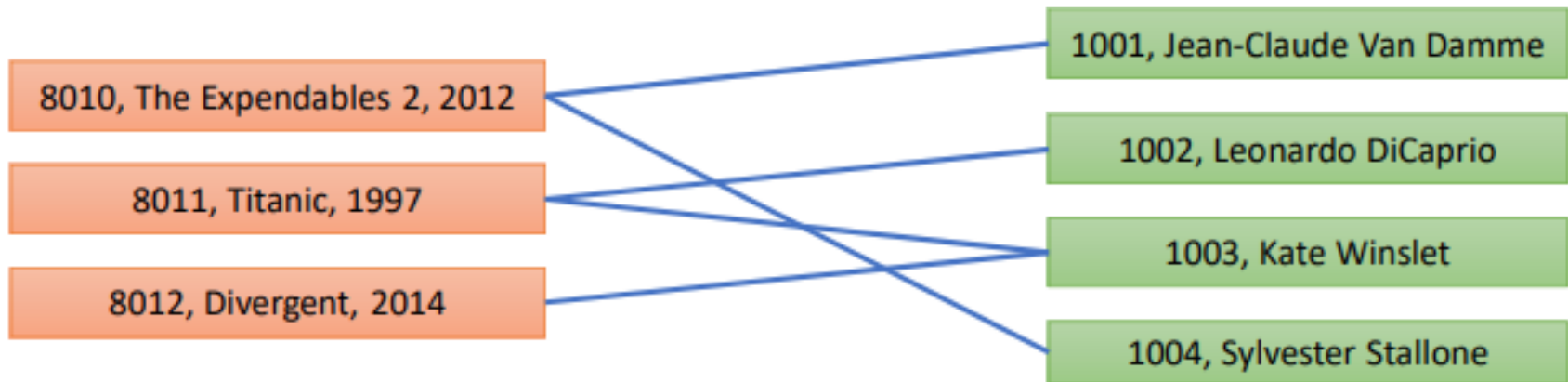
Engenharia de Software - AP00



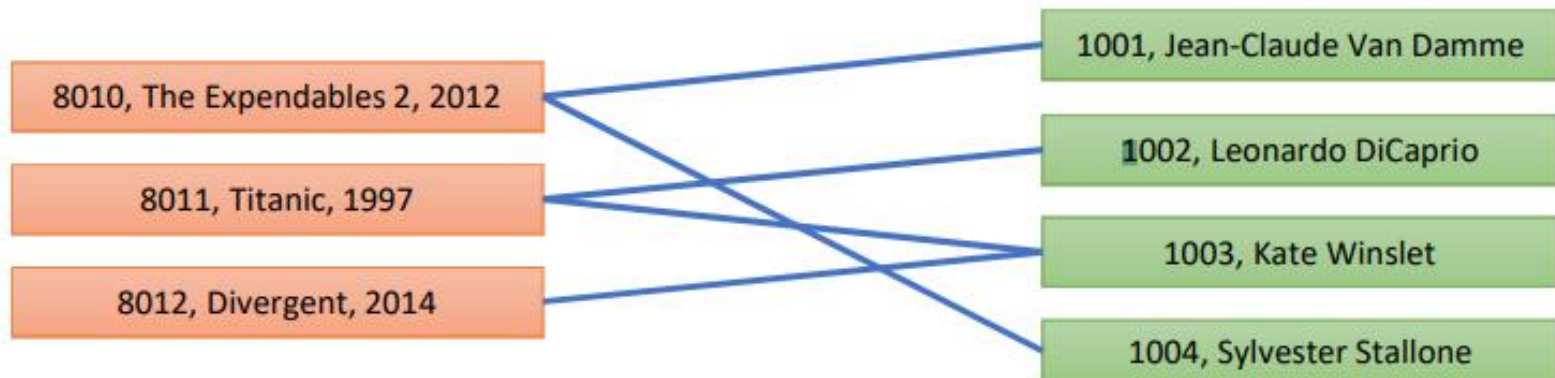
Enumerações

Engenharia de Software - APOO

Deseja-se fazer um sistema para manter um cadastro de filmes e artistas (atores/atrizes), bem como a informação de qual artista atuou em cada filme.



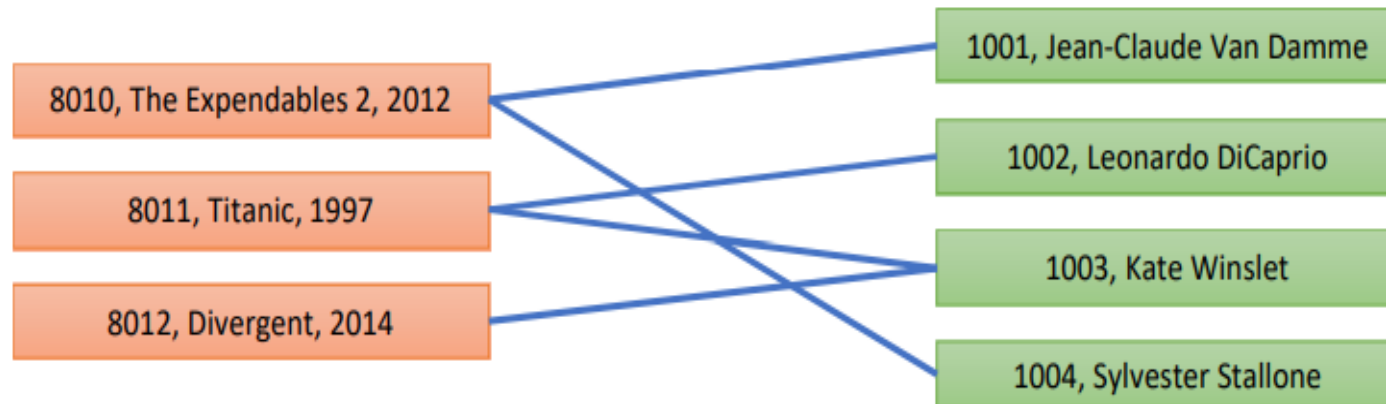
Engenharia de Software - APOO



Engenharia de Software - APOO

Problema

Além disso, desejo saber também o nome do personagem desempenhado por cada artista em cada filme



Problema

Além disso, desejo saber também o nome do personagem desempenhado por cada artista em cada filme

8010, The Expendables 2, 2012

8011, Tinatic, 1997

8012, Divergent, 2014

1001, Jean-Claude Van Damme, **Vilain**

1002, Leonardo DiCaprio, **Jack Dawson**

1003, Kate Winslet, **Rose**

1005, Kate Winslet, **Jeanine**

1004, Sylvester Stallone, **Barney Ross**

ERRADO

8010, The Expendables 2, 2012, **Vilain**

8013, The Expendables 2, 2012, **Barney Ross**

8011, Tinatic, 1997, **Jack Dawson**

8014, Tinatic, 1997, **Rose**

8012, Divergent, 2014, **Jeanine**

1001, Jean-Claude Van Damme

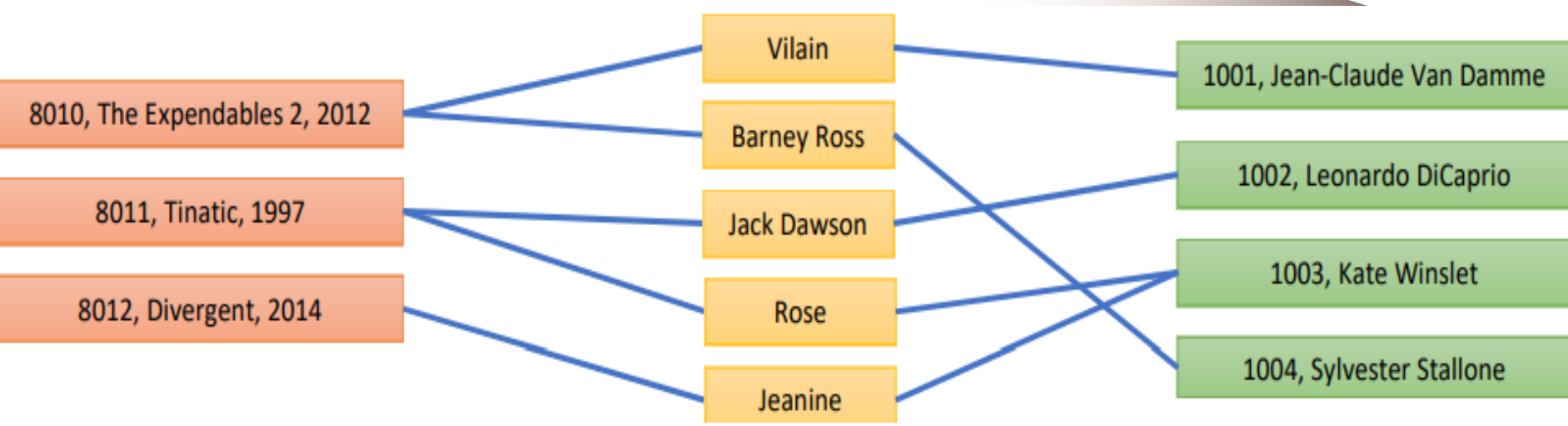
1002, Leonardo DiCaprio

1003, Kate Winslet

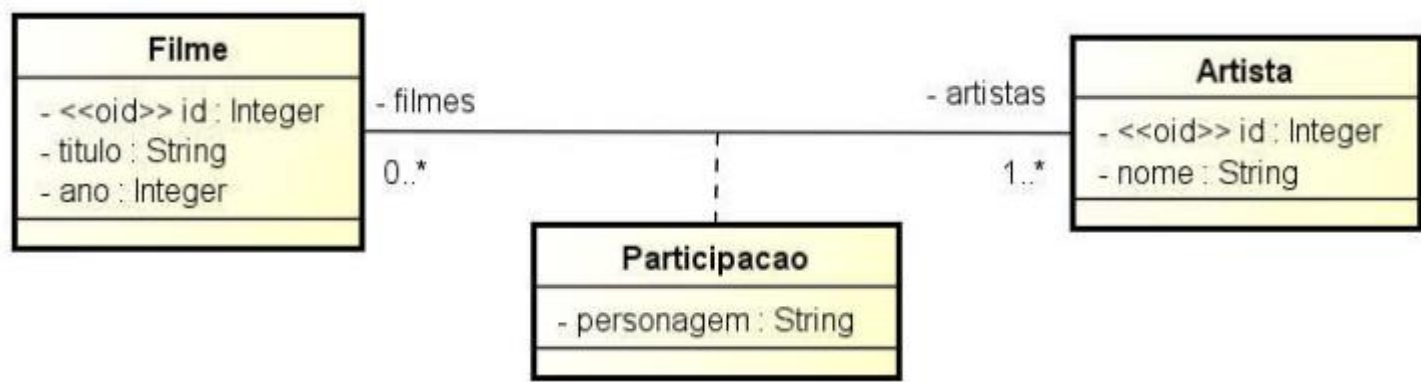
1004, Sylvester Stallone

ERRADO

Conclusão: o nome do personagem é um dado que pertence à associação.
Neste caso, o nome do personagem deverá ser armazenado em um objeto intermediário.

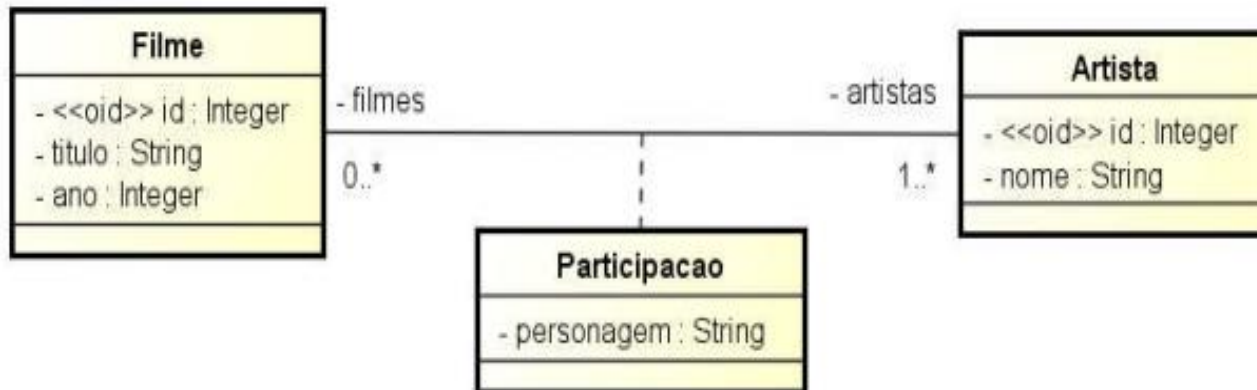
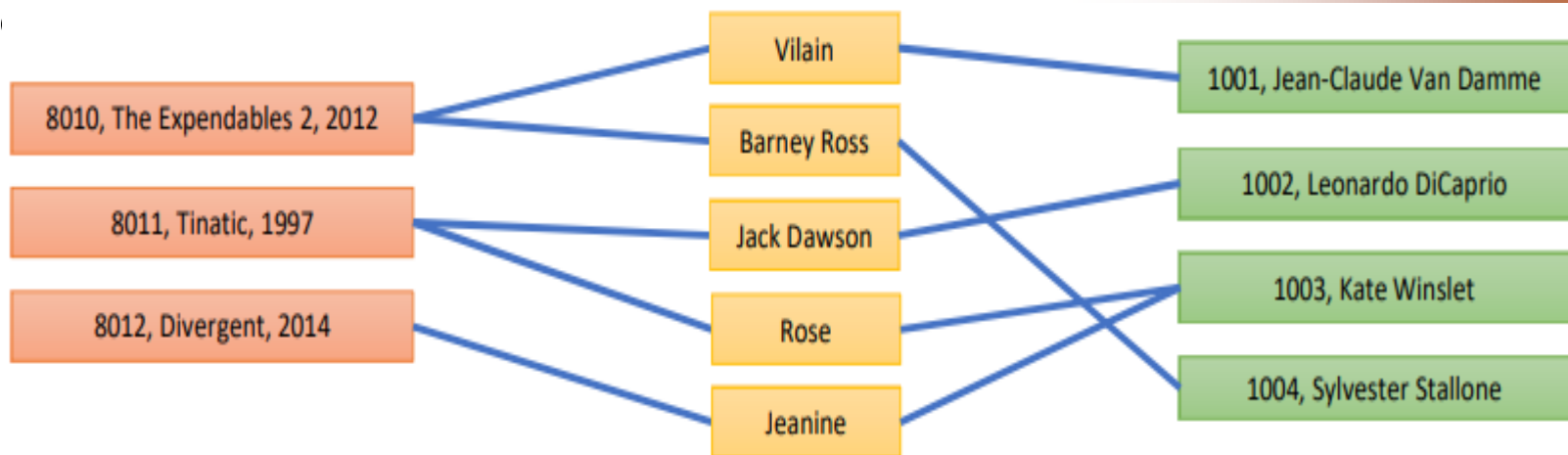


Classe de associação:



Atenção!

A classe de associação indica associação ÚNICA!

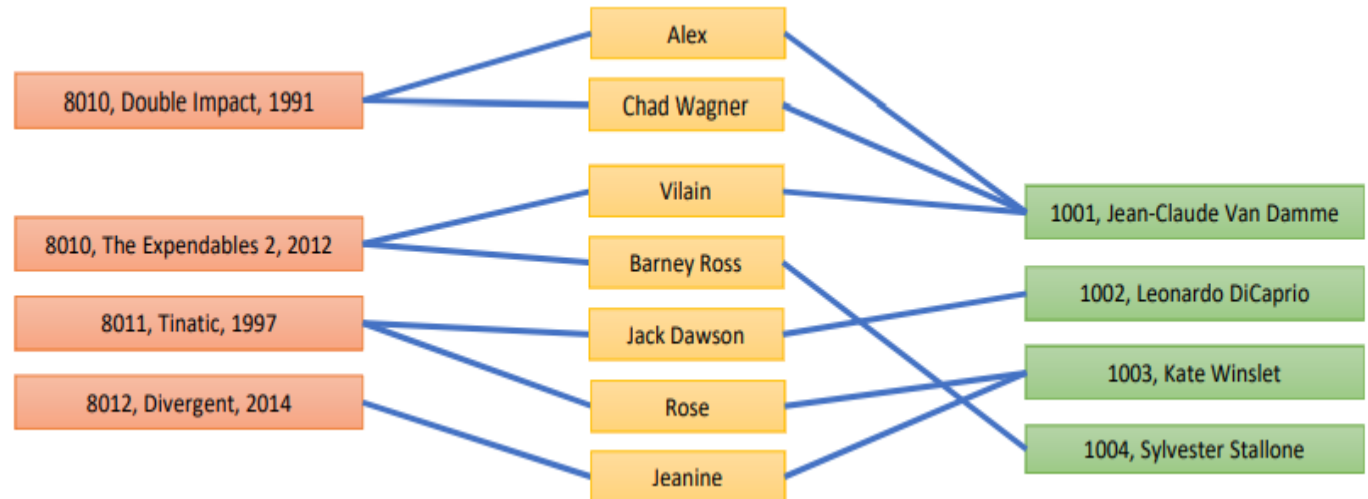


Este modelo indica que para um mesmo artista e um mesmo filme, só pode haver uma participação

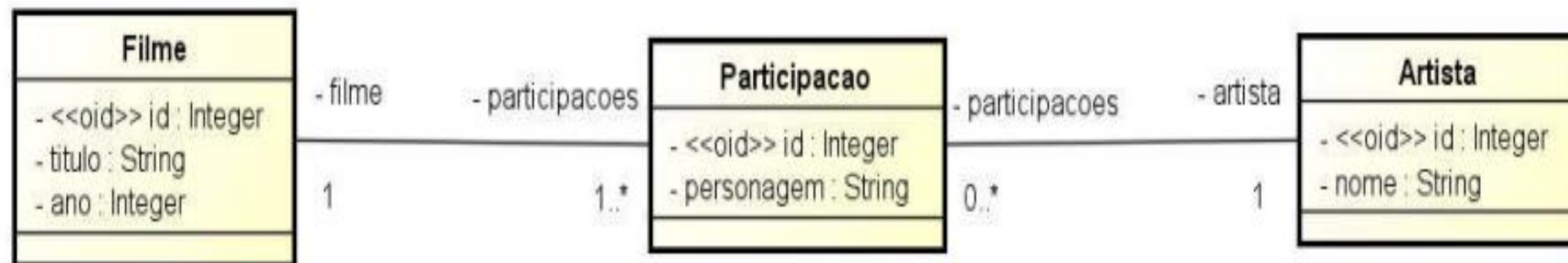
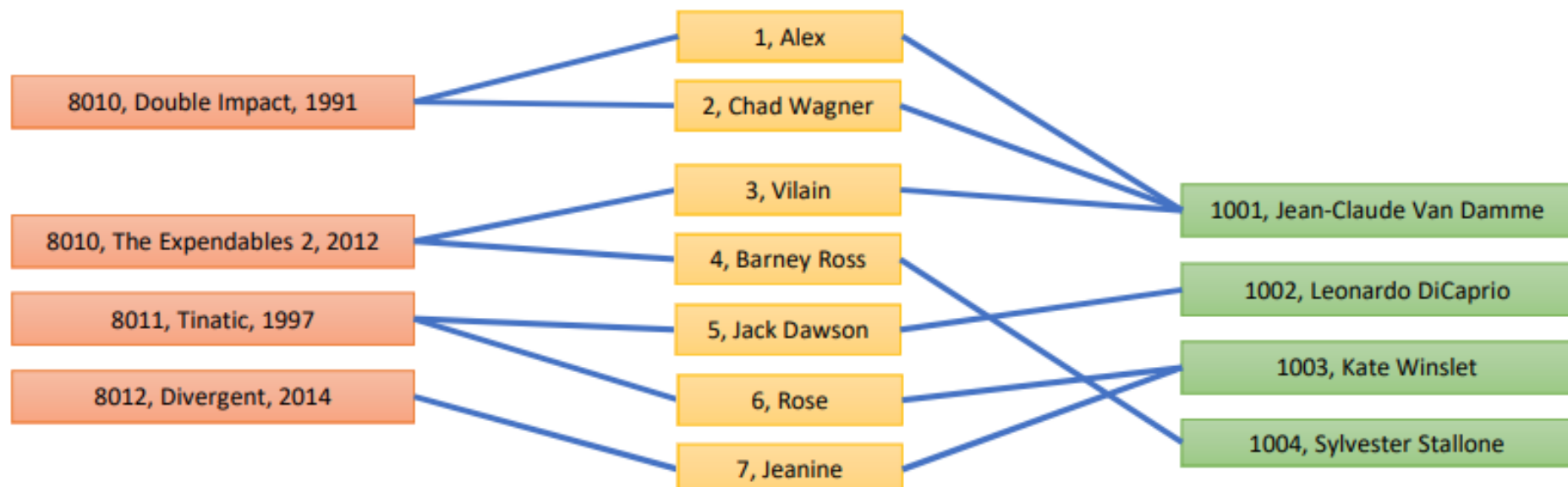
Isso não pode:



Então como representar um modelo no qual um mesmo artista pode representar mais de um personagem em um mesmo filme?



Tem que usar uma **classe normal** e **nunca usar** uma **classe de associação**!!!!



Resumo da aula

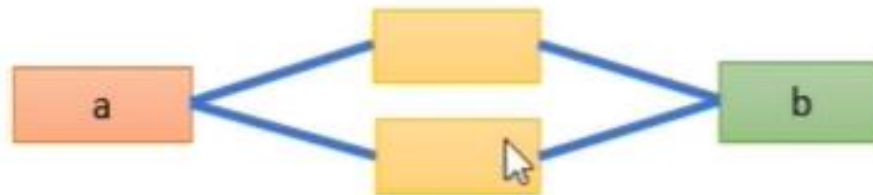


Associação muitos-para-muitos: ALERTA!

Tem atributo na associação?

SIM: então haverá um novo conceito no meio!

O mesmo "A" pode estar associado com o mesmo "B" mais de uma vez?



O mesmo artista pode participar do mesmo filme mais de uma vez?

Engenharia de Software - AP00



Enumerações

Engenharia de Software - APOO

São valores que ele pode assumir durante o seu estado de vida

Uma enumeração pode ser considerada um "meio termo" entre um conceito e um atributo.

Ser para identificar os valores do conceito pelo nome e não pelo número
Internamente são identificados pelo número

<<enumeration>> EstadoDoPedido	
- <<enum constant>>	AGUARDANDO_PAGAMENTO : int
- <<enum constant>>	ENVIADO : int
- <<enum constant>>	ENTREGUE : int
- <<enum constant>>	DEVOLVIDO : int
- <<enum constant>>	CANCELADO : int

<<enumeration>> DiaDaSemana	
- <<enum constant>>	Domingo : int
- <<enum constant>>	Segunda : int
- <<enum constant>>	Terca : int
- <<enum constant>>	Quarta : int
- <<enum constant>>	Quinta : int
- <<enum constant>>	Sexta : int
- <<enum constant>>	Sabado : int

Engenharia de Software - APOO

Para simplificar o diagrama principal do sistema, recomenda-se incluir as enumerações em um diagrama separado, representando seu nome no diagrama do sistema diretamente como um tipo de atributo.

Diagrama de enumerações

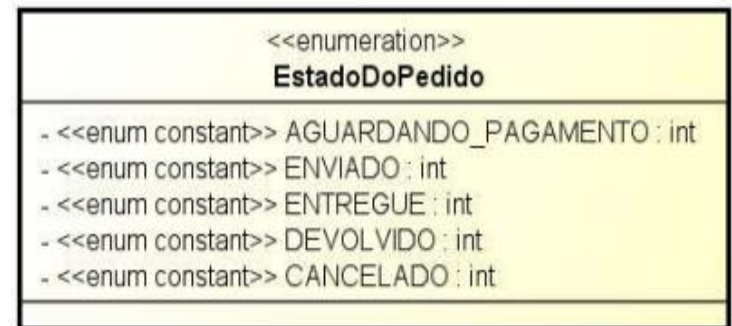
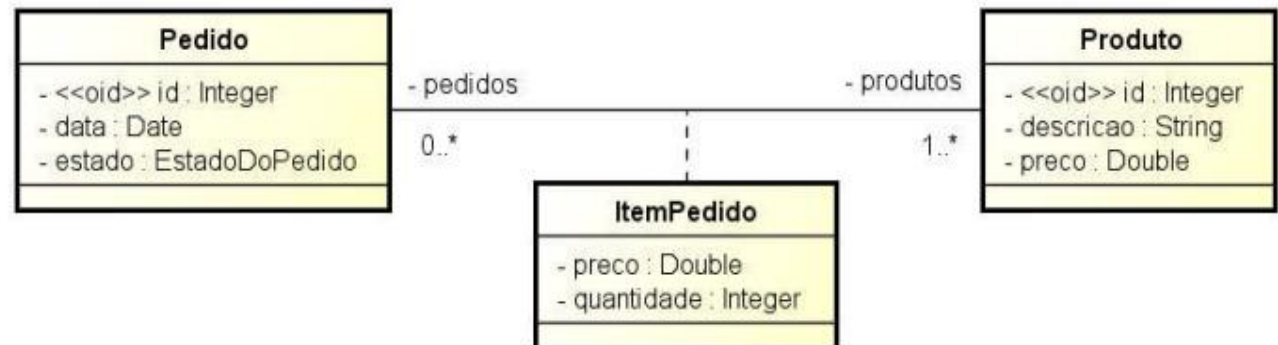


Diagrama do sistema



Exercicio:

- Construa um diagrama de classe seguindo o problema apresentado a seguir.
- Uma seguradora de veículos deseja construir um sistema para controlar seus clientes e funcionários.
- Para isso deseja manter um cadastro para os clientes deseja-se manter os seguintes dados: nome, telefone, data de nascimento e endereço.
- Para os veículos são necessários: modelo do veículo, marca, ano e quilometragem do veículo no momento da criação do seguro. Ainda é necessário ter um cadastro de funcionário, onde cada funcionário pode ser responsável pela assessoria de alguns clientes, para os funcionários é necessário armazenar: nome, telefone e endereço.
- Sobre as assessoria é necessário manter: cliente, funcionário, motivo da consultoria e data realizada.