



**UAlg FCT**

UNIVERSIDADE DO ALGARVE  
FACULDADE DE CIÊNCIAS E TECNOLOGIA

# Bases de Dados

---

DESENHO DE BASES DE DADOS – MODELO  
ENTIDADE ASSOCIAÇÃO

# Fases de desenho

---

Fase inicial – caracterizar as necessidades de dados dos possíveis utilizadores da base de dados.

Segunda fase – escolha de um modelo de dados

- Aplicar os conceitos do modelo de dados escolhido
- Traduzir esses requisitos num esquema conceptual da base de dados.
- Um esquema conceptual totalmente desenvolvido indica os requisitos de dados para um determinado domínio.
  - Descreve os tipos de operações (ou transações) que serão realizadas nos dados.

# Fases de desenho (Cont.)

- Fase Final – Passar de um modelo de dados abstrato para a implementação da base de dados
  - Desenho Lógico – Decidir sobre o esquema da base de dados.
    - O desenho da base de dados exige que encontremos um “bom” conjunto de *relações*.
    - Decisão de negócio – Que atributos devemos registrar na base de dados?
    - Decisão de especialista de BD – Que relações devemos ter e como os atributos devem ser distribuídos entre as várias relações?
  - Desenho Físico – Decidir sobre o layout físico da base de dados

# Alternativas de desenho

- Ao projetar um esquema de base de dados, devemos garantir que evitamos duas *armadilhas* principais:
  - Redundância: um mau desenho pode resultar em informação repetida.
    - A representação redundante da informação pode levar à inconsistência dos dados entre as diversas cópias da informação
  - Incompletude: um mau desenho pode tornar certos aspetos dos dados difíceis ou impossíveis de modelar.
- Evitar desenhos maus não é suficiente. Pode haver um grande número de bons desenhos que devemos escolher.

# Abordagens de desenho

## ■ Modelo Entidade Associação

- Modela o problema como um conjunto de *entidades* e *associações*
  - Entidade: uma “coisa” ou “objeto” que é distinguível de outros objetos
    - Descrito por um conjunto de *atributos*
  - Associação: uma associação entre diversas entidades
- Representado graficamente por um *diagrama Entidade-Associação (ER, do inglês Entity Relationship)*

---

# Modelo Entidade-Associação (ER)

# Conjuntos de entidades

---

Uma **entidade** é um objeto que existe e é distinguível de outros objetos.

- Exemplo: pessoa específica, empresa, sensor

Um **conjunto de entidades** é um conjunto de entidades do mesmo tipo, que compartilham as mesmas propriedades.

- Exemplo: conjunto de todas as pessoas, empresas, sensores

Uma entidade é representada por um conjunto de atributos; isto é, propriedades descritivas possuídas por todos os membros de um conjunto de entidades.

- Exemplo:

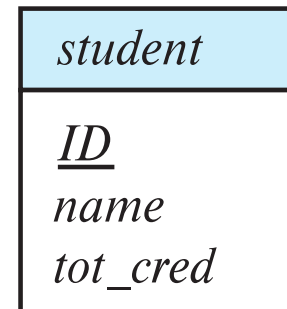
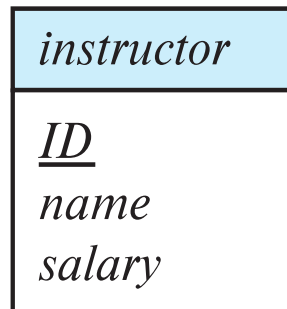
*professor* = ( *ID*, *nome*, *salário* )

*disciplina* = ( *disciplina\_id*, *título*, *créditos* )

Um subconjunto de atributos forma uma **chave primária** do conjunto de entidades; isto é, identifica univocamente cada membro do conjunto.

# Representando conjuntos de entidades no diagrama ER

- Conjuntos de entidades podem ser representados graficamente da seguinte forma:
  - Retângulos representam conjuntos de entidades.
  - Atributos listados dentro do retângulo da entidade
  - Sublinhado indica atributos da chave primária





# Conjuntos de associações (*Relationship Sets*)

---

Um **associação** faz a ligação entre várias entidades

Exemplo:

44553 (João )	<u>orientador</u>	22222 ( Maria)
aluno	conjunto de associações	professor

Um **conjunto de associações** é uma relação matemática entre  $n \geq 2$  entidades, cada uma pertencente a conjuntos de entidades

$$\{(e_1, e_2, \dots, e_n) \mid e_1 \in E_1, e_2 \in E_2, \dots, e_n \in E_n\}$$

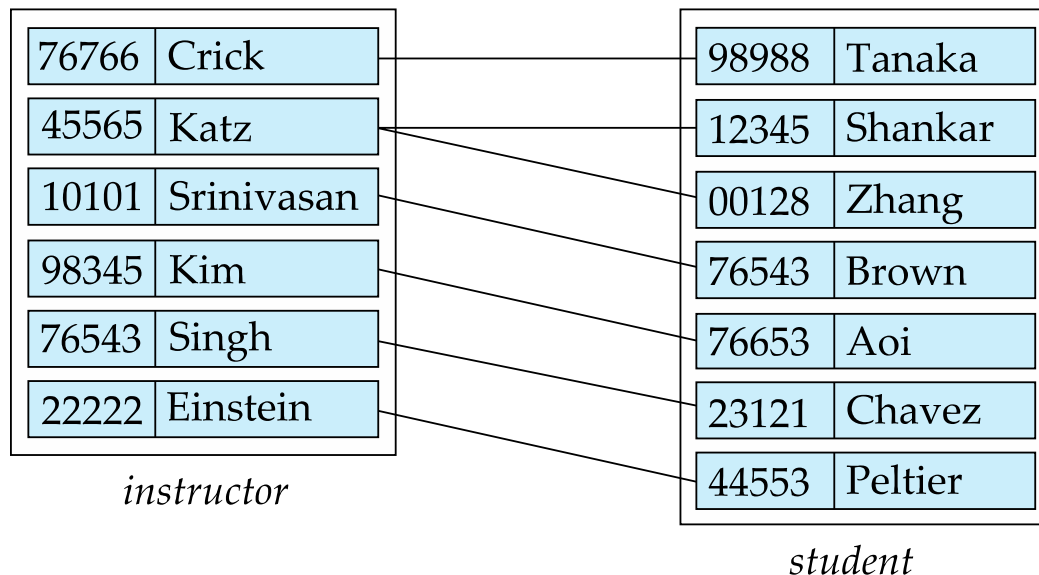
onde  $(e_1, e_2, \dots, e_n)$  é uma associação

- Exemplo:

$(44553, 22222) \in \text{orientador}$

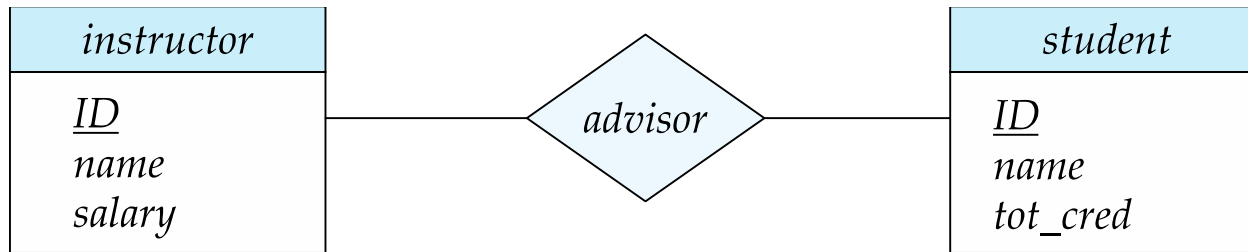
# Conjuntos de associações (Cont.)

- Exemplo: definimos o conjunto de associações *orientador* para denotar as associações entre os alunos e os professores que atuam como seus orientadores.
- Visualmente, traçamos uma linha entre entidades associadas.



# Conjuntos de Associações em Diagramas Entidade-Associação (ER)

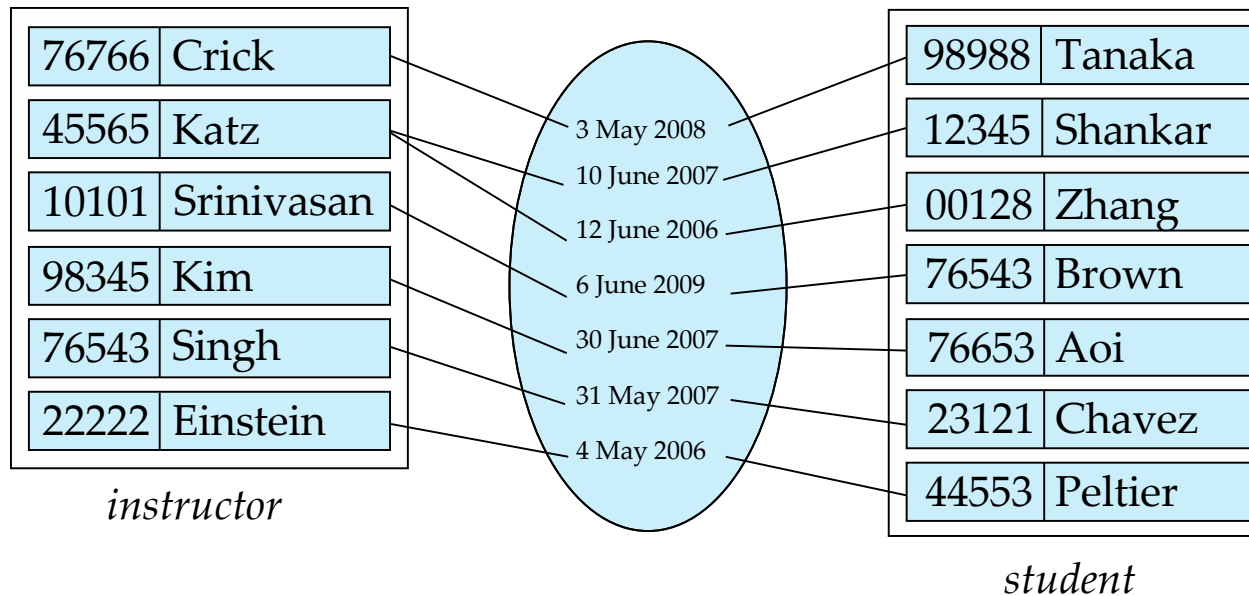
- Os losangos representam conjuntos de associações.



# Conjuntos de associações (Cont.)

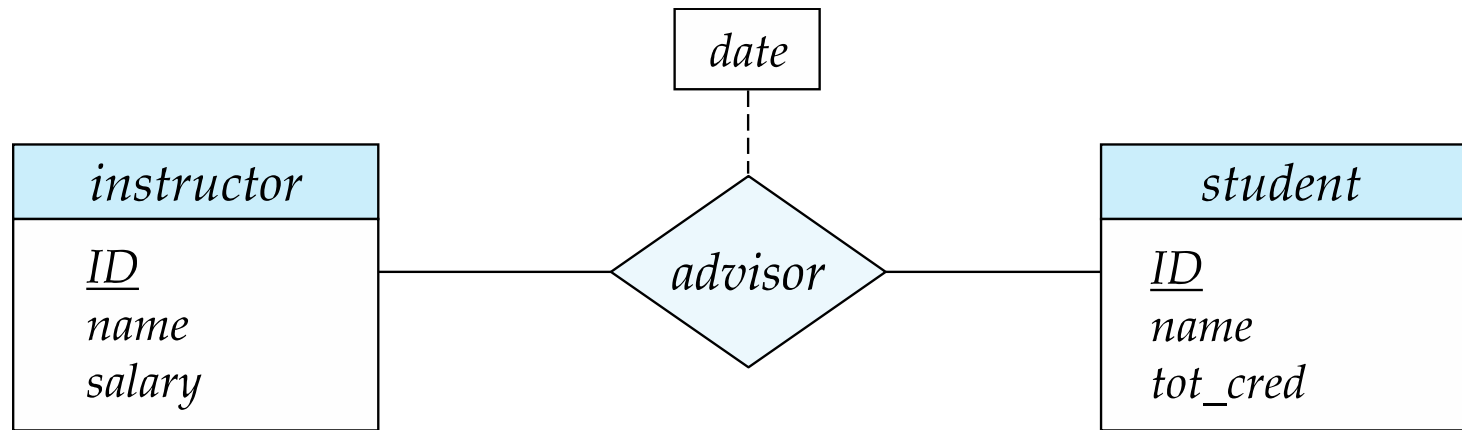
Um atributo também pode ser associado a um conjunto de associações.

Por exemplo, o conjunto de associações *orientador*, definido entre os conjuntos de entidades *instrutor* e *student* pode ter o atributo *date* que regista quando o aluno começou a ser associado ao orientador



# Conjuntos de associações com atributos

---

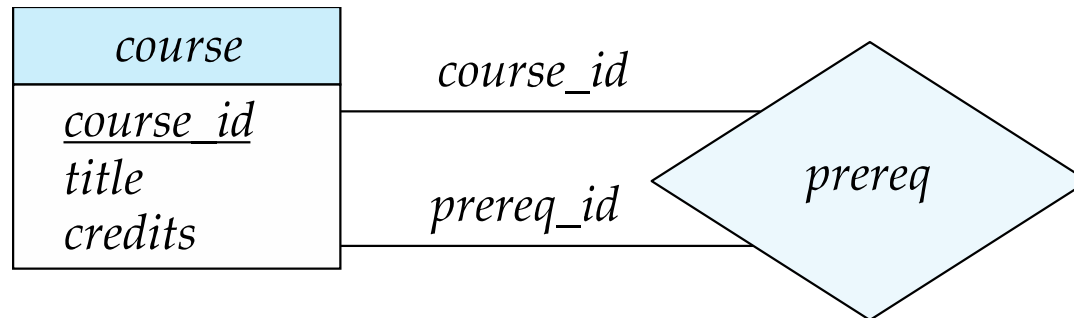


# Papéis

Conjuntos de entidades de um conjunto de associações não precisam de ser distintos

- Cada ocorrência de um conjunto de entidades desempenha um “papel”.

Os rótulos “*course\_id*” e “*prereq\_id*” são chamados de **papéis**.



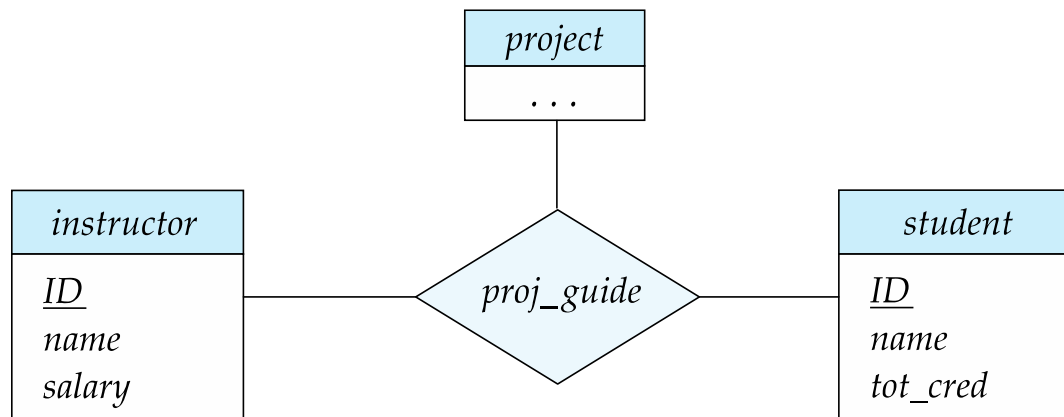
# Grau de um conjunto de associações

---

- Conjuntos de Associações binários
  - envolve dois conjuntos de entidades (grau dois).
  - a maioria dos conjuntos de associações numa base de dados são binários.
- associações entre mais de dois conjuntos de entidades são raras.
  - Exemplo: *os students* trabalham em *projetos de investigação* sob a orientação de um *instrutor*.
  - O conjunto de associações *proj\_guide* é ternário entre *instrutor*, *student* e *project*

# Conjuntos de associações não binários

Diagrama ER com conjunto de associações ternário





# Atributos Complexos

---

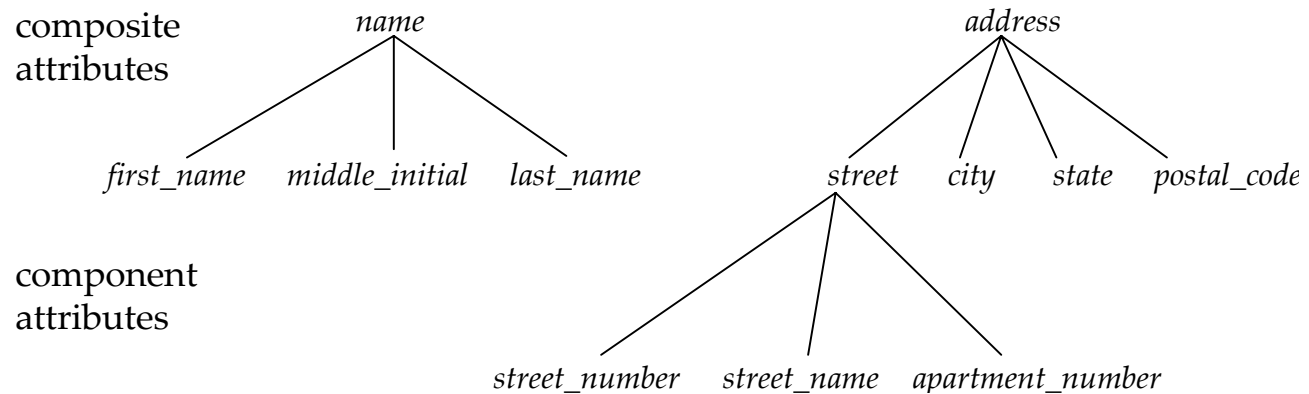
Tipos de atributos:

- Atributos **simples** e **compostos**
- Atributos **de valor único** e **múltiplo-valor**
  - Exemplo: atributo múltiplo-valor: *números\_telefone*
- Atributos **derivados**
  - Podem ser calculados a partir de outros atributos
  - Exemplo: idade a partir da *data\_de\_nascimento*

**Domínio** – o conjunto de valores permitidos para cada atributo

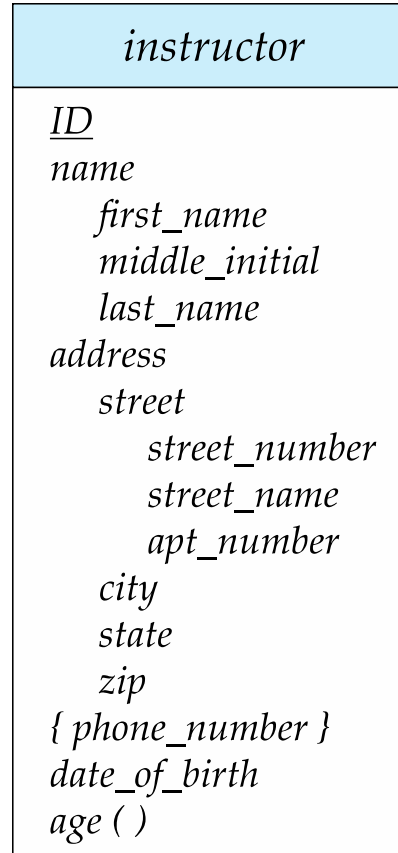
# Atributos compostos

Os atributos compostos permitem dividir os atributos em subpartes (outros atributos).



# Representação de atributos complexos no Diagrama ER

---



# Restrições de cardinalidade

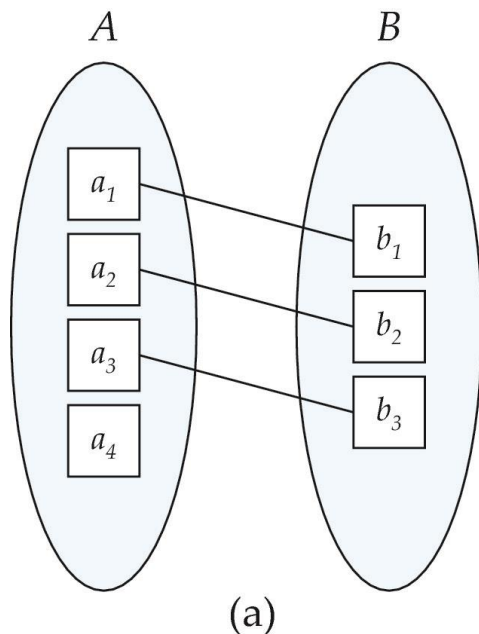
---

Expressam o número de entidades às quais a outra entidade pode ser associada através de um conjunto de associações.

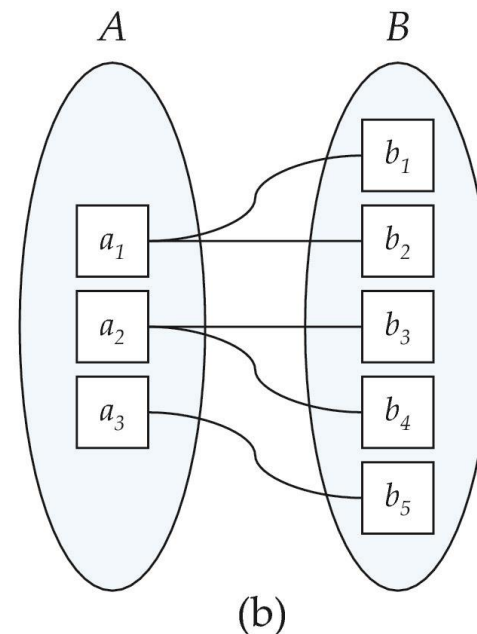
Para um conjunto de associações binárias, a cardinalidade deve ser de um dos seguintes tipos:

- Um para um
- Um para muitos
- Muitos para um
- Muitos para muitos

# Cardinalidades



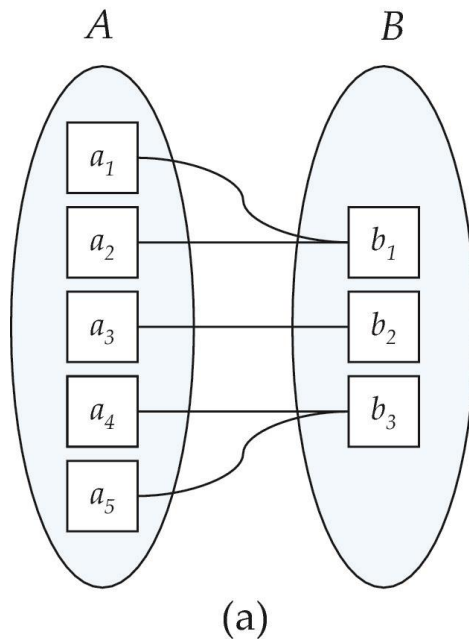
Um para um



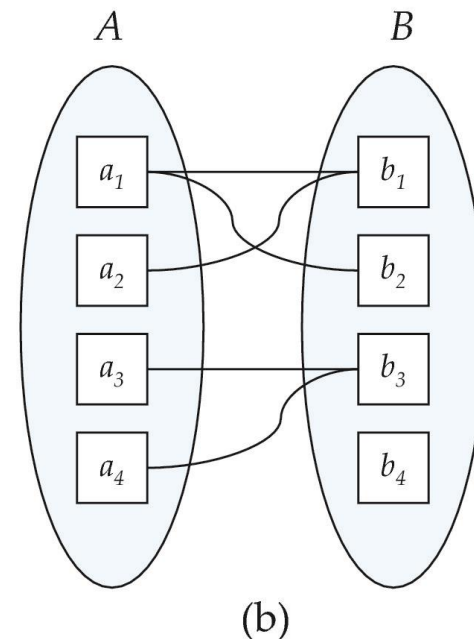
Um para muitos

Nota: Alguns elementos em  $A$  e  $B$  podem não estar associados a nenhum elemento do outro conjunto (veremos mais à frente o conceito de 0 ou 1)

# Cardinalidades



Muitos para um



Muitos para muitos

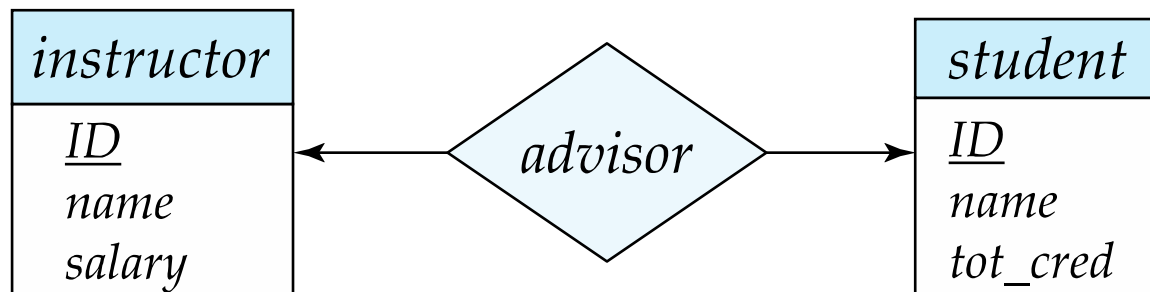
Nota: Alguns elementos em A e B podem não estar associados a nenhum elemento do outro conjunto (veremos mais à frente o conceito de 0 ou 1)

# Representando restrições de cardinalidade no diagrama ER

Expressamos restrições de cardinalidade desenhando uma linha direcionada ( $\rightarrow$ ), significando “no máximo um”, ou uma linha não direcionada ( $—$ ), significando “muitos”, entre o conjunto de associações e o conjunto de entidades.

Conjunto de Associações **um-para-um** entre um *instrutor* e um *aluno* :

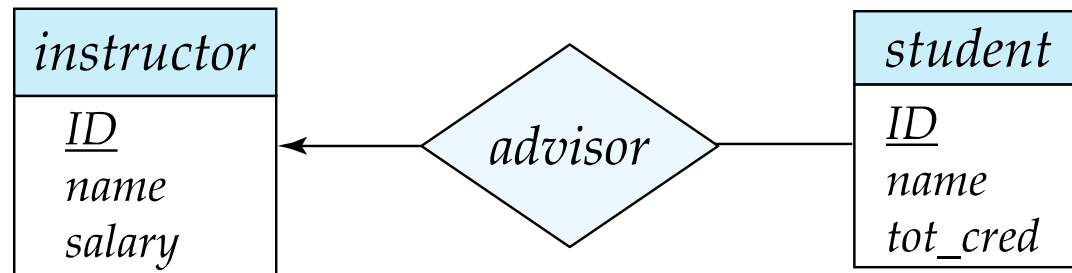
- Um *student* está associado a no máximo um *instrutor* por meio do conjunto de associações *advisor*
- Um *instructor* está associado a no máximo um *student* através do conjunto de associações *advisor*



# Conjunto de Associações um-para-muitos

Conjunto de Associações um-para-muitos entre *instrutor* e *aluno*

- um *instrutor* está associado a vários (incluindo 0) *student* através do conjunto de associações *advisor*
- um *student* está associado a no máximo um *instrutor*, através do conjunto de associações *advisor*

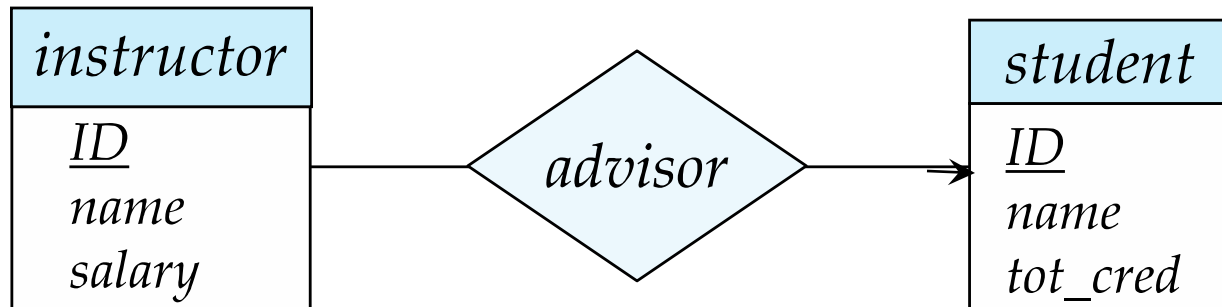




# Conjunto de Associações muitos-para-um

Conjunto de Associações muitos-para-um entre um *instructor* e um *student*,

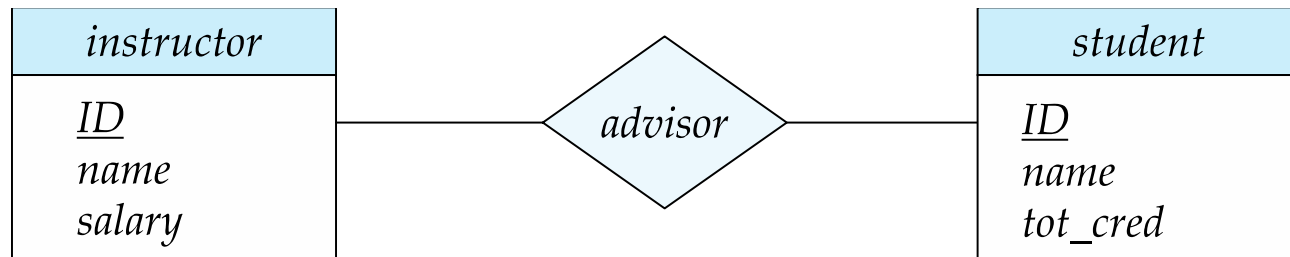
- um *instructor* está associado a no máximo um *student* através do Conjunto de Associações *advisor*
- um *student* está associado a vários (incluindo 0) *instructors* através do Conjunto de Associações *advisor*



# Conjunto de Associações muitos-para-muitos

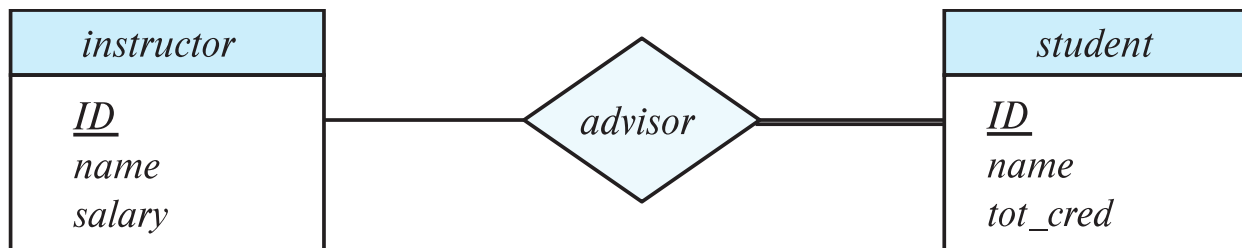
Um instrutor está associado a vários (possivelmente 0) *students* através do Conjunto de Associações *advisor*

Um *student* está associado a vários (possivelmente 0) *instrutores* através do Conjunto de Associações *advisor*



# Participação Total e Parcial

- **Participação total** (indicada por linha dupla): cada entidade do conjunto de entidades participa em pelo menos uma associação no conjunto de associações

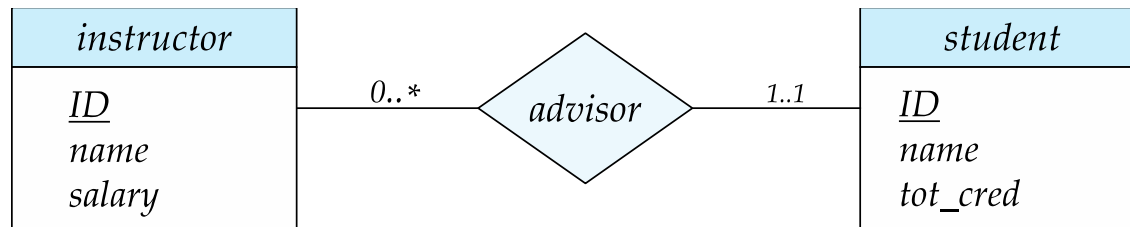


a participação do *student* na associação *advisor* é total

- Todas as entidades de *student* devem ter um instrutor associado no conjunto de associações *advisor* (*student* tem pelo menos um *instructor*)
- **Participação parcial** : algumas entidades podem não participar em nenhuma associação no conjunto de associações
  - Exemplo: a participação do *instructor* no conjunto de associações *advisor* é parcial, indicando que podem haver instrutores sem qualquer *student* associado no conjunto de associações *advisor*

# Notação para expressar restrições mais complexas

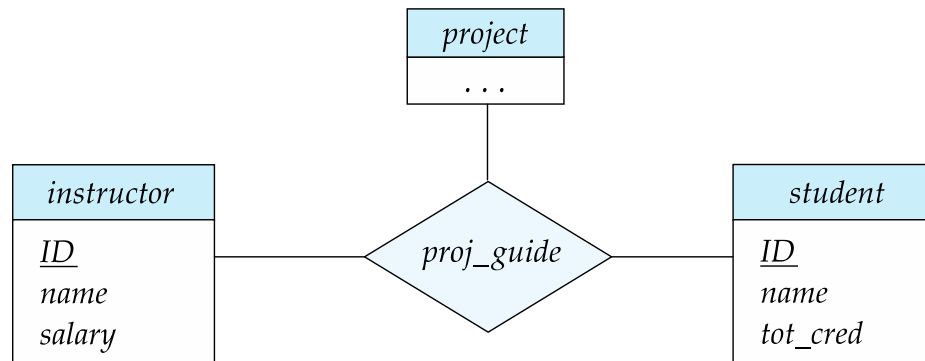
- Uma linha pode ter uma cardinalidade mínima e máxima, representada na forma  $i..h$ , onde  $i$  é o mínimo e  $h$  a cardinalidade máxima
  - Um valor mínimo de 1 indica participação total.
  - Um valor máximo de 1 indica que a entidade participa no máximo numa associação
  - Um valor máximo de \* indica que não há limite.
- Exemplo



- O instrutor pode orientar 0 ou mais alunos. O aluno deverá ter 1 orientador; não pode ter vários vários e tem que ter forçosamente 1

# Restrições de cardinalidade em conjuntos de associações ternário

Permitimos no máximo uma seta num conjunto de associações ternário (ou de grau superior) para indicar uma restrição de cardinalidade



Por exemplo, uma seta de *proj\_guide* para *instructor* indica que cada aluno tem no máximo um orientador para um projeto

# Restrições de cardinalidade em conjuntos de associações ternários

---

Se houvesse mais de uma seta, existem duas formas de definir o significado.

- Por exemplo, uma relação ternária  $R$  entre  $A$ ,  $B$  e  $C$  com setas para  $B$  e  $C$  poderia significar
  1. Cada entidade  $A$  está associada a uma entidade única de  $B$  e  $C$  ou
  2. Cada par de entidades de  $(A, B)$  está associado a uma entidade  $C$  única, e cada par  $(A, C)$  está associado a uma entidade  $B$  única
- Cada alternativa foi usada em diferentes formalismos
- Para evitar confusão, proibimos a utilização de mais do que uma seta em conjuntos de associações ternários

# Chave Primária

---

As chaves primárias fornecem uma forma de especificar como as entidades e as relações se distinguem.

Vamos considerar:

- Conjuntos de entidades
- Conjuntos de associações

# Chave primária para conjuntos de entidades

---

Por definição, entidades individuais são distintas.

Do ponto de vista da base de dados, as diferenças entre elas devem ser expressas em termos dos seus atributos.

Os valores dos atributos de uma entidade devem ser tais que possam identificar a entidade de forma única.

- Não é permitido que duas entidades num conjunto de entidades tenham exatamente o mesmo valor para todos os atributos.

Uma chave para uma entidade é um conjunto de atributos suficientes para distinguir as entidades umas das outras.

Como vimos anteriormente, a chave primária deve ser uma das possíveis chaves candidatas.



# Chave primária para conjuntos de associações

---

Para distinguir entre as várias associações de um conjunto de associações usamos as chaves primárias individuais das entidades no conjunto de associações.

- Seja  $R$  um conjunto de associações envolvendo os conjuntos de entidades  $E_1, E_2, \dots E_n$
- A chave primária para  $R$  consiste na união das chaves primárias dos conjuntos de entidades  $E_1, E_2, \dots E_n$

Exemplo: conjunto de associações “*advisor*”.

- A chave primária consiste em *instrutor.ID* e *aluno.ID*

A escolha da chave primária para um conjunto de associações depende da cardinalidade do mapeamento do conjunto de associações.

# Escolha da chave primária para conjunto de associações binário

---

## ■ Associações muitos-para-muitos

- A união anterior das chaves primárias é uma superchave mínima e é escolhida como chave primária.

## ■ Associações um-para-muitos

- A chave primária do lado “Muitos” é uma superchave mínima e é usada como chave primária.

## ■ Associações muitos para um

- A chave primária do lado “Muitos” é uma superchave mínima e é usada como chave primária.

## ■ Associações um para um

- A chave primária de qualquer dos conjuntos de entidades participantes forma uma superchave mínima e qualquer uma delas pode ser escolhida como chave primária.

# Atributos Redundantes

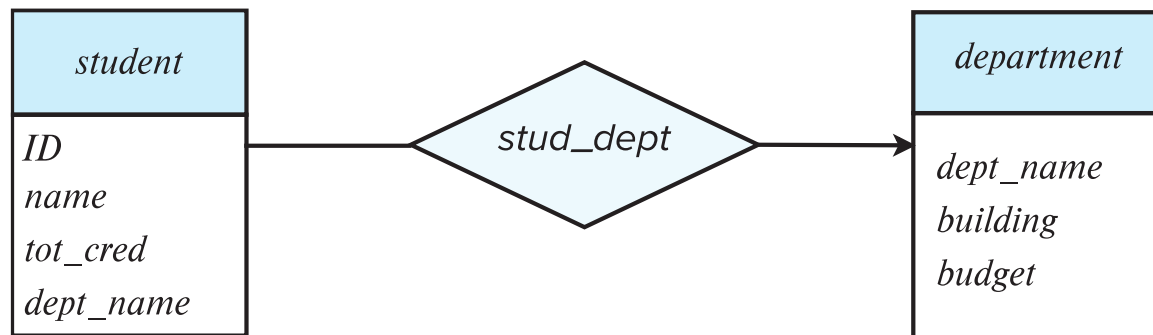
Suponha que temos os conjuntos de entidades:

- *estudante*, com atributos: *ID* , *nome*, *tot\_cred* , *dept\_name*
- *departamento*, com atributos: *dept\_name* , *edifício*, *orçamento*

Modelamos o facto de que cada aluno tem um departamento associado usando um **conjunto de associações** *stud\_dept*

O atributo *dept\_name* em *student* abaixo replica informações presentes na associação e portanto é redundante

- **tem que ser removido**



# Diagrama ER para uma Universidade

