



ORACLE

Academy



Database Foundations

3-4

Terminologia de Modelagem de Dados e Mapeamento

ORACLE
Academy



Roteiro

Mais sobre
Relacionamentos

Rastreando
Alterações
de Dados

Normalização e
Regras
de Negócios

**Terminologia de
Modelagem de
Dados e
Mapeamento**



Parte 3

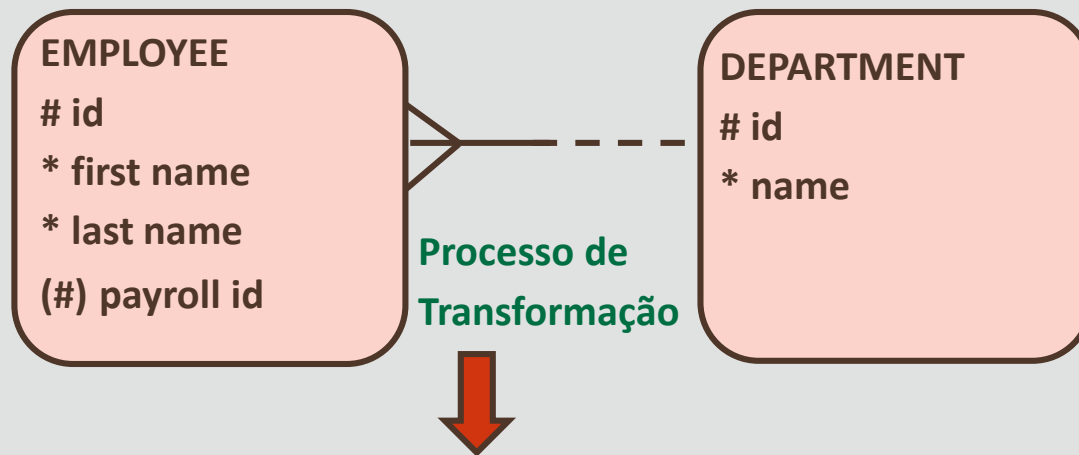
Objetivos

- Esta lição abrange os seguintes objetivos:
 - Aplicar o mapeamento de terminologia entre os modelos lógicos e físicos
 - Entender e aplicar as convenções de nomenclatura Oracle para tabelas e colunas usadas em modelos físicos
 - Aplicar as regras de mapeamento de relacionamentos para transformar relacionamentos corretamente



Transformação de Lógico em Físico Exemplo

Modelo Lógico
(ERD)



EMPLOYEES (EPE)

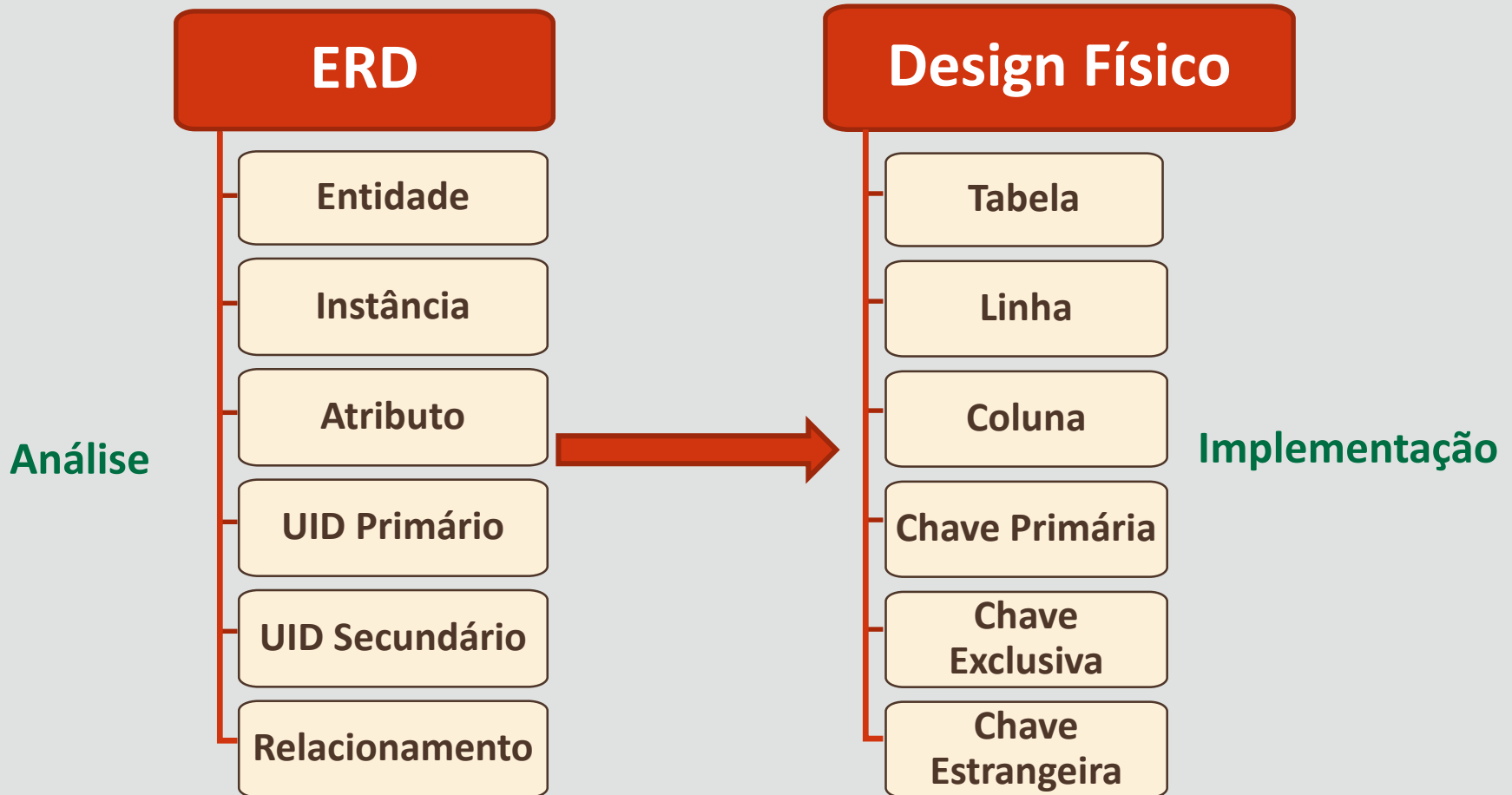
Key Type	Optionality	Column name
pk	*	id
uk	*	payroll_id
	*	last_name
	*	first_name
fk	*	department_id

Implementação física: Banco
de Dados Relacional

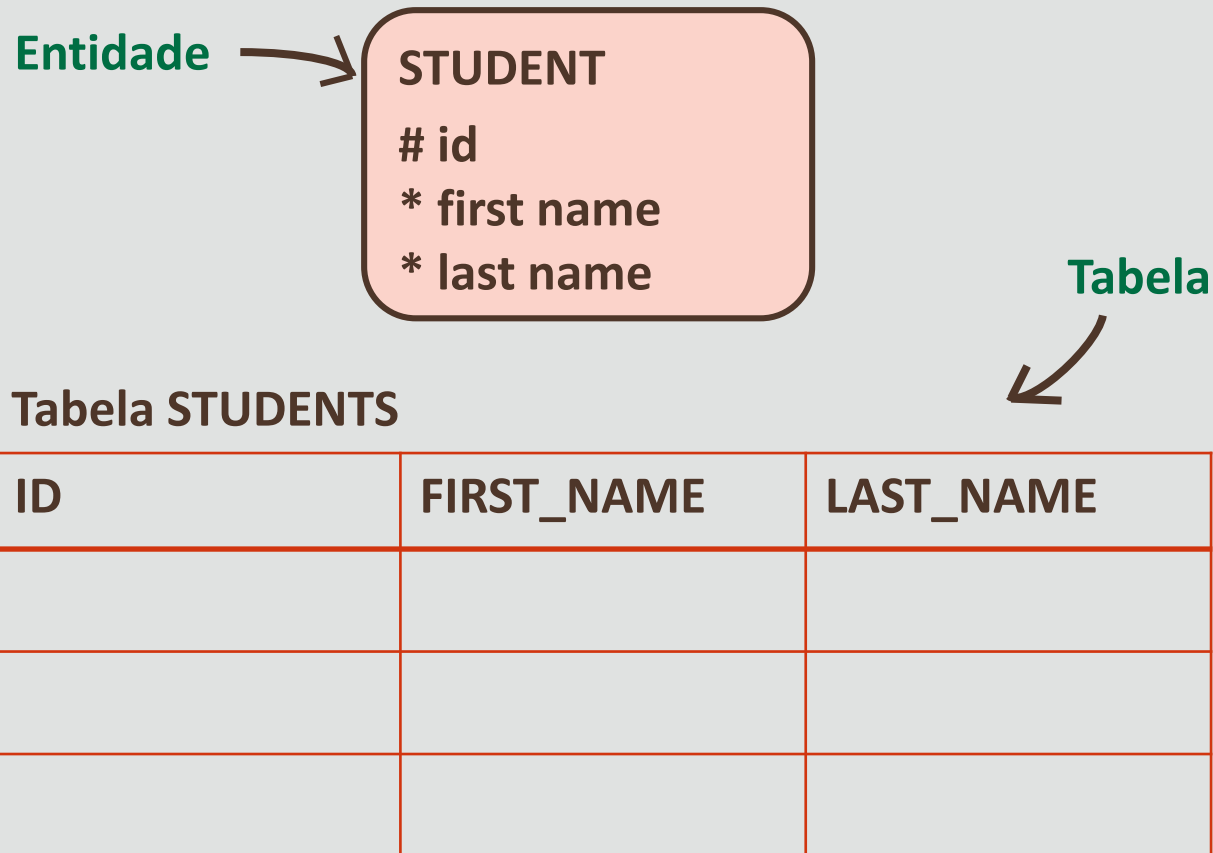
DEPARTMENTS (DPT)

Key Type	Optionality	Column name
pk	*	id
	*	name

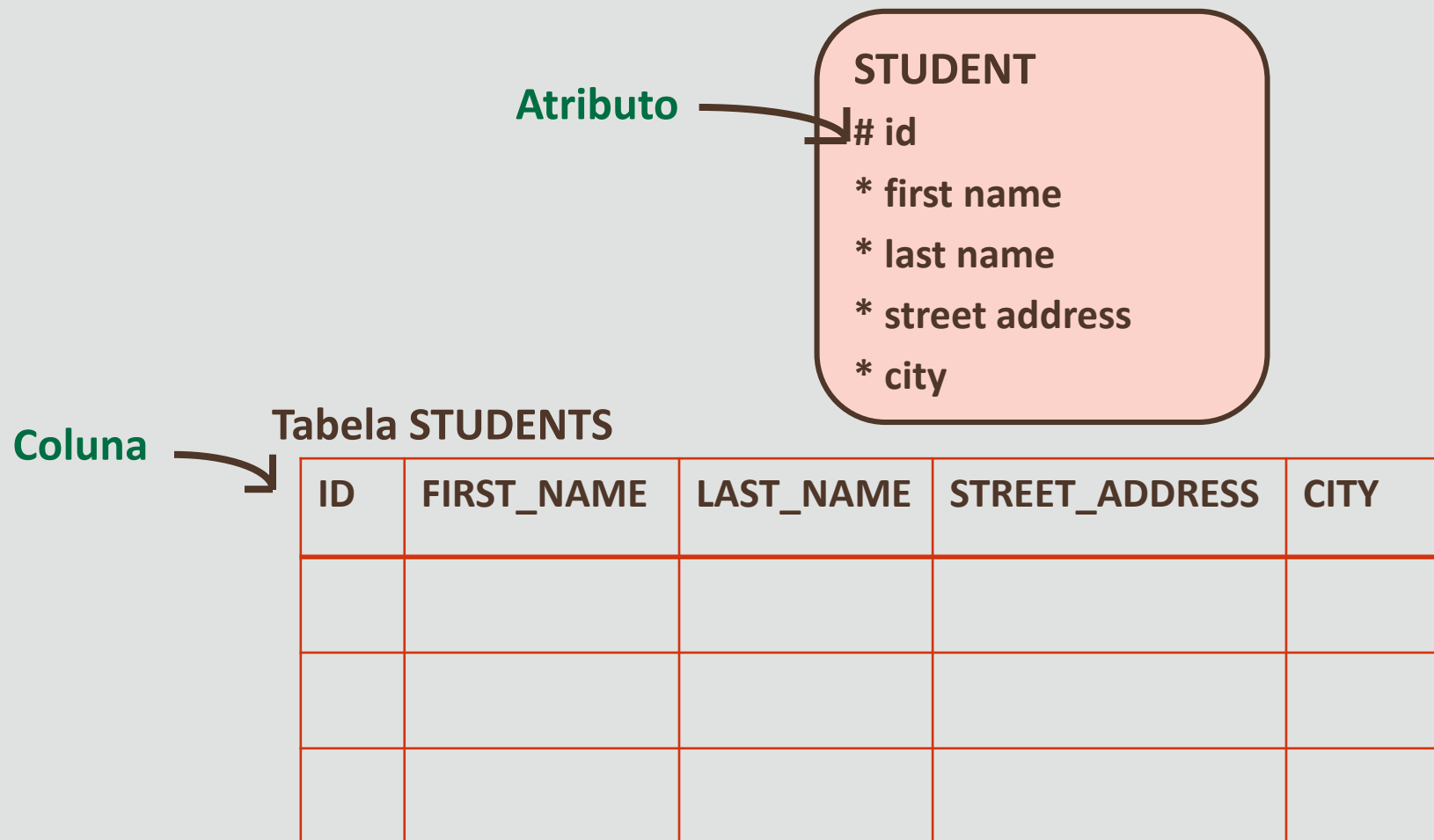
Mapeamento de Terminologia



Uma Entidade e uma Tabela Correspondente



Atributos e Colunas Correspondentes



Uma Instância e uma Linha Correspondente

Entidade

STUDENT

Instância

J Smith

ID	FIRST_NAME	LAST_NAME	STREET_ADDRESS	CITY
101	Sam	Linkin	99B, Chuah Street	LA
102	Neena	Markin	44A, Church Street	NZ
103	Rick	Austina	1 st Cross, Palm Street	SA
104	J	Smith	Alpha Street	CA → Linha



Notações do Diagrama de Tabela

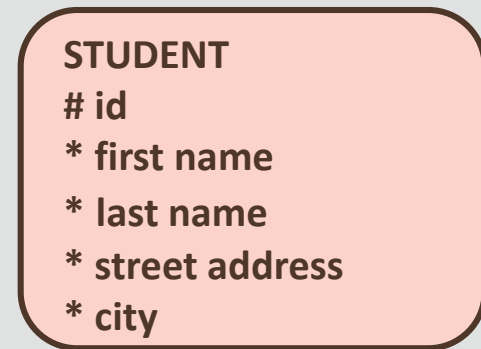
- Um diagrama de tabela é uma documentação adicional que é geralmente usada para explicar com mais detalhes chaves e colunas no banco de dados físico

Tabela **STUDENTS**

Key Type	Optionality	Column Name
pk	*	id
	*	first_name
	*	last_name
	*	street_address
	*	city

Convenções de Nomenclatura para Tabelas

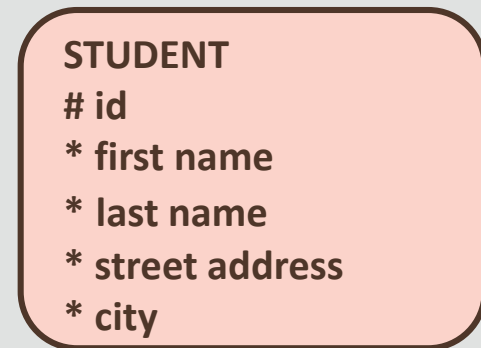
- O nome da tabela é o plural do nome da entidade.
 - Exemplo: STUDENT torna-se STUDENTS
- Os nomes das colunas são idênticos aos dos atributos, sendo que caracteres especiais e espaços são substituídos por sublinhados



STUDENTS		
Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
	*	street_address
	*	city

Convenções de Nomenclatura para Colunas

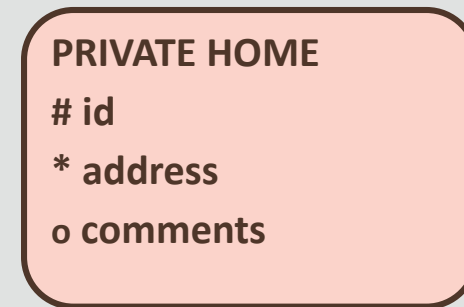
- Os nomes das colunas são semelhantes aos dos atributos, sendo que caracteres especiais e
- espaços são substituídos por sublinhados
- Em geral, os nomes das colunas usam mais abreviações do que os nomes dos atributos
 - Exemplo: First name transforma-se em first_name ou fname



STUDENTS		
Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
	*	street_address
	*	city

Nomes Curtos de Tabelas

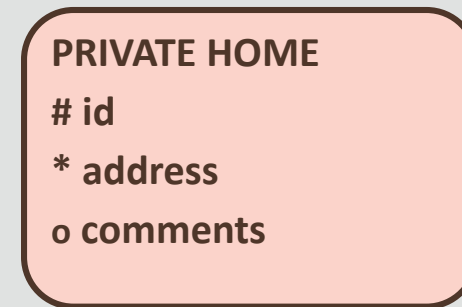
- Um nome curto exclusivo para cada tabela é útil ao nomear colunas de chaves estrangeiras.



PRIVATE_HOMES (PHE)		
Key Type	Optionality	Column name
pk	*	id
	*	address
	o	comments

Nomes Curtos de Tabelas

- Crie nomes curtos com base em:
 - Nomes de entidades que contêm mais de uma palavra
 - Nomes de entidades que contêm uma palavra, mas mais de uma sílaba
 - Nomes de entidades que contêm uma sílaba, mas mais de um caractere



PRIVATE_HOMES (PHE)		
Key Type	Optionality	Column name
pk	*	id
	*	address
	o	comments



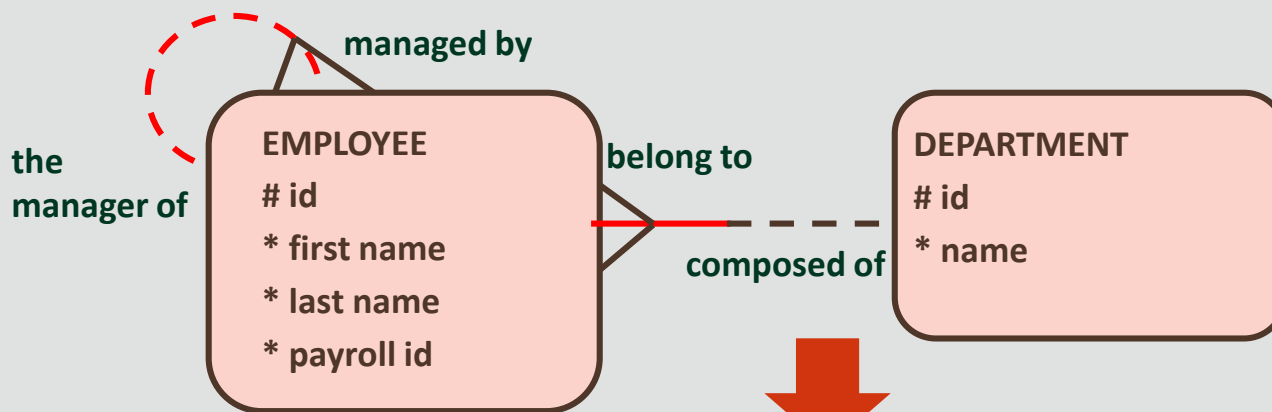
Restrições de Nomenclatura com o Oracle

- Os nomes de tabelas e colunas:
 - Devem começar com uma letra
 - Podem conter até 30 caracteres alfanuméricos
 - Não podem conter espaços, nem caracteres especiais como “!”, mas “\$”, “#” e “_” são permitidos
 - Não podem ser “palavras reservadas” no Oracle DB ou em SQL
- Os nomes de tabelas devem ser exclusivos em uma conta de usuário do banco de dados Oracle
- Os nomes de colunas devem ser exclusivos em uma tabela

Mapeamento de Relacionamentos

- Os relacionamentos são mapeados entre chaves primárias e chaves estrangeiras para permitir que uma tabela faça referência a outra
- Um relacionamento cria uma ou mais colunas de chave estrangeira na tabela, no lado muitos do relacionamento
- Usamos o nome curto da tabela para nomear a coluna de chave estrangeira
- No exemplo da próxima página, a coluna de chave estrangeira na tabela EMPLOYEES é dpt_id para o relacionamento com DEPARTMENT e mgr_id para o relacionamento recursivo com ela mesma

Ilustração do Mapeamento de Relacionamentos



EMPLOYEES (EPE)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
uk	*	payroll_id
fk1	*	dpt_id
fk2	o	mgr_id

DEPARTMENTS (DPT)

Key Type	Optionality	Column name
pk	*	id
uk	*	name

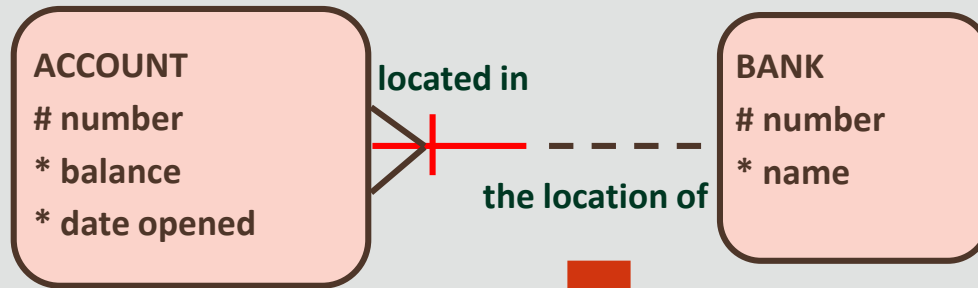
chave estrangeira faz referência a

chave estrangeira faz referência a

Mapeamento de Relacionamentos Associativos

- Um relacionamento associativo é mapeado para uma coluna de chave estrangeira no lado muitos, como qualquer outro relacionamento 1:M
- Nesse caso, a coluna de chave estrangeira desempenha um duplo papel porque também faz parte da chave primária
- No exemplo, bak_number é uma coluna de chave estrangeira em ACCOUNTS que faz referência à chave primária de BANKS
- Ela também faz parte da chave primária de ACCOUNTS

Mapeamento de Relacionamentos Associativos



ACCOUNTS (ACT)

Key Type	Optionality	Column name
pk	*	number
	*	balance
	*	date_opened
pk,fk	*	bak_nbr

BANKS (BAK)

Key Type	Optionality	Column name
pk	*	number
	*	name

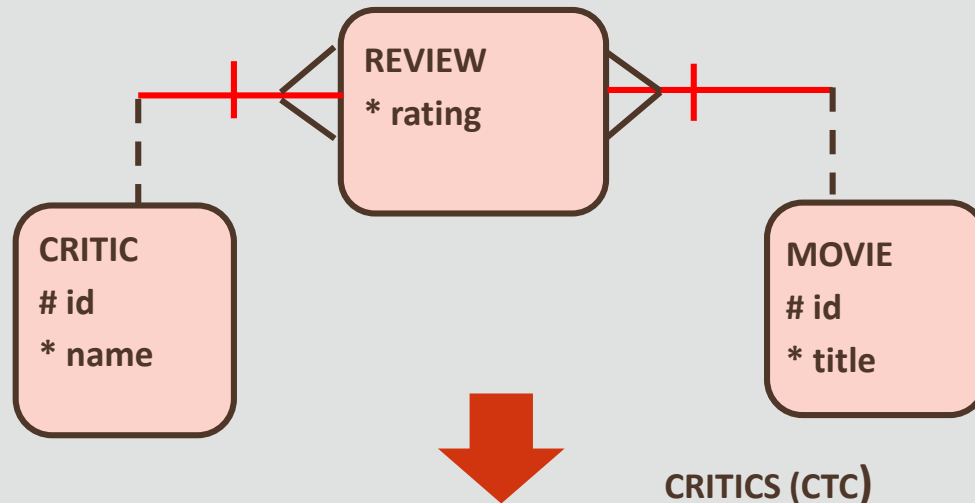
chave estrangeira faz referência a

Mapeando Relacionamentos

Muitos para Muitos

- Um relacionamento M:M é resolvido com uma entidade de interseção, que é mapeada para uma tabela de interseção
- Essa tabela de interseção conterá as colunas de chave estrangeira que fazem referência às tabelas de origem
- No exemplo, REVIEWS contém todas as combinações que existem entre um CRITIC e um MOVIE

Mapeando Relacionamentos Muitos para Muitos



REVIEWS (RVW)

Key Type	Optionality	Column name
pk,fk1	*	ctc_id
pk,fk2	*	mve_id
	*	rating

CRITICS (CTC)

Key Type	Optionality	Column name
pk	*	id
	*	name

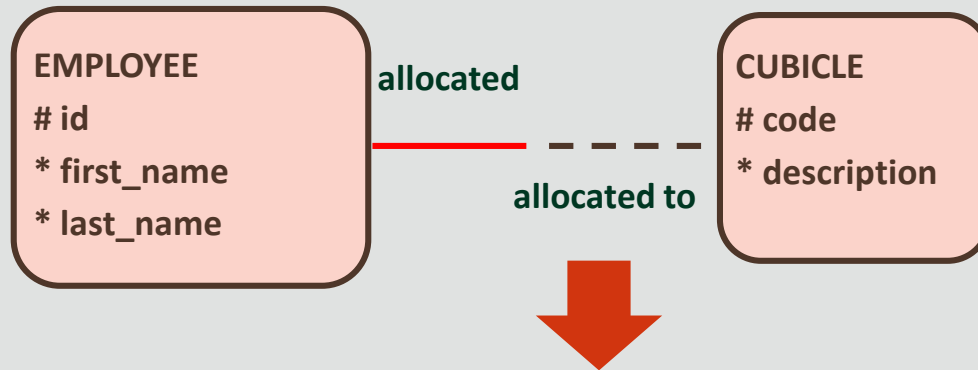
MOVIES (MVE)

Key Type	Optionality	Column name
pk	*	id
	*	title

Mapeando Relacionamentos Um para Um

- Ao transformar um relacionamento 1:1, você cria uma chave estrangeira e uma chave exclusiva
- Todas as colunas dessa chave estrangeira também fazem parte da chave exclusiva
- Se o relacionamento for obrigatório em um lado, a chave estrangeira será criada na tabela correspondente
- No exemplo, cbe_code é uma coluna de chave estrangeira em EMPLOYEES que faz referência à chave primária de CUBICLES
- Cbe_code também seria exclusiva na tabela EMPLOYEES

Mapeando Relacionamentos Um para Um



EMPLOYEES (EPE)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
fk,uk	*	cbe_code

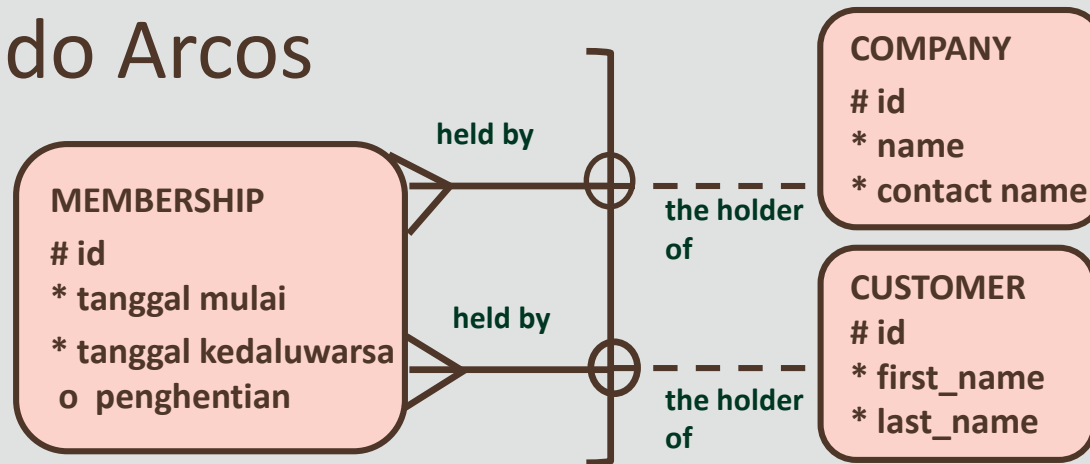
CUBICLES (CBE)

Key Type	Optionality	Column name
pk	*	code
	*	description

Mapeando Arcos

- A entidade que tem o arco será mapeada para uma tabela que contém chaves estrangeiras das tabelas na extremidade “um” dos relacionamentos
- Observe que, mesmo se os relacionamentos do arco forem obrigatórios no lado muitos, as chaves estrangeiras resultantes terão que ser opcionais (porque uma delas estará sempre em branco). Uma restrição de verificação armazenada no banco de dados pode fazer isso facilmente.

Mapeando Arcos



MEMBERSHIPS (MBP)

Key Type	Optionality	Column name
pk	*	id
	*	start_date
	*	expiration_date
	o	termination
fk1	o	cpe_id
fk2	o	cms_id

COMPANIES (CPE)

Key Type	Optionality	Column name
pk	*	id
	*	name
	*	contact_name

CUSTOMERS (CMS)

Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name

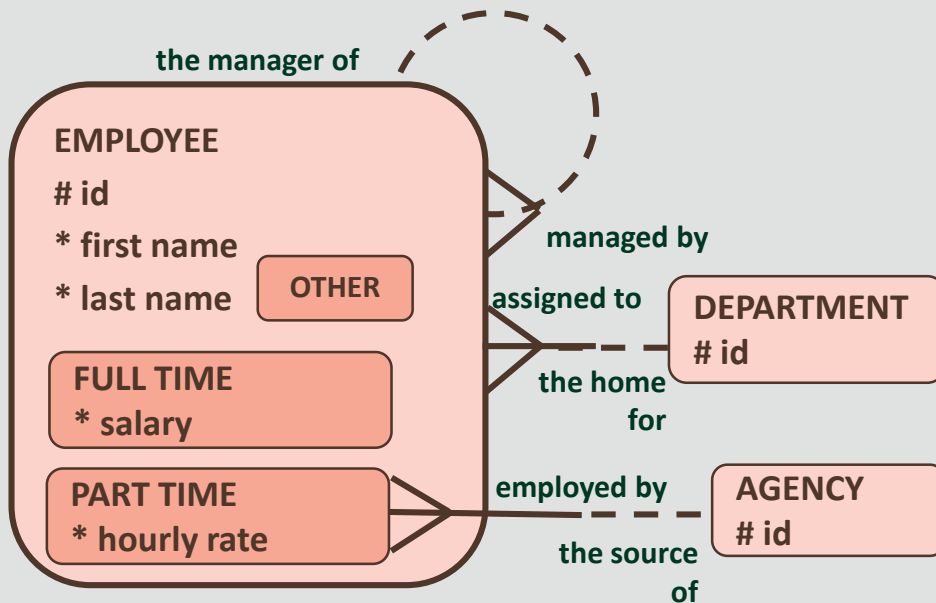
Mapeando Supertipo/Subtipos

- Há várias maneiras de mapear entidades de supertipo/subtipo:
 - Implementação de tabela única: uma tabela é criada independentemente do número de subtipos; usada quando a maioria dos atributos e dos relacionamentos são compartilhados e, portanto, no nível de supertipo
 - Implementação de duas tabelas: uma tabela é criada para cada um dos subtipos (portanto, podem existir mais de duas tabelas); usada quando os subtipos têm pouco em comum e poucos atributos e relacionamentos compartilhados

Implementação de Tabela Única

- A tabela única tem uma coluna para cada atributo do supertipo, com a opcionalidade original do atributo
- A tabela também tem uma coluna para cada atributo pertencente ao supertipo, mas todas as colunas se tornam opcionais
- Além disso, uma coluna obrigatória deve ser criada para atuar como discriminatória, a fim de diferenciar os diferentes subtipos da entidade

Implementação de Tabela Única



DEPARTMENTS (DPT)

Key Type	Optionality	Column name
pk	*	id

AGENCIES (AGY)

Key Type	Optionality	Column name
pk	*	id

EMPLOYEES (EPE)

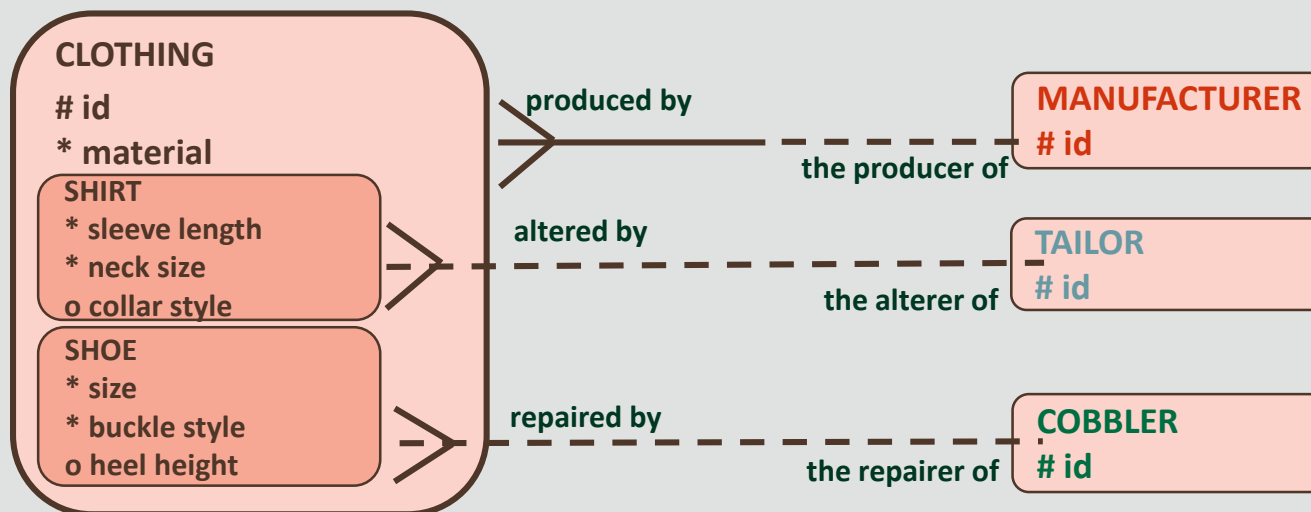
Key Type	Optionality	Column name
pk	*	id
	*	first_name
	*	last_name
	o	salary
	o	hourly_rate
fk1	*	dpt_id
fk2	o	agy_id
	*	epe_type
fk3	o	mgr_id



Implementação de Duas Tabelas

- Uma tabela por subtipo de primeiro nível
- Cada tabela tem uma coluna para cada atributo do supertipo, com a opcionalidade original
- Cada tabela também tem uma coluna para cada atributo pertencente ao supertipo, com a opcionalidade original

Implementação de Duas Tabelas



SHIRTS (SHT)

Key Type	Optionality	Column name
pk	*	id
	*	material
	*	sleeve_length
	*	neck_size
	o	collar_style
fk1	o	tlr_id
fk2	*	mnr_id

SHOES (SHE)

Key Type	Optionality	Column name
pk	*	id
	*	material
	*	size
	*	buckle_style
	o	heel_height
fk1	o	clr_id
fk2	*	mnr_id

refere-se a **tailors**

refere-se a **manufacturers**

refere-se a **cobblers**

Exercício do Projeto

- DFo_3_4_Project
 - Banco de Dados da Oracle Baseball League
 - Aplicar as Regras de Mapeamento de Relacionamentos para Transformar Relacionamentos



Resumo

- Nesta lição, você deverá ter aprendido a:
 - Aplicar o mapeamento de terminologia entre os modelos lógicos e físicos
 - Entender e aplicar as convenções de nomenclatura Oracle para tabelas e colunas usadas em modelos físicos
 - Aplicar as regras de mapeamento de relacionamentos para transformar relacionamentos corretamente





ORACLE

Academy

