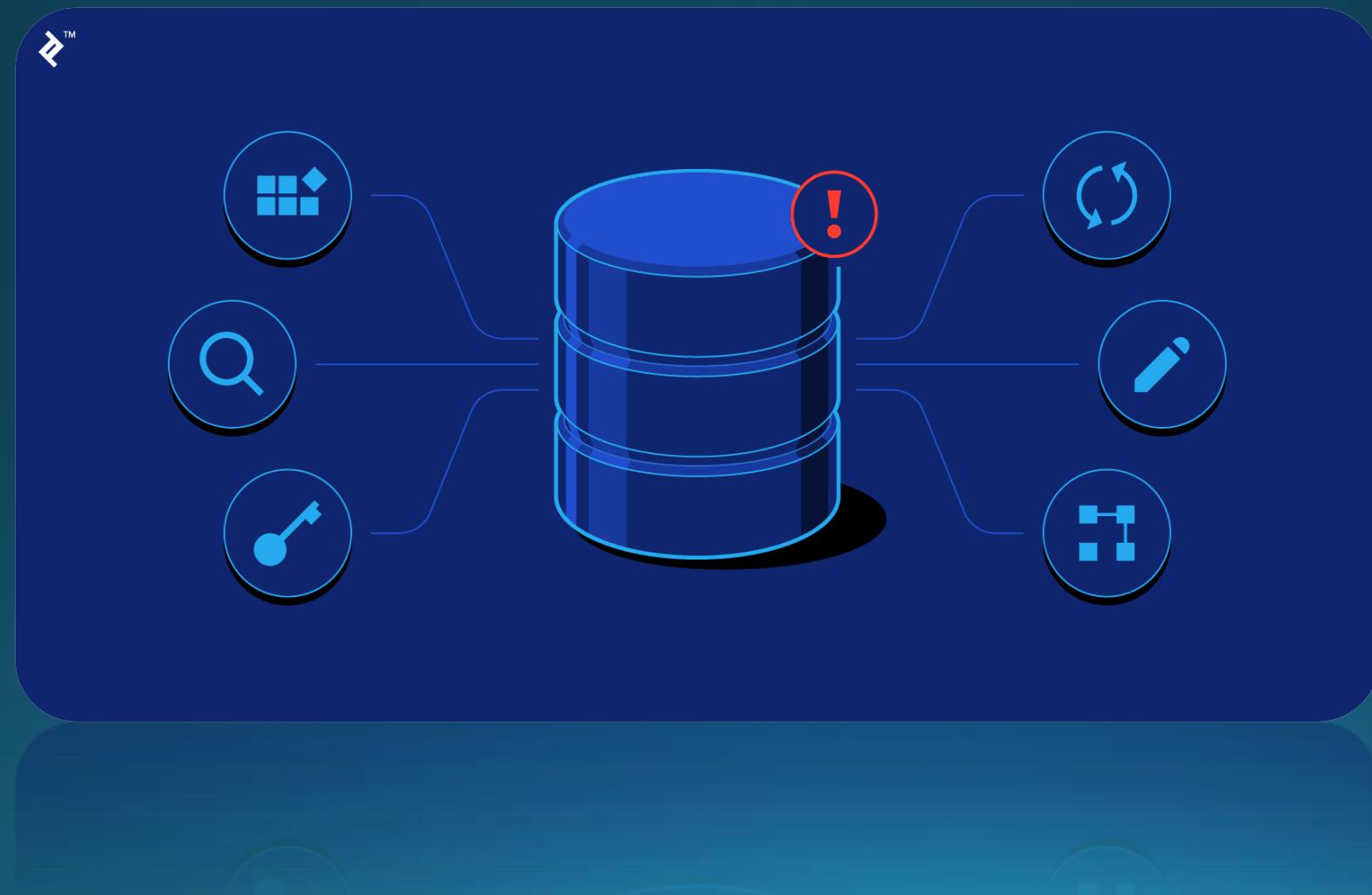


# Banco de Dados





## Ciência de Dados

Você sabia que, ao acessarmos praticamente qualquer dispositivo eletrônico, estamos, de certa forma, inserindo nossos dados e, com o tempo, os dispositivos acabam guardando esses dados, gerando diversas informações sobre nós? Estamos vivendo na era da informação.

# Qual a diferença entre dado e informação?

Pense no nº 25:

INFORMAÇÃO	DADO
Um dado isolado não faz sentido.	Dados devem trazer contexto e complementos informacionais.
Caso não seja seu aniversário ou alguma data especial para você, pode ser visto apenas como um número qualquer.	Se adicionarmos o contexto de 25 de dezembro, podemos associá-lo às festividades natalinas.

# Dados

- São ativos de um empresa.
- Ter um profissional que saiba manipular esse ativo é algo de valor inestimável à empresa.





## O que são Bancos de Dados?

Representa o arquivo físico de dados, armazenado em dispositivos periféricos, onde estão armazenados os dados de diversos sistemas, para consulta e atualização pelo usuário.

# E antes dos Bancos de Dados eletrônicos?

**SEGURANÇA SOCIAL**

**DECLARAÇÃO DE SITUAÇÃO DE DESEMPREGO** (1)

**1 ELEMENTOS DO EMPREGADOR**

Nome (pessoas singulares ou coletivas) \_\_\_\_\_ N.º de identificação de Segurança Social \_\_\_\_\_ N.º de identificação Fiscal \_\_\_\_\_

**2 ELEMENTOS DO TRABALHADOR**

Nome completo \_\_\_\_\_ Data de nascimento \_\_\_\_\_ N.º de identificação de Segurança Social \_\_\_\_\_ N.º de identificação Fiscal \_\_\_\_\_ Data da cessação do contrato de trabalho \_\_\_\_\_

**3 MOTIVOS DE CESSAÇÃO DO CONTRATO DE TRABALHO** (Assinale, apenas, o motivo correspondente)

**Initiativa do empregador**

Justa causa de despedimento por facto imputável ao trabalhador.  
 Despedimento colectivo.  
 Despedimento por extinção do posto de trabalho.  
 Denúncia do contrato no período experimental.  
 Despedimento por inadaptação superiormente ao posto de trabalho.  
 Cessação da comissão de serviço ou situação equiparada, quando não substitua um contrato de trabalho.

**Initiativa do trabalhador**

Resposta com justa causa.  
 Resposta com justa causa por retribuições em mora (salários adeidos).  
 Denúncia do contrato de trabalho/demissão.  
 Denúncia do contrato de trabalho no período experimental.

**Revolução por acordo**

Acordo de revogação por a empresa se encontrar em processo de recuperação previsto no Código da Insolvência e Recuperação, que não é de natureza a impedir a continuação do contrato. Indique o n.º do processo e entidade:  
\_\_\_\_\_

Acordo de revogação com redução de efeitos por a empresa se encontrar em processo de recuperação, com vista ao reforço da qualificação e capacidade técnica da empresa. Indique o diploma aplicável:  
\_\_\_\_\_

**Certificação do empregador**

Assinatura e carimbos \_\_\_\_\_

**CERTIFICAÇÃO DA ENTIDADE COMPETENTE EM MATÉRIA DE INSPEÇÃO DO TRABALHO** (2)

Motivo de cessação do contrato:  O empregador não cumpriu as formalidades previstas no Código do Trabalho  Outro \_\_\_\_\_

(1) Aentrega ao empregador nos termos da artigo 43.º n.º 2, do Decreto-Lei nº 220/2006, de 16 de novembro, que procede à devolução das actas a contar da data do pedido do despedimento.  
(2) A preencher no caso de impossibilidade ou de recusa de emissão por parte do empregador.

Esta declaração pode ser apresentada on-line no sítio da Segurança Social, em [www.sgsocial.pt](http://www.sgsocial.pt)

Modelo RP 904/2013 - DGSS (Página 1 de 1)



# E antes dos Bancos de Dados eletrônicos?

Atributos

Valores

## 1 ELEMENTOS DO EMPREGADOR

Nome (pessoa singular ou colectiva)

N.º de Identificação de Segurança Social

N.º de Identificação Fiscal

## 2 ELEMENTOS DO TRABALHADOR

Nome completo

Data de nascimento

ano	mês	dia
-----	-----	-----

N.º de Identificação de Segurança Social

N.º de Identificação Fiscal

Data da cessação do contrato de trabalho

ano	mês	dia
-----	-----	-----

# Por que utilizar Bancos de Dados?

- Padronização do acesso
- Segurança do acesso
- Integridade das informações (validação / duplicatas)
- Escalabilidade (Possibilidade de expansão)
- Trabalho em equipe (pessoas utilizando ao mesmo tempo)



# Entidade



Representam as estruturas de armazenamento de dados (arquivos) dos sistemas.

# SGBD



Sistema Gerenciador de Banco de Dados é uma coleção de programas que permitem ao usuário definir, construir e manipular bases de dados para as mais diversas finalidades.

# Propriedades

- Um BD é uma coleção de dados logicamente relacionados, com algum significado.

De modo que associações aleatórias de dados não possam ser chamadas de bases de dados.

- Um BD é projetado, construído e preenchido com dados para um propósito específico.

Possui um grupo de usuários e algumas aplicações concebidas para atender a esses usuários .

- Um BD é a representação de algum aspecto do mundo real.

# Visões de BD

Interna: É aquela vista pelo responsável pela manutenção e desenvolvimento do SGBD. Existe a preocupação com a forma de recuperação e manipulação dos dados dentro do Banco de Dados.

# Visões de BD

Externa: É aquela vista pelo usuário que opera os sistemas aplicativos, através de interfaces desenvolvidas pelo analista (programas), buscando o atendimento de suas necessidades

# Visões de BD

**Conceitual**: É aquela vista pelo analista de desenvolvimento e pelo administrador das bases de dados. Existe a preocupação na definição de normas e procedimentos para manipulação dos dados, para garantir a sua segurança e confiabilidade, o desenvolvimento de sistemas e programas aplicativos e a definição no banco de dados de novos arquivos e campos.

# Profissionais



## Administrador de BD:

Responsável por autorizar o acesso ao banco de dados e coordenar e monitorar seu uso.

# Profissionais

Projetista de BD:

responsável pela identificação dos dados a serem armazenados no banco de dados; por escolher estruturas apropriadas para representar e armazenar tais dados.



# Profissionais

## Usuários Finais:

Profissionais que precisam ter acesso ao banco de dados para inserir, consultar, modificar e remover dados.



# Profissionais

## Analistas de Sistemas e Desenvolvedores de Software:

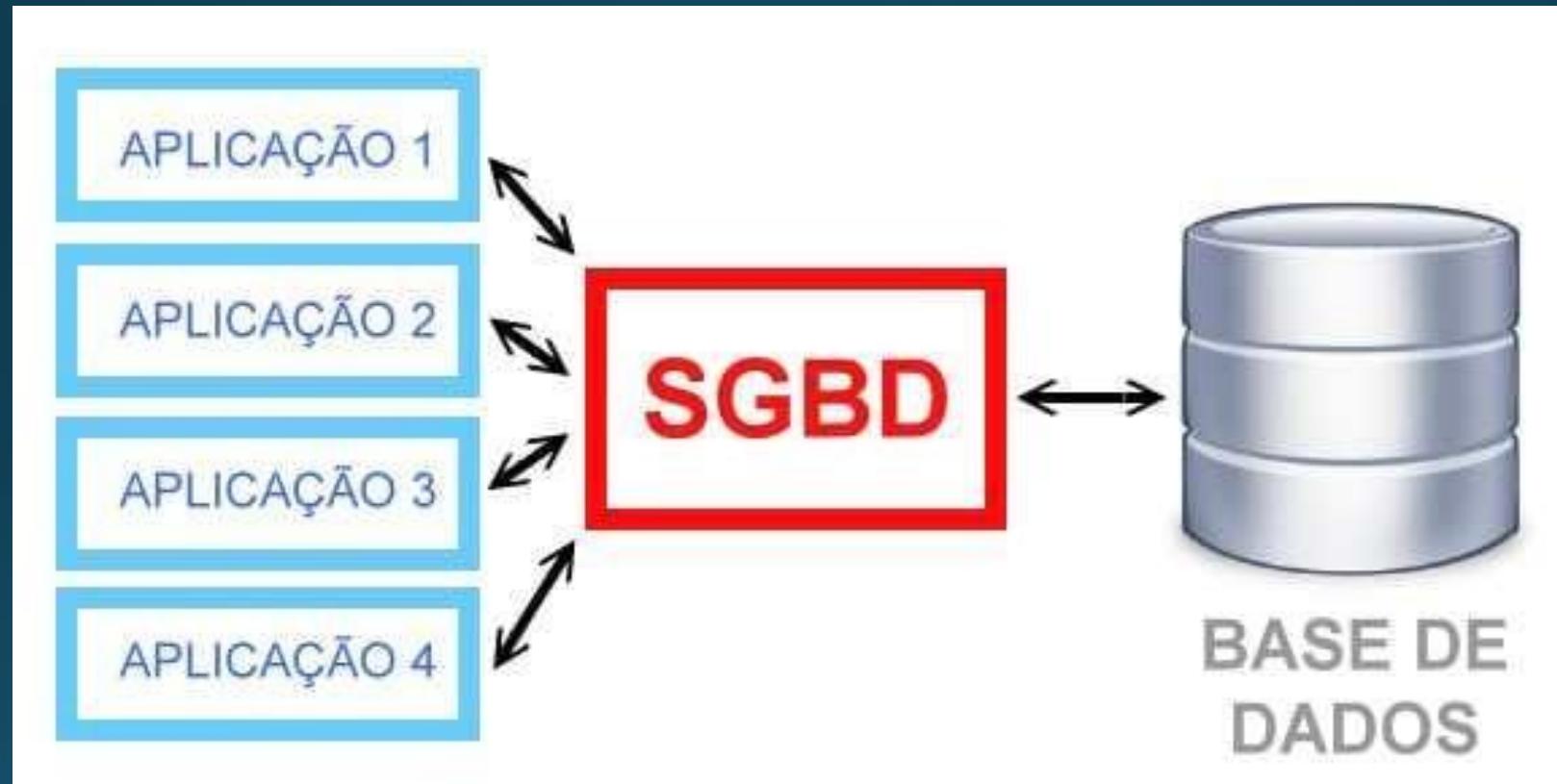
Os analistas desenvolvem especificações das transações que atendem aos requisitos dos usuários, e os programadores implementam estas especificações produzindo os programas (softwares). Devem estar familiarizados com todas as capacidades do SGBD

# SGBDs

- Cada programa:
- “vê” apenas os dados que lhe interessam;
- Não precisam entrar em detalhes de como seus dados estão fisicamente gravados;
- Não precisa ser modificado se a estrutura de dados que ele não utiliza for mudada.



# SGBDs



# Propriedades de SGBDs



## 1. Consistência de Dados:

- O SGBD avalia os dados recebidos através de regras de integridade, garantindo que sempre estejam corretos.

# Propriedades de SGBDs



## 2. Validade dos registros:

- Os dados são válidos quando pertencem ao domínio de valores possíveis naquele caso.

# Propriedades de SGBDs



Nome	Endereço	Telefone
José de Almeida	R. Tulipas 89	0162761112
Jose da Silva	Av. Clóvis Penteado 98	Residencial
Aparecido Navarro	R. Marechal Deodoro	0192445618

Dado inválido !

# Propriedades de SGBDs



## 3. Disponibilidade:

- Todos os dados que precisam ser conhecidos devem estar disponíveis sempre que necessário.

# Propriedades de SGBDs



## 4. Consistência:

- Sempre que a mesma informação é gravada, mesmo que em locais diferentes, ela tem o mesmo valor.

# Principais SGBDs



# Principais SGBDs



Faça agora uma pesquisa sobre os SGBDs apresentados no slide anterior.  
Essa pesquisa deverá ser confeccionada no Word e deve conter:

- Capa;
  - Sumário (Hierarquia de títulos);
  - Imagens (legendas e índice de Imagens);
  - Nº de páginas;
  - Pelo menos uma página para cada SGBD;
  - Uma tabela exibindo o tipo de licença de cada SGBD;
  - Bibliografia;
  - Fonte: Times New Roman (tamanho: 12).
- \* Não utilize a Wikipedia e o ChatGpt como fontes de dados;.

# Modelagem de BD

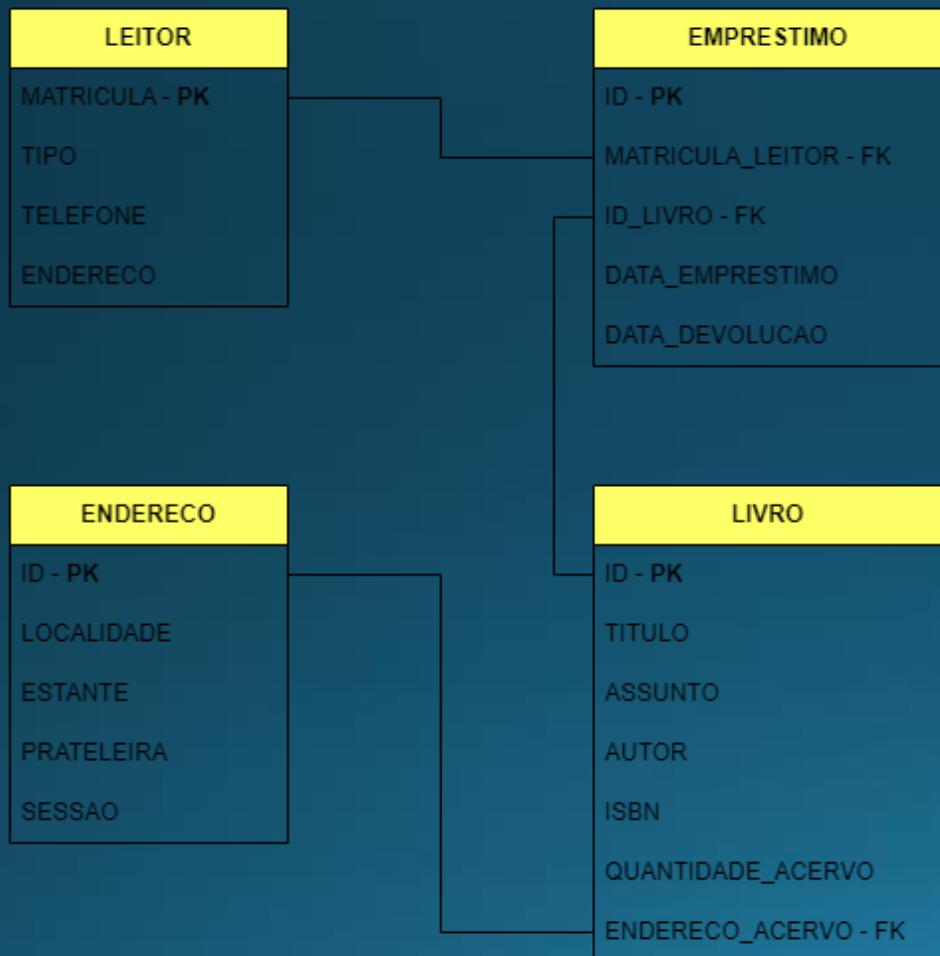
Como você modelaria um banco de dados para biblioteca do SENAI?



1. A biblioteca deseja manter um cadastro de seus leitores.
2. Sobre cada leitor, é importante manter sua matrícula, telefone, endereço e lista de livros que este já pegou emprestado.
3. Um leitor pode emprestar vários livros. Um título pode ser emprestado a mais de um leitor, pois geralmente há mais de um livro do mesmo título na prateleira.
4. Por falar em prateleira, é importante para o bibliotecário, que as estantes e prateleiras tenham endereço para sabermos em que local o livro se encontra. Ex: estante: 3, prateleira 2, nicho: 1.
5. Um leitor pode ser um aluno ou funcionário.
6. Deve-se manter um cadastro sobre cada livro na Biblioteca. Para cada livro é imprescindível armazenar o nome do autor, assunto, editora, ISBN e a quantidade de livros no acervo.

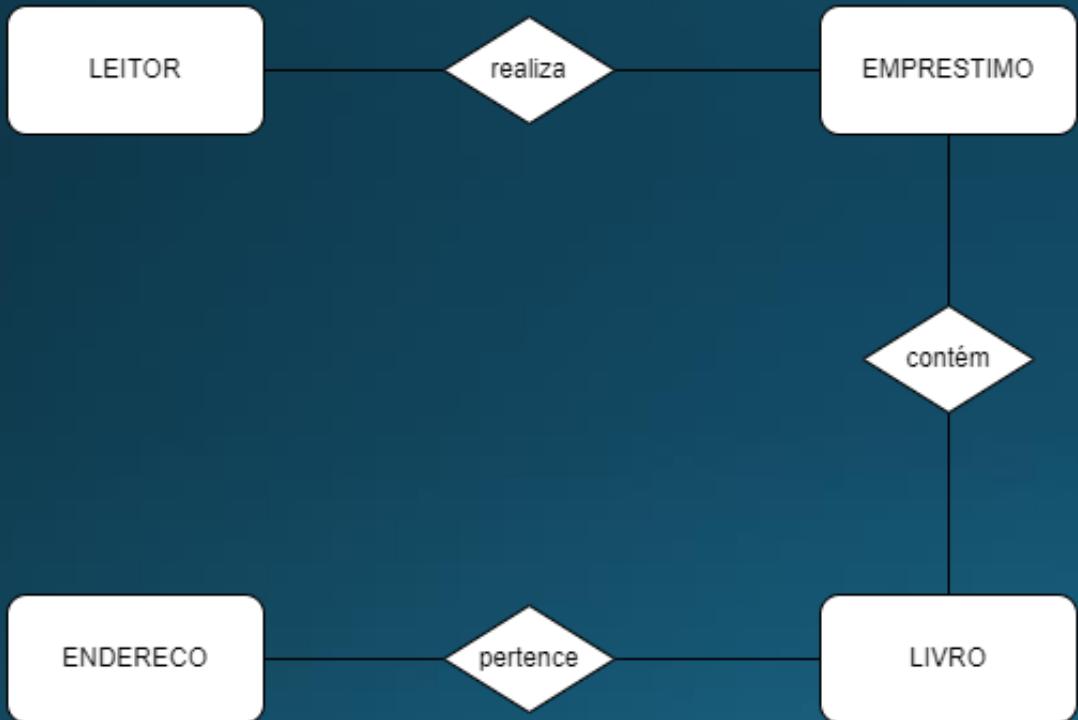
# Modelagem de BD

## Modelo E-R



# Modelagem de BD

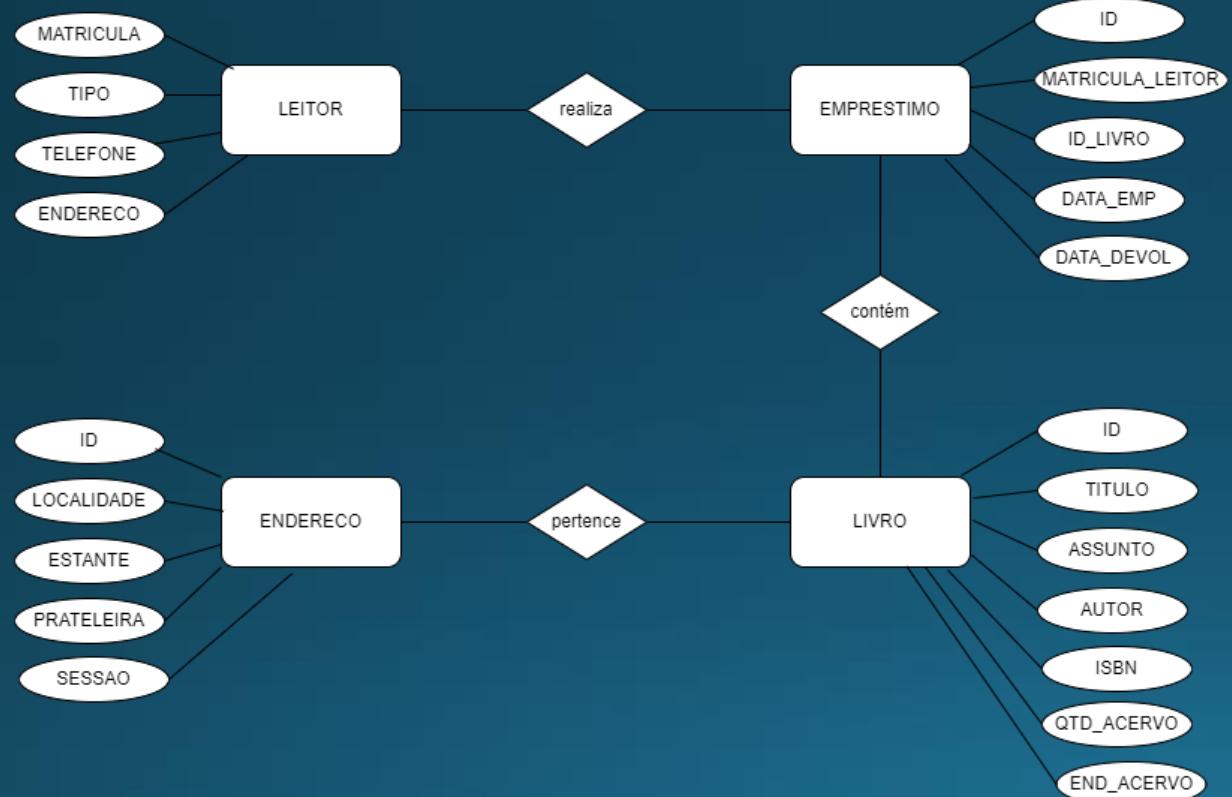
## Diagrama E-R



- Relacionamentos:
  - a) Um Usuário efetua vários Empréstimos,
  - b) Vários Empréstimos contêm vários Livros,
  - c) Vários Livros pertencem a uma Sessão.

# Modelagem de BD

## Diagrama E-R

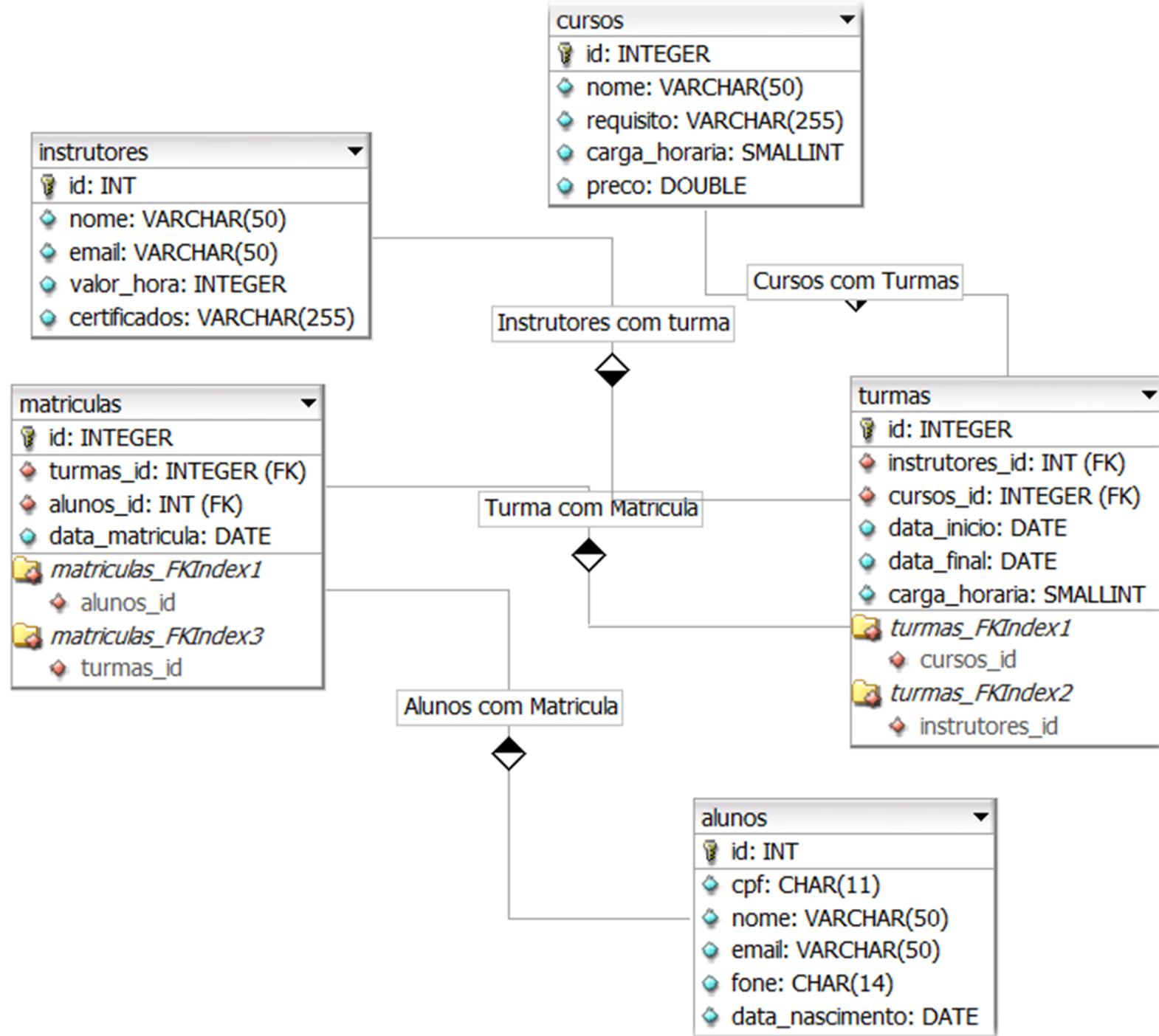


# Criar um modelo de banco de dados para um sistema de escola.



## **Requisitos principais do sistema:**

- Gerenciar informações dos alunos (nome, data de nascimento, matrícula...)
- Gerenciar informações dos professores (nome, especialidade, etc...)
- Gerenciar informações das disciplinas (nome, código, etc...)
- Gerenciar turmas, incluindo qual professor leciona qual disciplina para qual turma.
- Gerenciar matrículas dos alunos nas turmas.
- Registrar notas dos alunos nas disciplinas.





SQL

# SQL

Structured Query Language



*Serviço Nacional de Aprendizagem Industrial*  
PELO FUTURO DO TRABALHO

- A linguagem SQL é utilizada para a realização de consultas em dados armazenados em um banco de dados.
- SQL permite a manipulação, de forma simples e poderosa, dos dados presentes em um Sistema de Gerenciamento de Banco de Dados (SGBD).

# **SQL** - Structured Query Language

Pode ser dividida em subgrupos:

- **DML** - Linguagem de manipulação de dados ou data manipulation language;
- **DDL** - Linguagem de definição de dados ou data definition language;
- **DCL** - Linguagem de controle de dados ou data control language;
- **DTL** - Linguagem de transação de dados ou data transaction language;

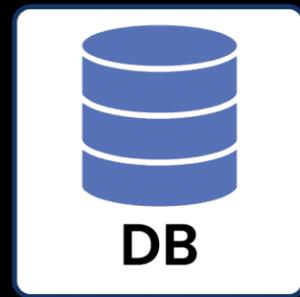
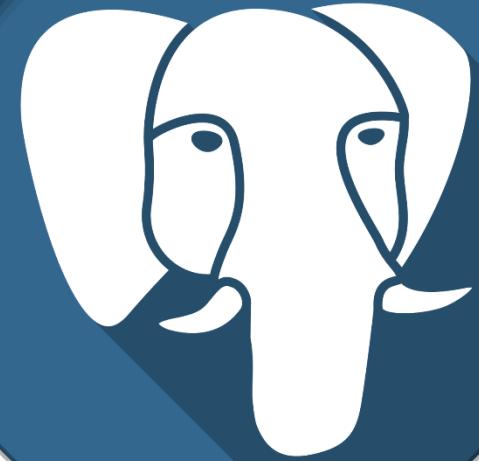


**SENAI**

*Serviço Nacional de Aprendizagem Industrial*  
PELO FUTURO DO TRABALHO

 PostgreSQL

# SQL



DML

- Select
- Insert
- Update
- Delete

DDL

- Create
- Alter
- Drop
- Truncate

DCL

- Grant
- Revoke

DTL

- Save Point
- Roll Back
- Commit

# SQL - DDL



A linguagem de definição de dados ou data definition language (**DDL**) é a encarregada de trabalhar com as tabelas do Sistema de Gerenciamento de Banco de Dados e suas propriedades; portanto, seu uso contempla chaves primárias, por exemplo.

# SQL - DDL

Principais comandos:

**DROP TABLE** (Exclusão de tabelas)

**CREATE TABLE** (Criação de tabelas)

**ALTER TABLE** (Alteração em tabelas)

# **SQL - DML**



A linguagem **DML** (Data Manipulation Language) nasce para respaldar o contexto de manipulação de dados em bancos de dados, principalmente em sistemas de informação que fazem consulta em bancos de dados relacionais por meio de SGBDs (Sistemas de Gerenciamento de Bancos de Dados).

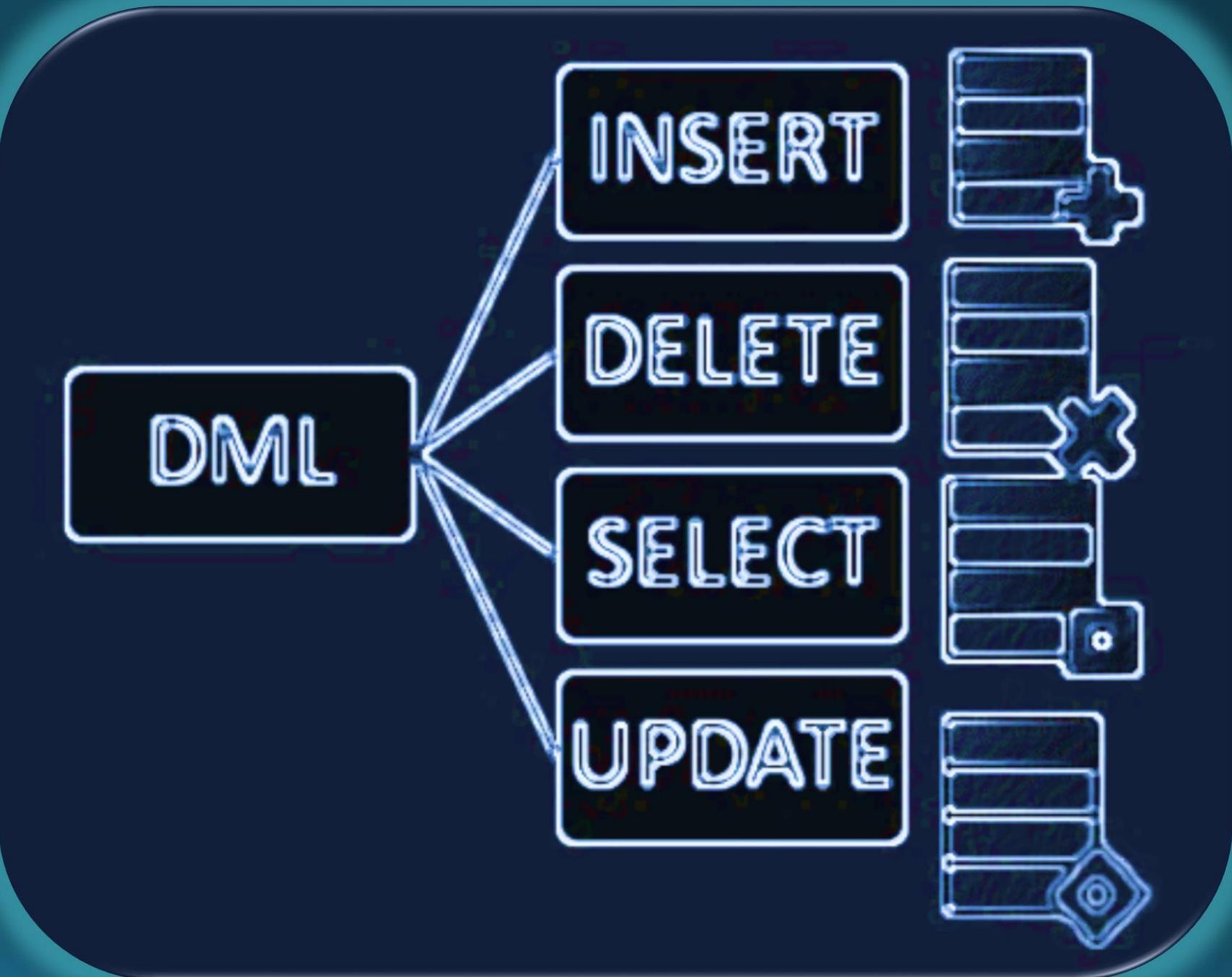


# SQL - DML

A linguagem DML também é utilizada no contexto de sistemas de informação como uma linguagem verbal, em que um sistema de informação externo ao relacionamento entre SGBD e banco de dados faz uma averbação da instrução que necessita, isto é, utiliza um verbo para demonstrar a ação desejada. Os verbos DML, por assim dizer, seriam:

**SELECT, INSERT, UPDATE e DELETE**

# SQL DML





# SQL - INSERT

A partir da coleção de dados nos schemas de um banco de dados, novos dados podem ser inseridos (**INSERT**) nos campos relacionais de uma tabela. Essa inserção ocorre por meio do comando: “**INSERT INTO tabela (...) values (...)**”, onde, no primeiro parêntese, temos os atributos e, no segundo parêntese, temos os valores.

# SQL - INSERT



```
INSERT INTO CAMPEONATO (selecao, atacante,  
meio_campo, lateral, zagueiro) VALUES ('Brasil',  
'Ronaldo', 'Kaká', 'Roberto Carlos', 'Lucio');
```

**Quadro 1.** Composição da tabela com os nomes dos jogadores

Seleção	Atacante	Meio-campo	Lateral	Zagueiro
Brasil	Ronaldo	Kaká	Roberto Carlos	Lúcio

# SQL - INSERT

Caso tenhamos a inserção de várias seleções como, por exemplo, em um banco de dados de convocações da Copa do Mundo, faríamos diversos **INSERTs** para incluir as seleções que participariam da disputa:

```
INSERT INTO CAMPEONATO(Seleção, Atacante, Meio_campo,  
Lateral, Zagueiro) VALUES ('Brasil', 'Neymar', 'Kaká',  
'Roberto Carlos', 'Lúcio');  
  
INSERT INTO CAMPEONATO(Seleção, Atacante, Meio_campo,  
Lateral, Zagueiro) VALUES ('Argentina', 'Messi', 'Di Maria',  
'Gabriel Mercado', 'Mascherano');  
  
INSERT INTO CAMPEONATO(Seleção, Atacante, Meio_campo,  
Lateral, Zagueiro) VALUES ('Espanha', 'Juan', 'Xavi',  
'Iniesta', 'Lavi');  
  
INSERT INTO CAMPEONATO(Seleção, Atacante, Meio_campo,  
Lateral, Zagueiro) VALUES ('Portugal', 'Cristiano Ronaldo',  
'Émerson', 'João', 'Davi');
```

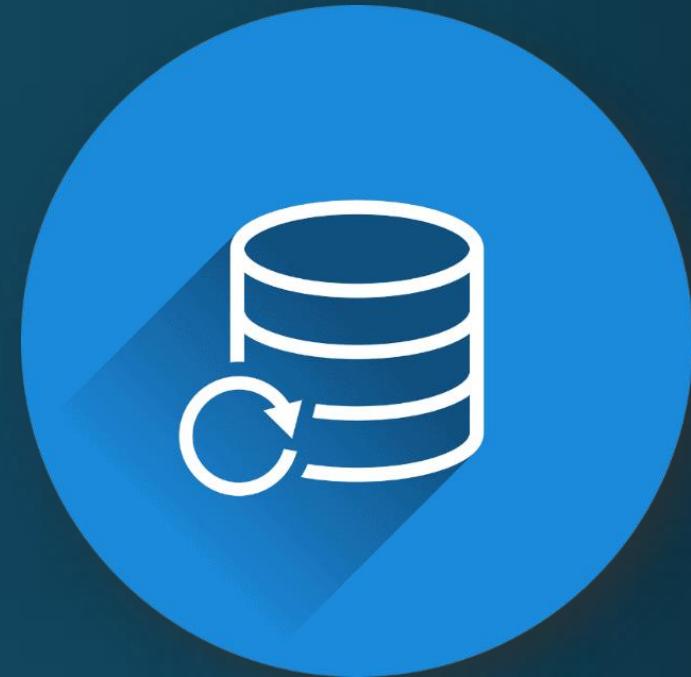


# SQL - UPDATE

Após a tabela estar populada, poderíamos utilizar a instrução DML ‘UPDATE’ para realizar a atualização dos dados inseridos na tabela. Por lógica, então, devemos, primeiramente, inserir dados para que estes possam ser atualizados.



# SQL - UPDATE



**UPDATE** Tabela **SET** nome\_do\_campo  
**WHERE** condição;

# SQL - UPDATE

**UPDATE** CAMPEONATO

**SET** atacante = 'Neymar'

**WHERE** seleção = 'Brasil';



Quadro 2. Atualização da tabela com a instrução UPDATE

Seleção	Atacante	Meio-campo	Lateral	Zagueiro
Brasil	Neymar	Kaká	Roberto Carlos	Lúcio

# SQL - SELECT

Depois de inserir (**INSERT**) e atualizar (**UPDATE**) os dados, poderemos selecioná-los, isto é, realizar a consulta a esses dados, que, de forma geral, é a principal função dos sistemas de informação. Ou seja, um sistema de informação, ao realizar o **SELECT**, consulta o dado na base de dados.

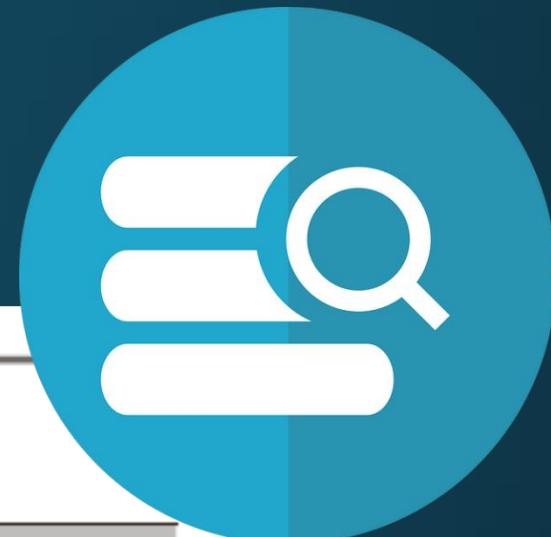


# **SQL - SELECT**

**SELECT** campo(s)  
**FROM** tabela(s)  
**WHERE** condição



# SQL - SELECT



Quadro 3. Base de dados CAMPEONATO montada com os nomes dos jogadores

Seleção	Atacante	Meio-campo	Lateral	Zagueiro
Brasil	Neymar	Kaká	Roberto Carlos	Lúcio
Argentina	Messi	Di Maria	Gabriel Mercado	Mascherano
Espanha	Juan	Xavi	Iniesta	Lavi
Portugal	Cristiano Ronaldo	Émerson	João	Pepe

# SQL - SELECT

Exemplo:

**SELECT** lateral, zagueiro  
**FROM** CAMPEONATO  
**WHERE** seleção = 'Argentina'



**Quadro 4.** Tabela CAMPEONATO com a instrução SELECT de alguns dados

Seleção	Lateral	Zagueiro
Argentina	Gabriel Mercado	Mascherano



# SQL - DELETE

- Por fim, teremos o comando **DELETE**, que tem o papel de excluir um dado, por exemplo:

```
DELETE FROM Campeonato  
WHERE selecao = 'Portugal';
```

# SQL - DELETE

Nesse caso, excluímos Portugal da nossa tabela CAMPEONATO:



**Quadro 5.** Tabela CAMPEONATO com a instrução DELETE aplicada no campo \

Seleção	Atacante	Meio-campo	Lateral	Zagueiro
Brasil	Neymar	Kaká	Roberto Carlos	Lúcio
Argentina	Messi	Di Maria	Gabriel Mercado	Mascherano
Espanha	Juan	Xavi	Iniesta	Lavi

# SQL - DELETE

Agora, se precisássemos excluir todos os dados da tabela CAMPEONATO usariamos:  
**DELETE FROM CAMPEONATO;**



Seleção	Atacante	Meio-campo	Lateral	Zagueiro

# Operadores Relacionais

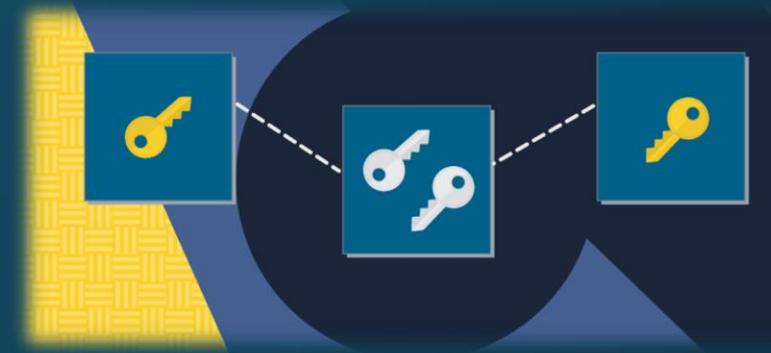


Os operadores relacionais são utilizados para realizar comparações entre valores e estrutura do Sistema de Gerenciamento de Banco de Dados. Veja um exemplo no Quadro 3.

**Quadro 3.** Operadores relacionais

Operador	Ação
=	Igual
<>	Diferente
>	Maior
>=	Maior ou igual
<	Menor
<=	Menor ou igual
BETWEEN	Buscar em um intervalo
NOT BETWEEN	Buscar o que não está em um intervalo
LIKE	Buscar com caractere curinga '%'
NOT LIKE	Buscar o que não está no caractere curinga '%'
IN	Buscar em uma lista
NOT IN	Buscar o que não está em uma lista

# CARDINALIDADE



- Refere-se ao número de itens que se relacionam nas Entidades;
- A cardinalidade demonstra, respectivamente, os números mínimos e máximos de instâncias de cada Entidade associadas no relacionamento.

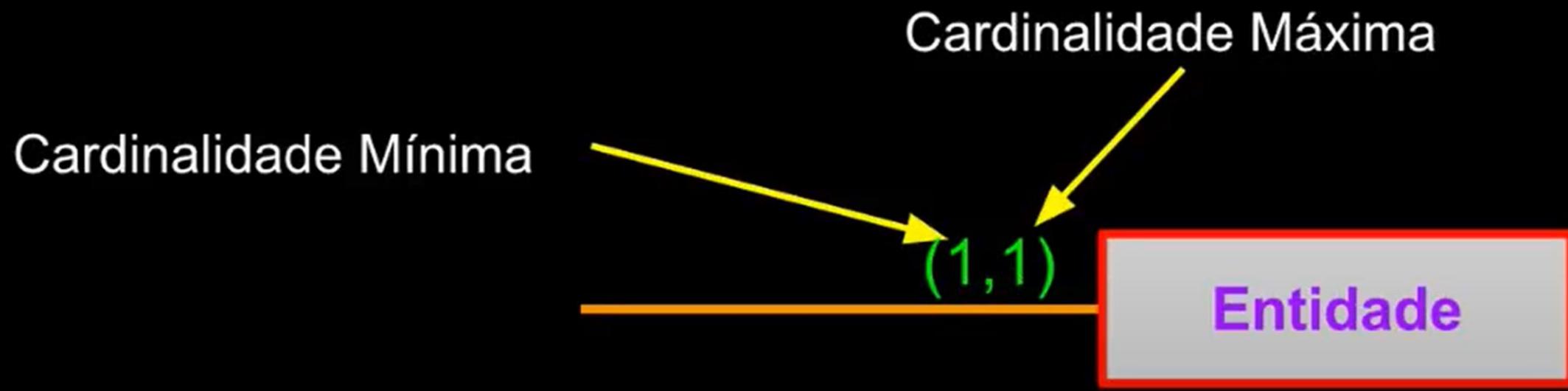
# Cardinalidades

Cardinalidade Máxima: Trata-se do número máximo de instâncias de entidade que podem participar em um relacionamento. Pode ser 1 ou N (muitos).

Cardinalidade Mínima: Número mínimo de instâncias de entidade que devem obrigatoriamente participar em um relacionamento; zero é participação opcional e um é obrigatoria.

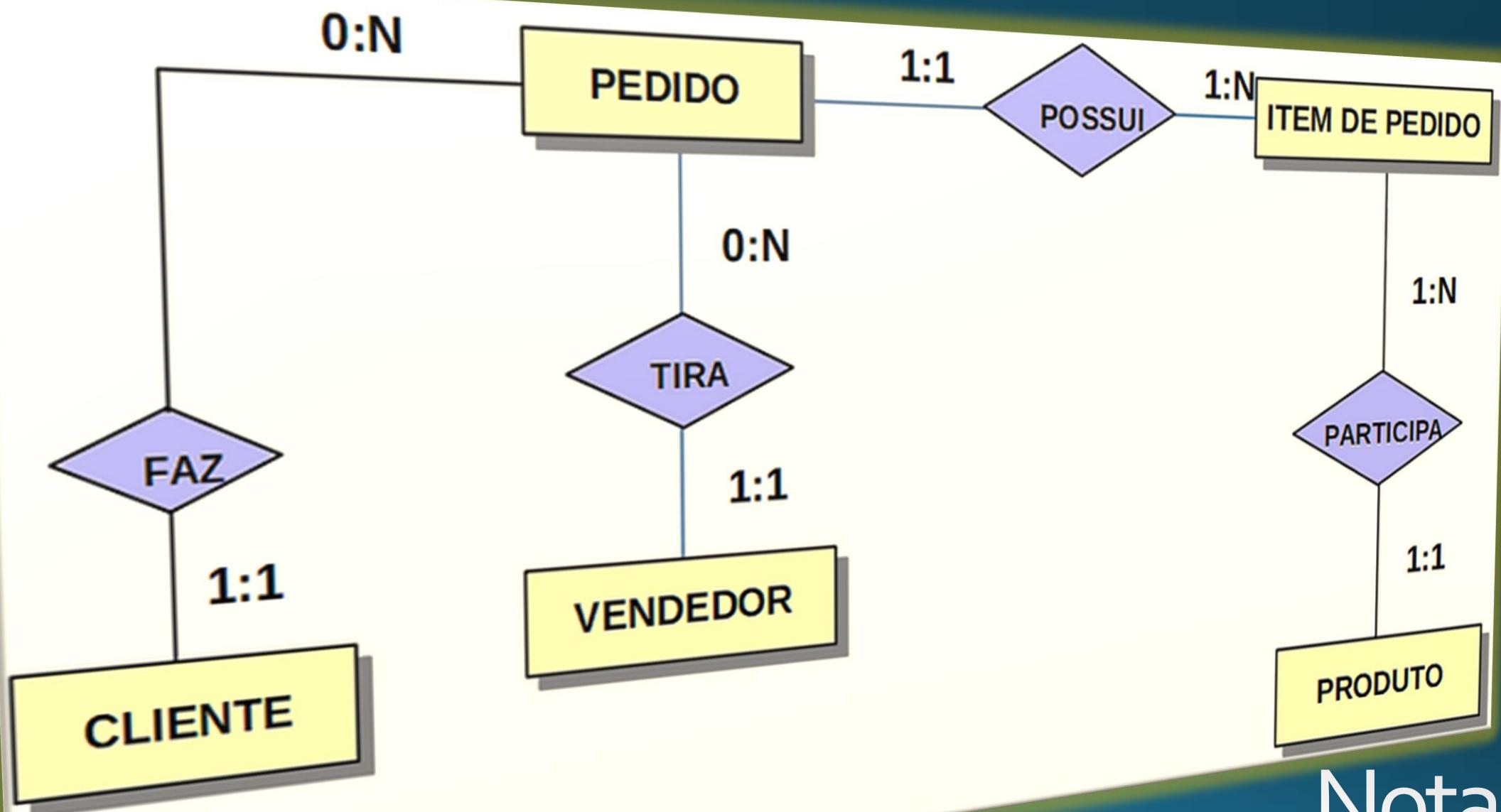
# Simbologia para Cardinalidades

Usando a notação de Peter Chen:



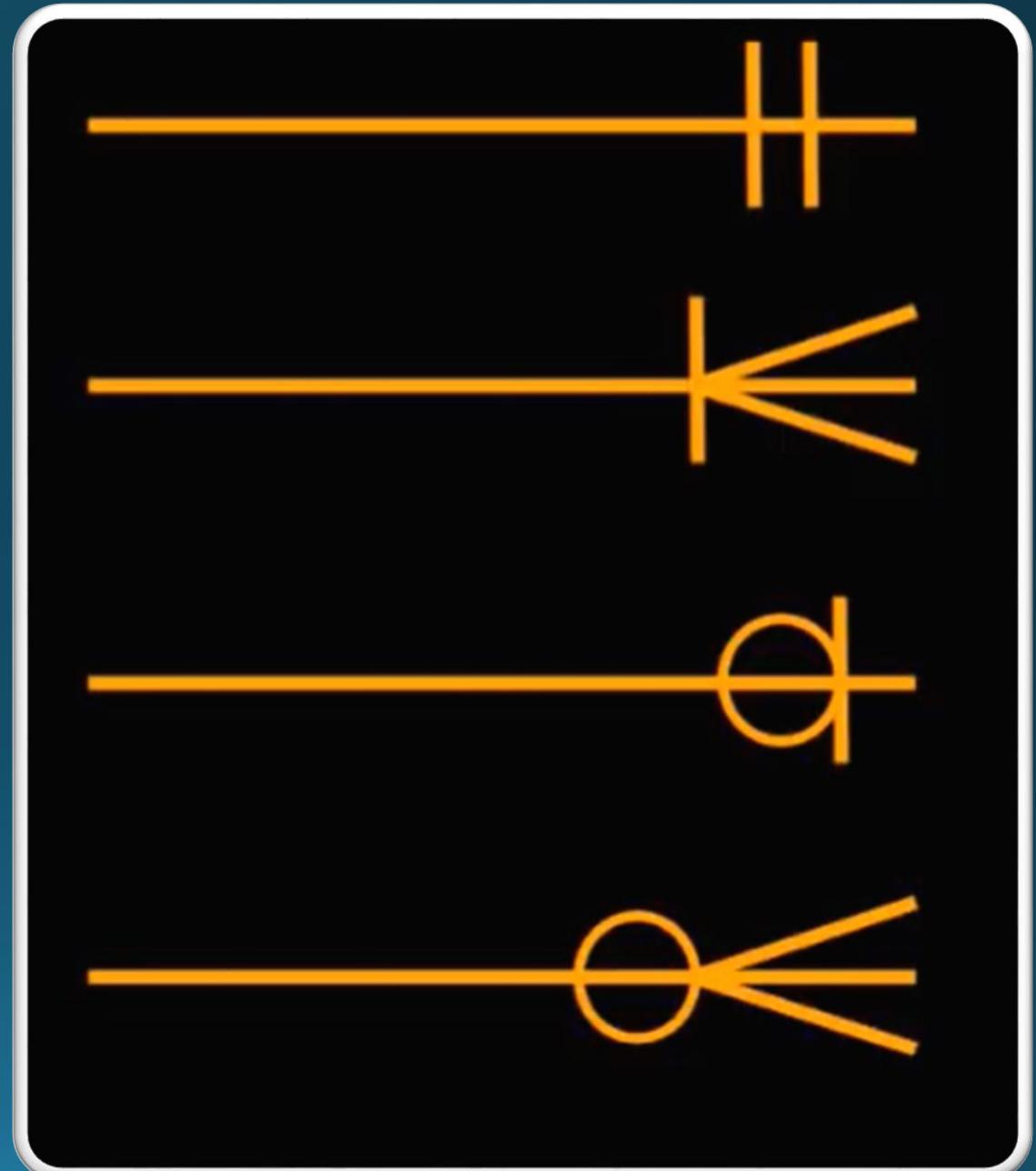


Notação  
Peter Chen



Notação  
Peter Chen

# Notação Pé de Galinha



# Simbologia para Cardinalidades



Um e apenas um (1)



De um a muitos (1 .. \*)

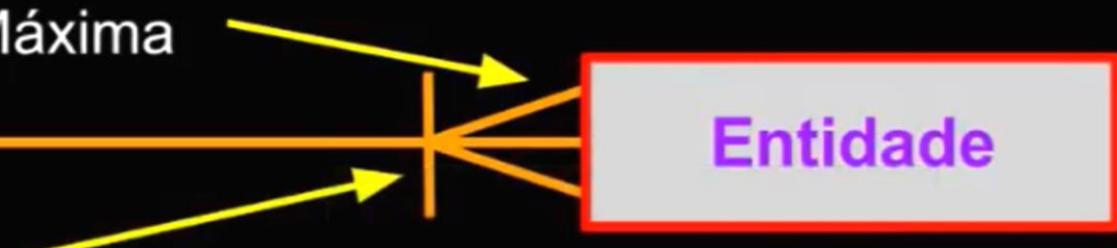


Zero ou um (0 .. 1)



De zero a muitos (0 .. \*)

Cardinalidade Máxima



Cardinalidade Mínima

# Exemplo de Cardinalidade



Cardinalidade Mínima: Um cliente para uma encomenda

Cardinalidade Máxima: Um cliente para muitas encomendas

# Cardinalidade: Notação Peter Chen

Pé-de-galinha



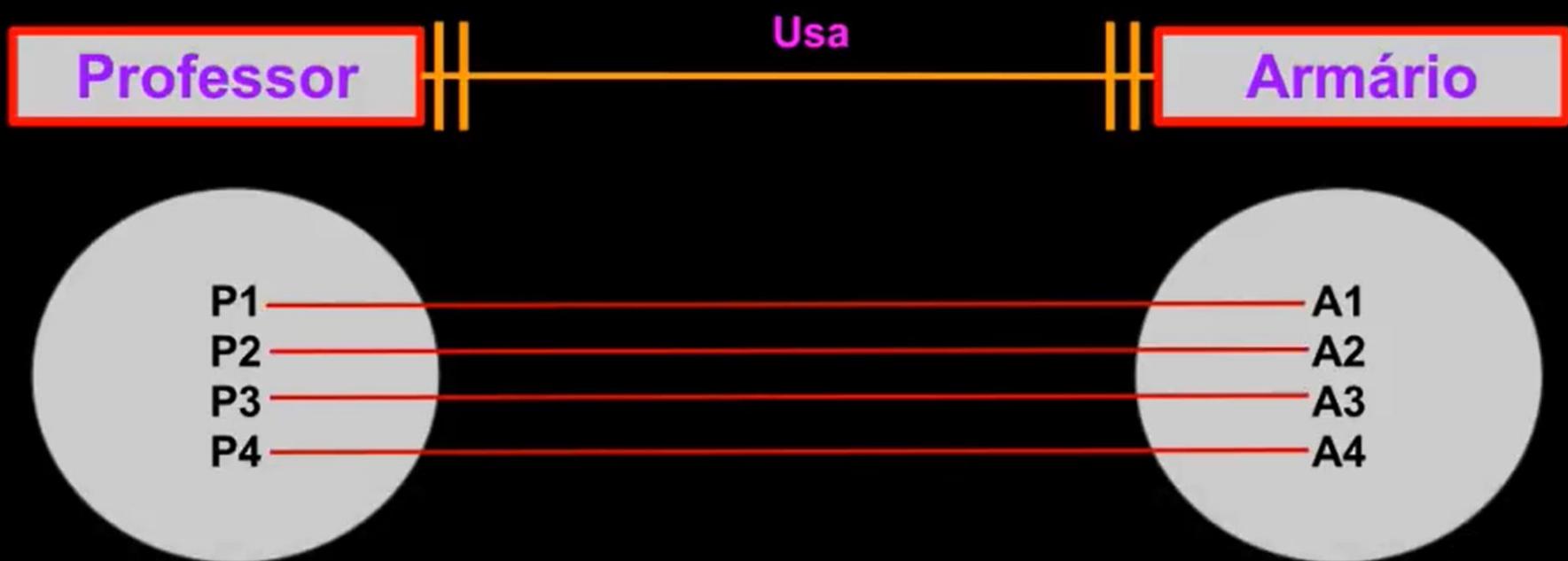
Peter Chen



# Relacionamento Binário um-para-um

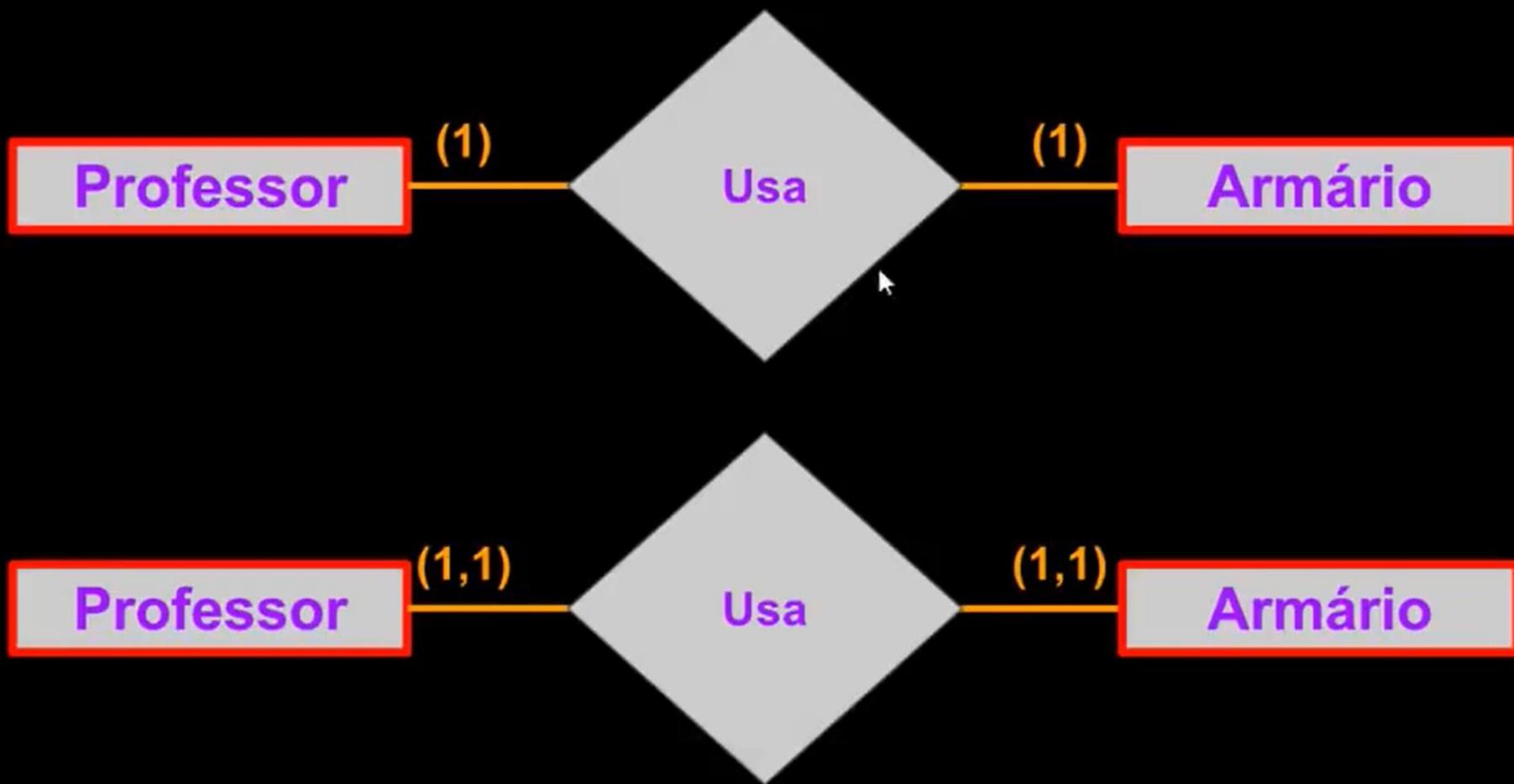
1:1

Uma instância de entidade única em uma entidade está relacionada com uma instância de entidade única em outra entidade.

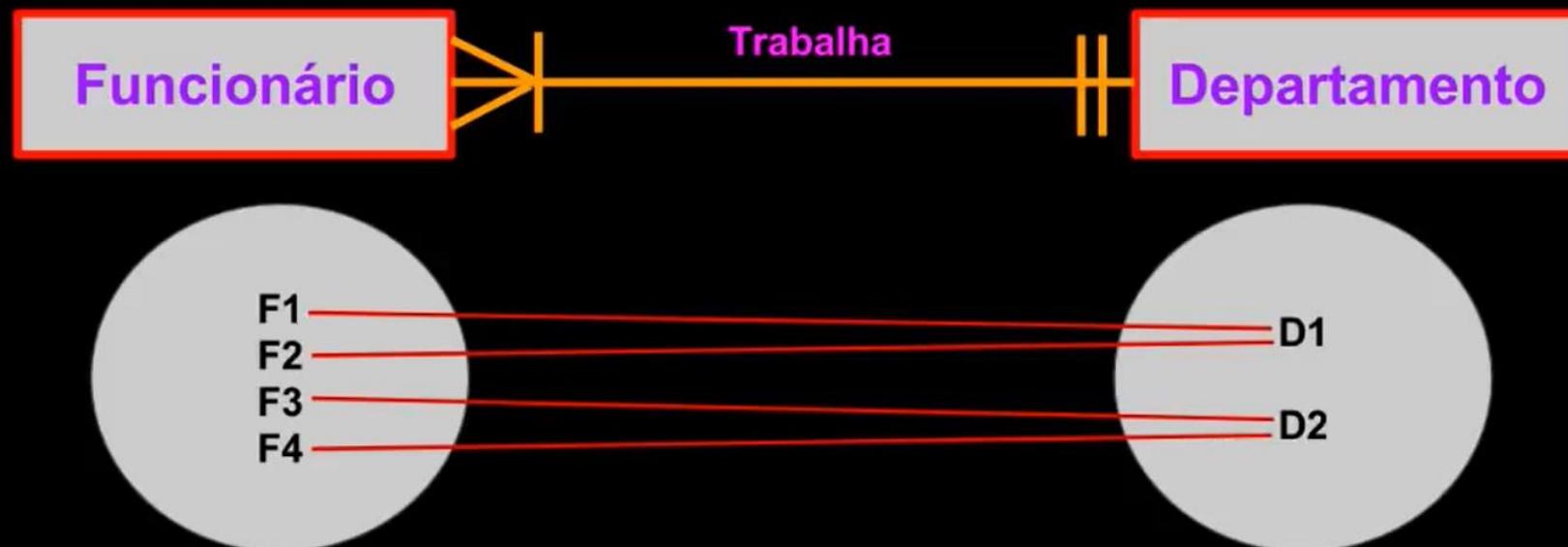


# Relacionamento Binário um-para-um

Usando a notação de Peter Chen:

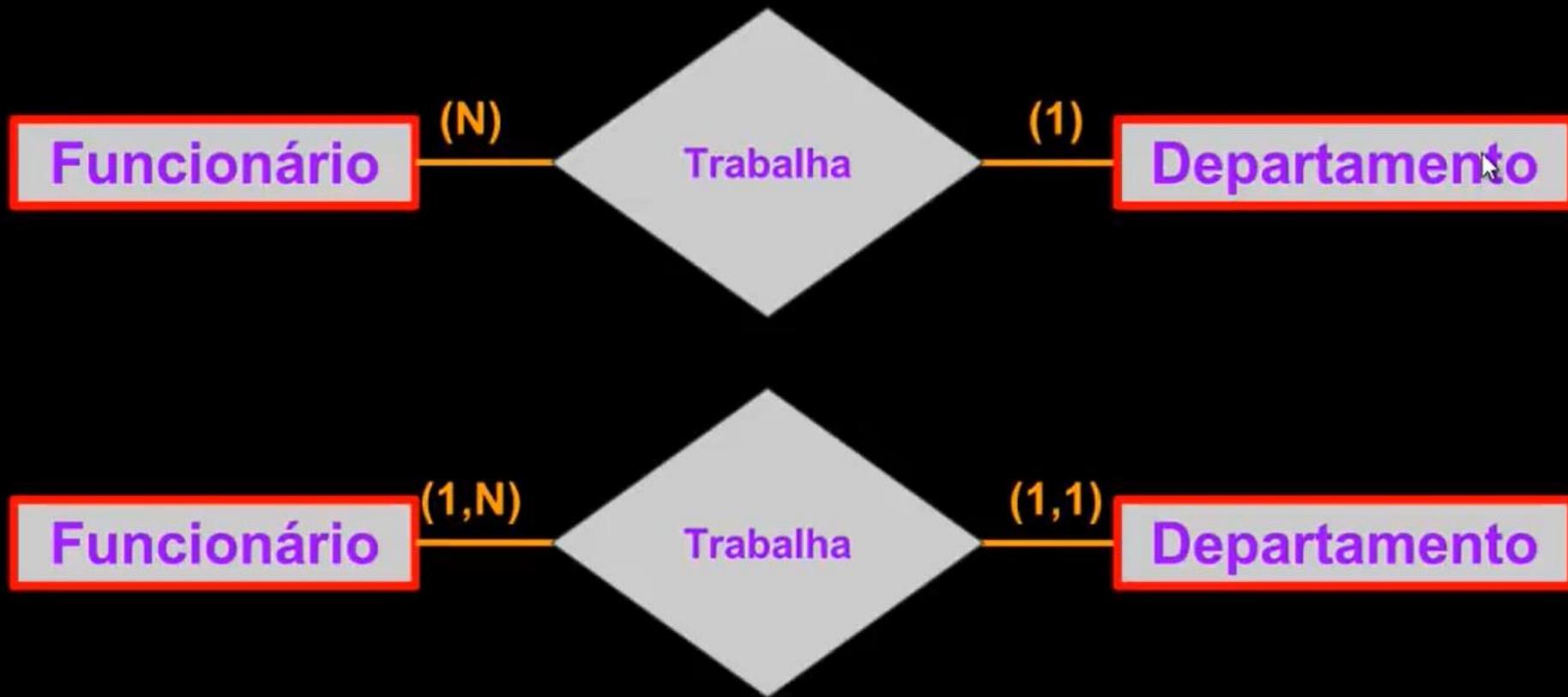


Uma instância de entidade única em uma classe de entidade está relacionada a muitas instâncias de entidade em outra classe de entidade.



# Relacionamento Binário um-para-muitos

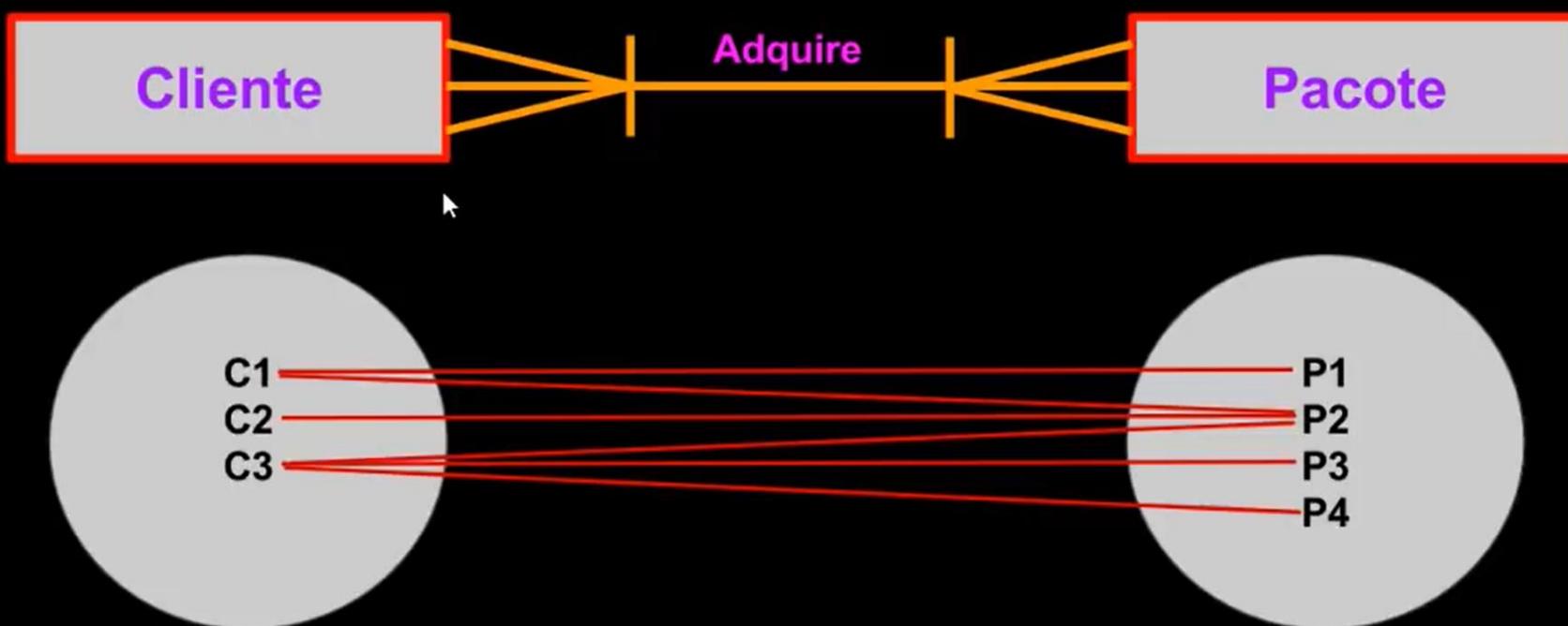
Usando a notação de Peter Chen:



# Relacionamento Binário muitos-para-muitos

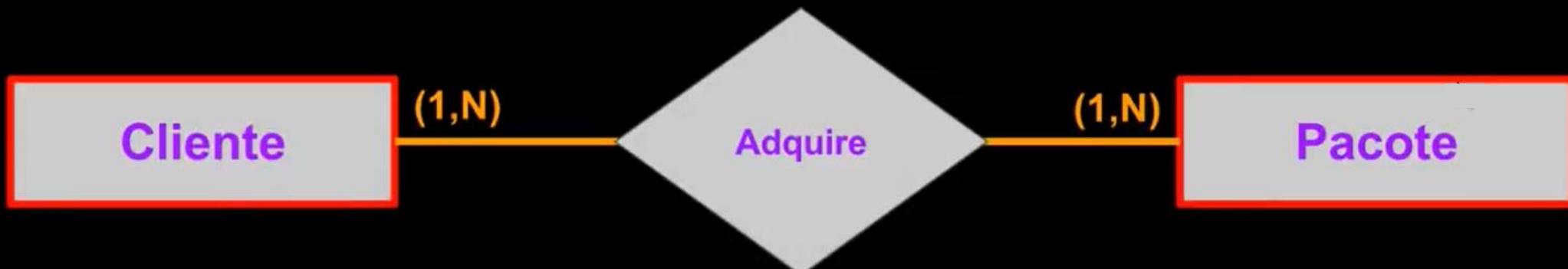
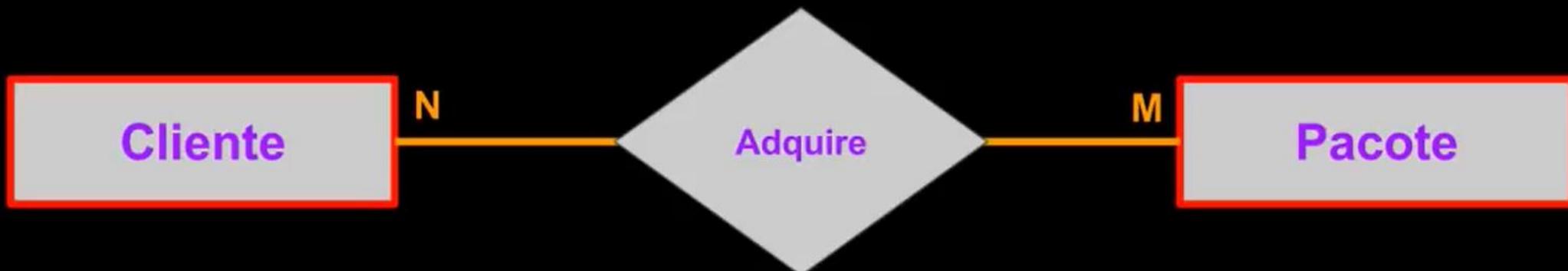
N:M

Muitas instâncias em uma entidade estão relacionadas a muitas instâncias de entidade em outra entidade.

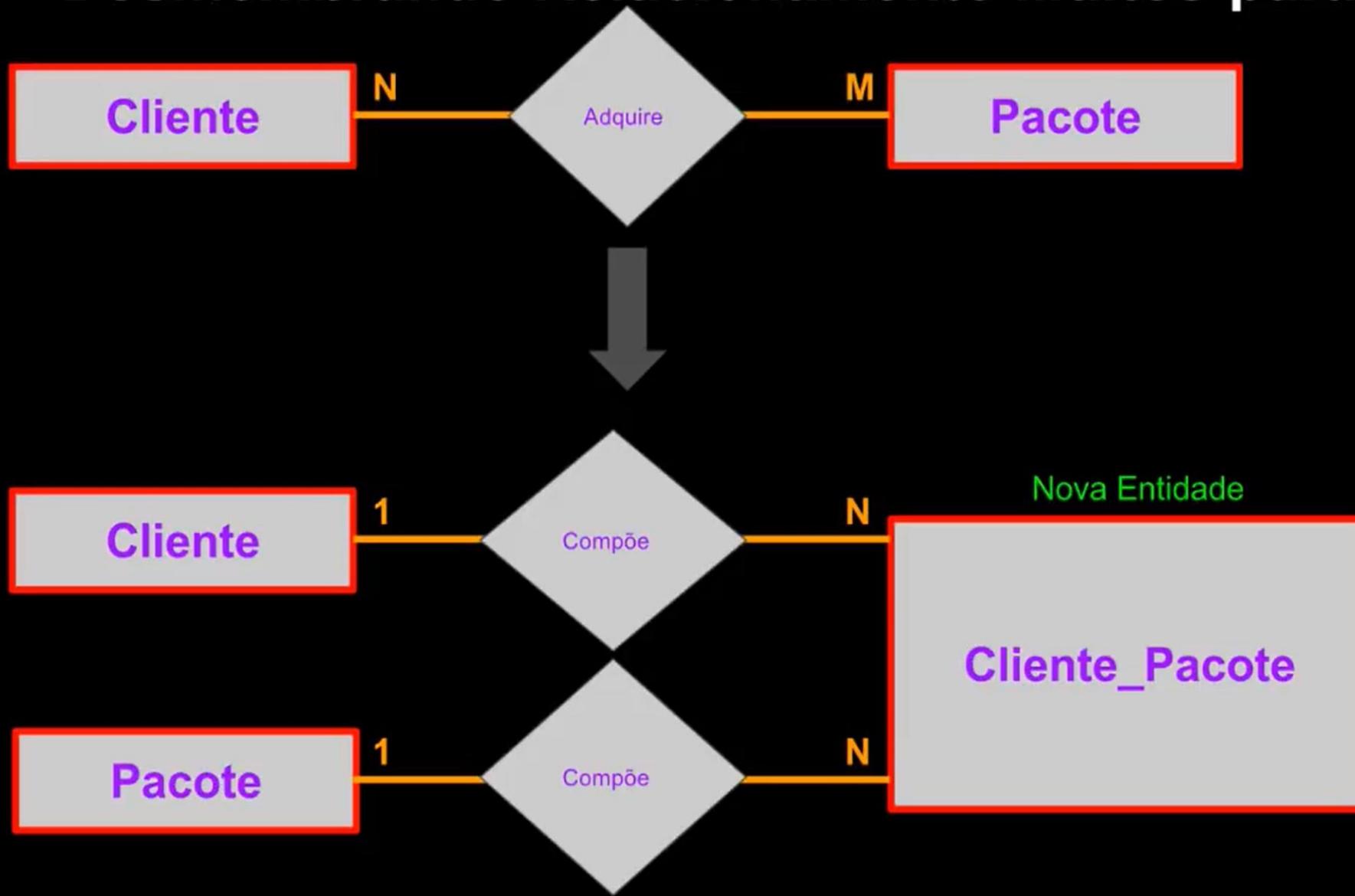


# Relacionamento Binário muitos-para-muitos

Notação Peter Chen



# Desmembrando Relacionamento Muitos-para-muitos



# Exercício 1:

**Desenvolva o Diagrama Entidade-Relacionamento para as seguintes situações (utilize, para cada Item abaixo, as notações de Peter Chen e Pé de Galinha):**

1. Um aluno realiza vários trabalhos. Um trabalho é realizado por um ou mais alunos.
2. Um diretor dirige no máximo um departamento. Um departamento tem no máximo um diretor.
3. Um autor escreve vários livros. Um livro pode ser escrito por vários autores.
4. Uma equipe é composta por vários jogadores. Um jogador joga apenas em uma equipe.
5. Um cliente realiza várias encomendas. Uma encomenda diz respeito apenas a um cliente.



# Exercício 2:

**D**esenvolva o Diagrama Entidade-Relacionamento para o armazenamento de dados e emissão de uma nota fiscal, conforme figura ao lado. Utilize, para cada relacionamento, as notações de Peter Chen e Pé de Galinha:

<h1><b>NOME FANTASIA</b></h1> <p><b>RAZÃO SOCIAL</b></p> <p><b>fone: (61) 3333-3333</b></p> <p>XXXXXXXXXXXX XXXXXXXXXX RUA 4B CHACARA 117 LOTE 0A LOJA 00 TAGUATINGA - DF CEP: 70.000-000</p> <p>CF/DF: 00.000.900/000-00 - CNPJ: 00.000.000/0000-00</p>		<p><b>NOTA FISCAL DE VENDA</b> <b>AO CONSUMIDOR</b> Modelo 002 Série D</p> <p>1ª Via - Destinatário 2ª Via - Contabilidade 3ª Via - Fixa no Talão</p>		
AIDF AUTORIZADA PARA ME OU EPP OPTANTE PELO SIMPLES NACIONAL      PROCON:151				
Data da Emissão _____	Valor _____			
Nome: _____ Endereço: _____ CPF: _____ Fone: _____ O ICMS já está incluído no preço das mercadorias. As informações abaixo deverão se preenchidas somente a pedido do consumidor (Dados relativos ao consumidor ou usuário final)				
Código	Quant.	Discriminação	P. Unitário	Preço Total
<b>TOTAL R\$</b>				
XX DA XXXX. CNPJ:00.000.000/0000-00 - CF/DF: 00.000.000/000-00 10 BL 50x3 - 001 a 500 - AIDF 0-000-00000/0000				



*Serviço Nacional de Aprendizagem Industrial*  
PELO FUTURO DO TRABALHO

# Dependências em Banco de Dados



# Dependência Funcional



Seja  $E$  uma entidade,  $X$  e  $Y$  dois atributos quaisquer de  $E$ . Dizemos que  $Y$  é funcionalmente dependente de  $X$ , se e somente se cada valor de  $X$  tiver associado a ele exatamente um valor de  $Y$ .

Simbolicamente:  $X \longrightarrow Y$

Lemos:  $X$  determina funcionalmente  $Y$

# Dependência Funcional



Ex: O prazo de entrega de um pedido depende do número do pedido:

Numero\_pedido → prazo\_entrega\_pedido

O atributo que determina o valor é chamado de Determinante.  
O outro atributo é chamado de Dependente.  
Uma chave primária em uma relação determina funcionalmente todos os outros atributos não-chave na linha.

# Dependência Funcional Total



Em uma relação com uma PK composta, um atributo não-chave que dependa dessa PK como um todo, e não somente de parte dela, possui uma dependência funcional total.

# Dependência Funcional Total



Exemplo:

Item_pedido	
PK	num_pedido
PK	cod_produto
	qtd_produto

\*qtd\_produto depende tanto de num\_pedido, quanto de cod\_produto ao mesmo tempo.

# Dependência Funcional Parcial



Quando os atributos não-chave não dependem funcionalmente de toda a PK, quando esta for composta.

Ou seja, existe uma dependência funcional mas somente de uma parte da chave primária (PK).

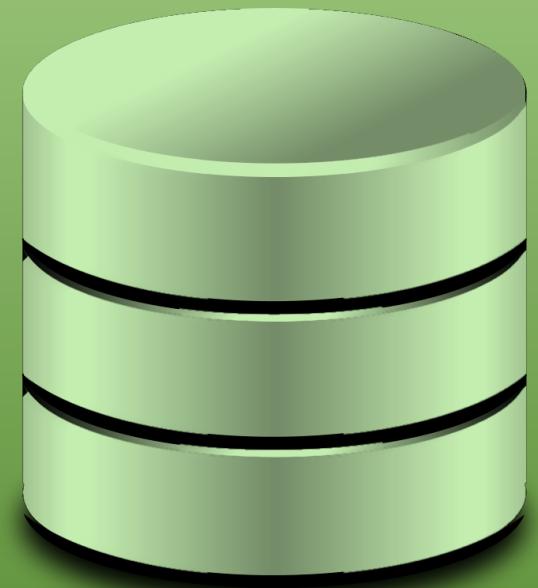
# Dependência Funcional Parcial

Exemplo:

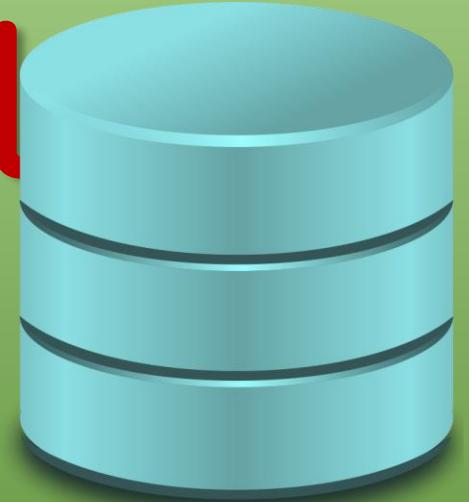
matriculas	
PK	<u>id_aluno</u>
PK	<u>cod_disciplina</u>
	nome_disciplina
	data_inicio



\*campo nome\_disciplina é dependente de cod\_disciplina,  
mas não do id\_aluno.



# Dependência Funcional Transitiva



Ocorre quando um campo não depende diretamente da chave primária da tabela, nem mesmo parcialmente, mas depende de um outro campo não-chave

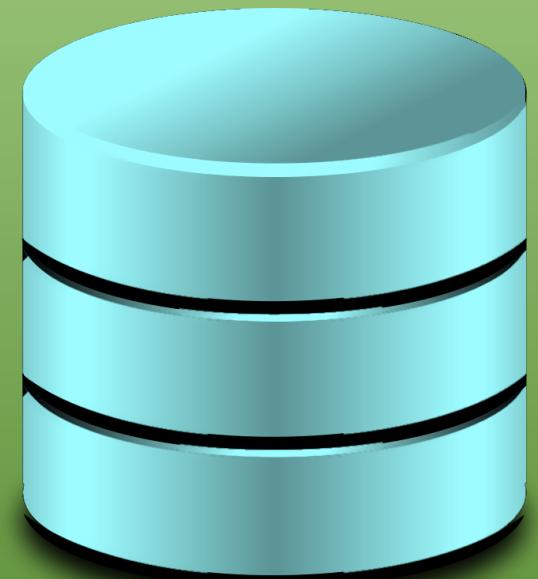
# Dependência Funcional Transitiva

Exemplo:

pedido	
PK	num_pedido
	prazo_entrega
FK	cod_vendedor
	nome_vendedor



\*campo nome\_vendedor depende funcionalmente de cod\_vendedor, que não é PK na Tabela. Já o campo prazo\_entrega depende da PK.



# Dependência Multivalorada



Ocorre quando um campo para cada valor de um atributo A, existe um conjunto de valores para outro atributo B e C que estão associados a ele, mas são independentes entre si.  
Representamos a dependência multivalorada assim:

**A >> B**

Onde B é a coluna que depende de A.

# Dependência Multivalorada

Exemplo:



Modelo	Ano	Cor
Gol	2016	Prata
Uno	2016	Preto
Uno	2015	Prata
Fox	2016	Vermelho
Fox	2014	Branco

\*Ano e Cor são independentes entre si e dependem do modelo do carro.  
Essas duas colunas são dependentes multivaloradas do Modelo.



*Serviço Nacional de Aprendizagem Industrial*  
PELO FUTURO DO TRABALHO

# Normalização em Banco de Dados



# Anomalias de Atualização

São problemas que ocorrem em bancos de dados mal planejados e não-normalizados, normalmente ocorrendo por excesso de dados armazenados em uma mesma tabela.



# Anomalia de Inclusão

Quando não for possível adicionar um dado a não ser que outro dado esteja disponível.

**Exemplo:** Não deve ser permitido cadastrar um livro, sem que um autor esteja cadastrado.



# Anomalia de Exclusão

Ao excluirmos um registro, dados referentes em outra tabela, devem ser também excluídos.

**Exemplo:** Se excluirmos um autor, os livros desse autor devem ser excluídos também.



# Anomalia de Modificação



Ao alterar os dados em uma tabela, dados em outras tabelas precisam ser alterados.

**Exemplo:** Se o código de um autor for modificado na tabela autores, o código do autor na tabela livros também deve ser modificado para manter o correto relacionamento entre ambas tabelas.

# Como eliminar tais anomalias?

Evitar que, no processo da modelagem do Banco de Dados, nenhuma anomalia de inclusão, exclusão ou modificação esteja presente nas relações.

E isso pode ser evitado utilizando-se a **NORMALIZAÇÃO**.



# Normalização

Processo de análise  
de uma relação para  
assegurar que seja  
bem realizada.



# Normalização

Decompor relações com anomalias para produzir relações menores e bem estruturadas.



# Objetivos:



Analizar esquemas de relação entre tabelas com base em suas dependências funcionais e chaves primárias para:

- 1 – Minimizar redundâncias;
- 2 – Minimizar anomalias de inserção, exclusão e modificação

As relações são decompostas em esquemas de relação menores, que atendem aos testes de Forma Normal.

# Primeira Forma Normal

Busca reprovar atributos multivvalorados, compostos e suas combinações.



# Primeira Forma Normal

O domínio de um atributo deve incluir apenas valores atômicos (indivisíveis) e o valor de qualquer atributo em uma tupla (registro) deve ser único valor do domínio desse atributo.



**Uma tabela está na 1<sup>a</sup> Forma Normal, quando:**

- \* Possui somente valores atômicos;
- \* Não há grupos de atributos repetidos;
- \* Existe uma chave Primária;
- \* Relação não possui atributos multivalorados ou relações aninhadas.



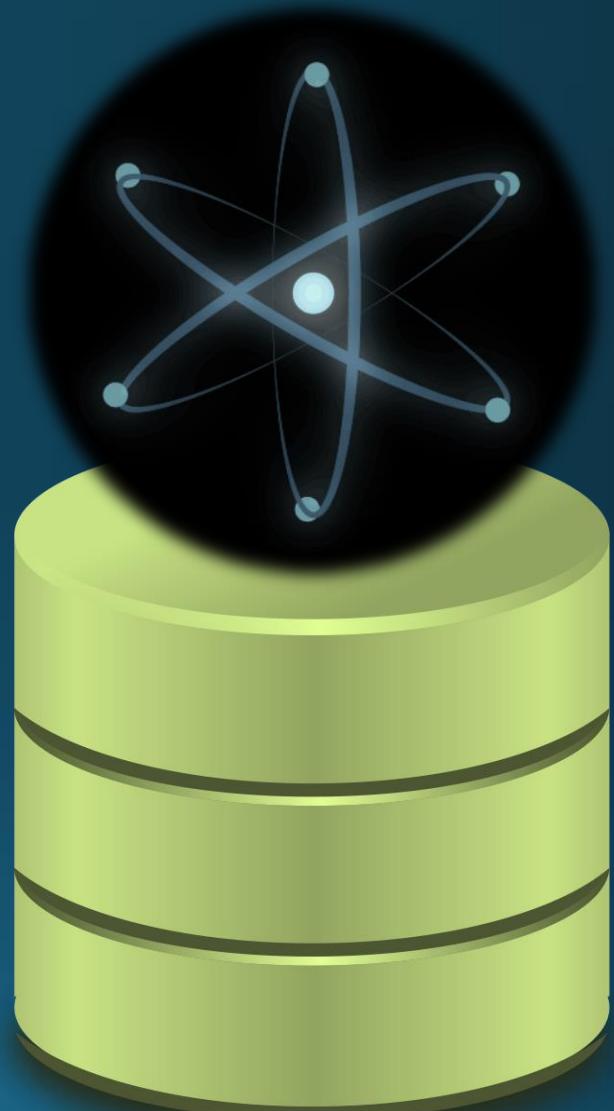
## Uma tabela está na **1<sup>a</sup> Forma Normal**, quando:

- \* Se somente houverem valores atômicos no domínio de seus atributos;
- \* Um valor atômico é indivisível;
- \* Como exemplo, um campo de Endereço possui subdomínios: rua, número e CEP.  
Esses itens devem ser separados no processo de normalização.
- \* Cada informação deve ser colocada em um campo diferente.



# Dados Atômicos:

- \* Elementos de dados que representam o nível mais baixo de detalhamento;
- \* Campos não-atômicos são aqueles que podem ser subdivididos em mais de um campo, pois eles escondem detalhes, como por exemplo o nome de uma pessoa, que contém o nome e o sobrenome.



# Normalizando até a 1FN:



Cliente
cod
nome
telefone
Endereco

Cliente				
cod	nome	telefone	Endereco	
2532	MARIA	99653-2145 2865-3212	Rua das Flores, 540 – Vila Rica	
2536	JOSE	2643-5321	Av. Moisés Mafra, 610 – Bela Vista	
2453	ANA	4213-6532 97563-5632	Rua Ver. Jorge Martins, 395 - Coqueiros	
2519	MARCOS	3241-6517	Rua do Limoeiro, 37 - Barreiros	

# Tabela Normalizada – 1FN:



Cliente				telefone	
cod	nome	rua	bairro	cod_cli	tel_cliente
2532	MARIA	Rua das Flores, 540	Vila Rica	2532	99653-2145
2536	JOSE	Av. Moisés Mafra, 610	Bela Vista	2532	2865-3212
2453	ANA	Rua Ver. Jorge Martins, 395	Coqueiros	2536	2643-5321
2519	MARCOS	Rua do Limoeiro, 37	Barreiros	2453	4213-6532
				2453	97563-5632
				2519	3241-6517

# Segunda Forma Normal



- Baseada no conceito da dependência Funcional total.
- Um esquema de relação R está na 2NF se cada atributo não-chave de R for total e funcionalmente dependente da PK de R.

# Segunda Forma Normal



- Para testar a 2FN, testamos as dependências funcionais cujos atributos fazem parte de PK.
- Caso a PK tenha um único atributo, esse teste não precisa ser aplicado.

# Uma tabela está na **2<sup>a</sup> Forma Normal**, quando:

- \* Está na Primeira Forma Normal;
- \* Todos os atributos não-chave são funcionalmente dependentes de todas as partes da chave primária;
- \* Não existem dependências parciais.
- Caso contrário deve-se gerar uma nova tabela com os dados.



*Um atributo chave é um que uma PK ou parte de uma PK composta.*

# Segunda Forma Normal



- Deve-se criar uma nova relação para cada chave PK ou combinação de atributos que forem determinantes em uma dependência funcional.
- Esse atributo será a PK na nova tabela.
- Mova os atributos não-chave dependentes desta PK para a nova tabela.



# Normalizando até a 2FN:

Peças	
cod	
cod_fornec	
local_fornec	
qtd_estoque	
tel_fornec	
qtd_caixas	

►

Peças						
cod	cod_fornec	local_fornec	qtd_estoque	tel_fornec	qtd_caixa	
0009	121	São Paulo	512	2365-6532	52	
0023	122	Manaus	263	4465-8632	27	
0065	121	São Paulo	196	2365-6532	20	
0071	123	Porto Alegre	89	2956-8653	9	
0073	122	Manaus	296	4465-8632	30	

PK



# Tabela Normalizada até a 2FN:

Peças			
cod	cod_fornec	qtd_estoque	qtd_caixa
0009	121	512	52
0023	122	263	27
0065	121	196	20
0071	123	89	9
0073	122	296	30

} PK

FK

FORNECEDOR		
cod_fornec	local_fornec	tel_fornec
121	São Paulo	2365-6532
122	Manaus	4465-8632
123	Porto Alegre	2956-8653

PK

# Terceira Forma Normal



- Baseada no conceito da dependência Transitiva.
- A relação não deve ter um atributo não-chave determinado funcionalmente por outro atributo não-chave (ou conjunto).

# Terceira Forma Normal



- Não deve haver dependência transitiva de um atributo não-chave sobre a PK.
- Deve-se decompor e montar uma nova relação que inclua os atributos não-chave que determinam funcionalmente outros atributos não-chave.

# Uma tabela está na 3<sup>a</sup> Forma Normal, quando:

- \* Estiver na Segunda Forma Normal;
- \* Não existirem **Dependências Transitivas**;
- \* Se ela estiver na Segunda Forma Normal e se nenhuma coluna-chave depender de outra coluna não-chave.

*Uma dependência transitiva em uma tabela é uma dependência funcional entre dois ou mais atributos não-chave.*



# Terceira Forma Normal



- Para cada atributo (ou grupo) não-chave que for um determinante na relação, crie uma nova tabela.
- Esse atributo será a PK na nova relação.
- Mova, então, todos os atributos que são dependentes funcionalmente do atributo chave para a nova tabela.
- O atributo (PK na nova relação) fica também na tabela original, e servirá como uma chave estrangeira para associar as duas relações.

# Normalizando até a 3FN:



venda			
NF	cod_vend	Cod_prod	qtd_vend
15326	002	132	10
15327	006	153	12
15328	002	143	14
15329	009	132	9
15330	007	153	22

PK

FK

vendedor	
cod_vend	Nome_vend
002	Maria
006	Mateus
007	Luiz
009	Paulo

PK

# Evolução da Normalização:

Tabela  
Não-Normalizada

Remover atributos  
multivalorados e compostos

1<sup>o</sup> FN

Remover Dependências  
Parciais

2<sup>o</sup> FN

Remover Dependências  
Transitivas

3<sup>o</sup> FN

