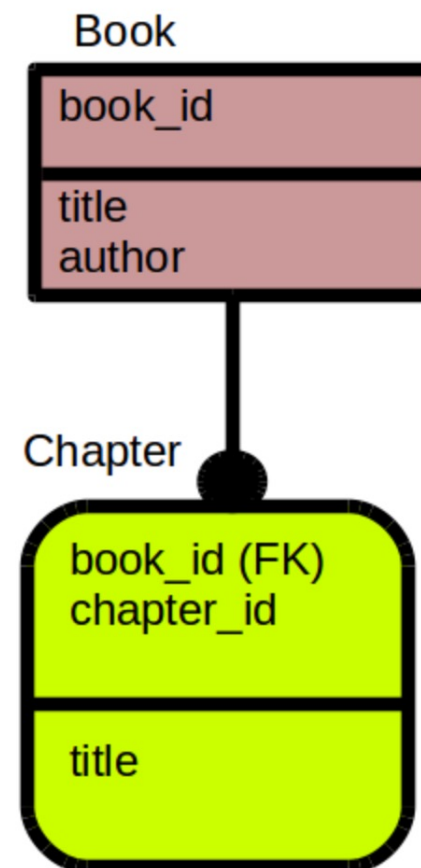
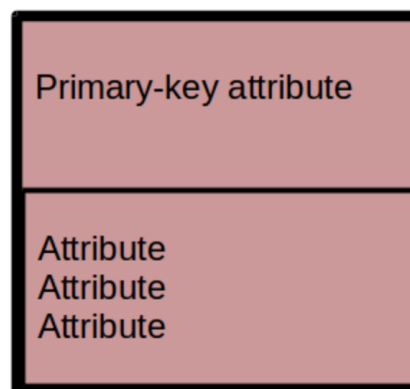
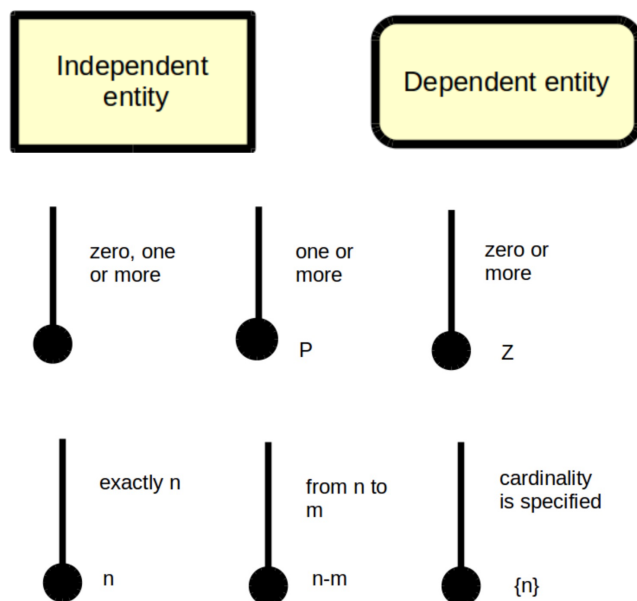
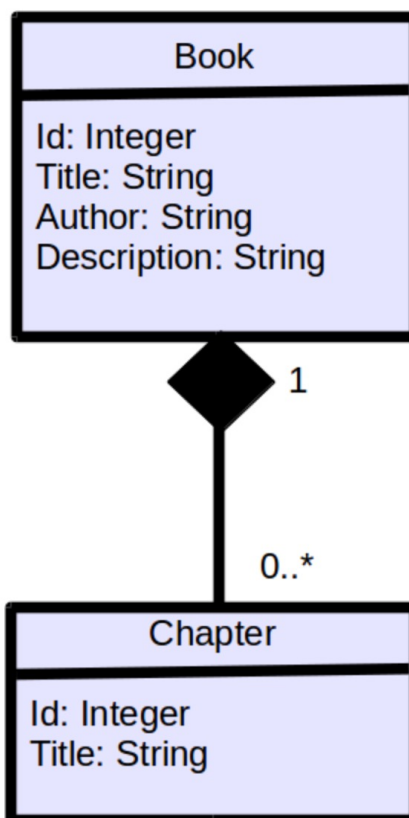
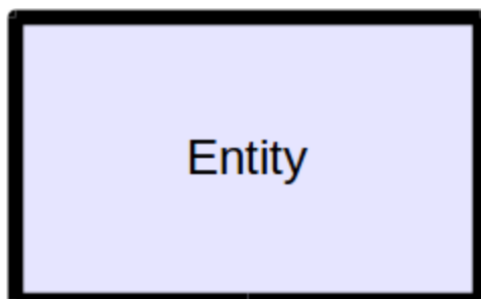


## IDEF1X Notation

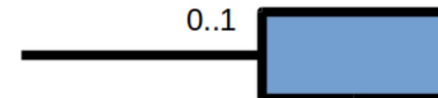


## UML Notation



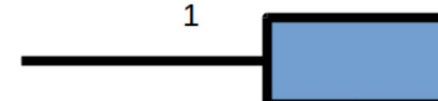
Zero or one

0..1



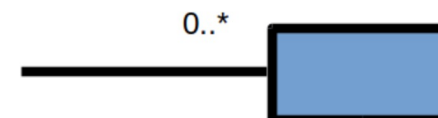
One and only one

1



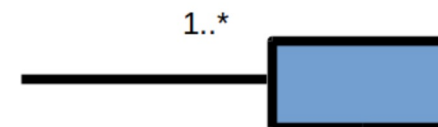
Zero or more

0..\*



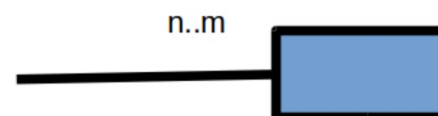
One or more

1..\*

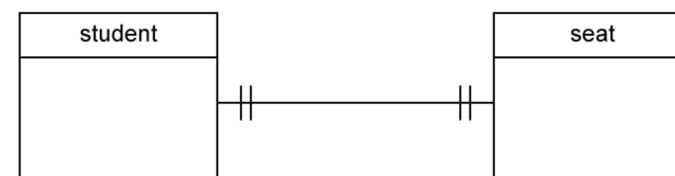
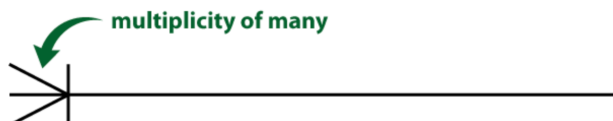
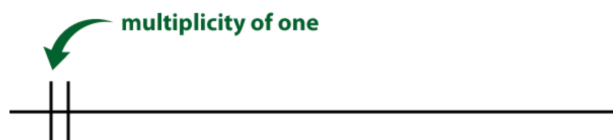
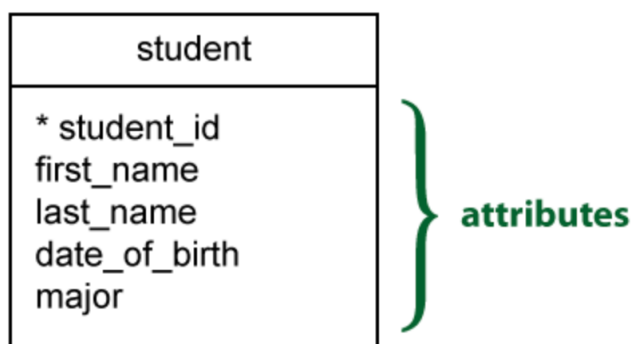


The range is specified

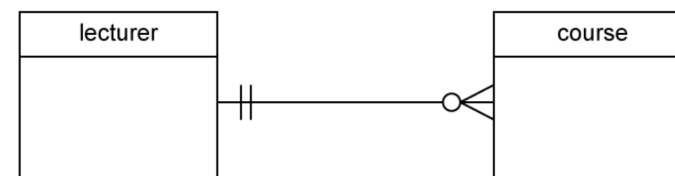
n..m



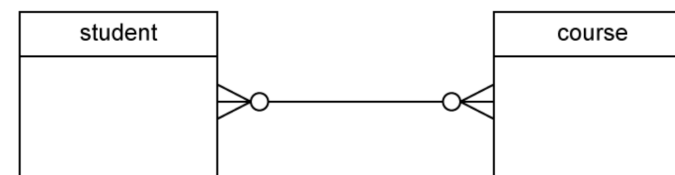
## Crow's foot Notation (IE - Information Engineering)



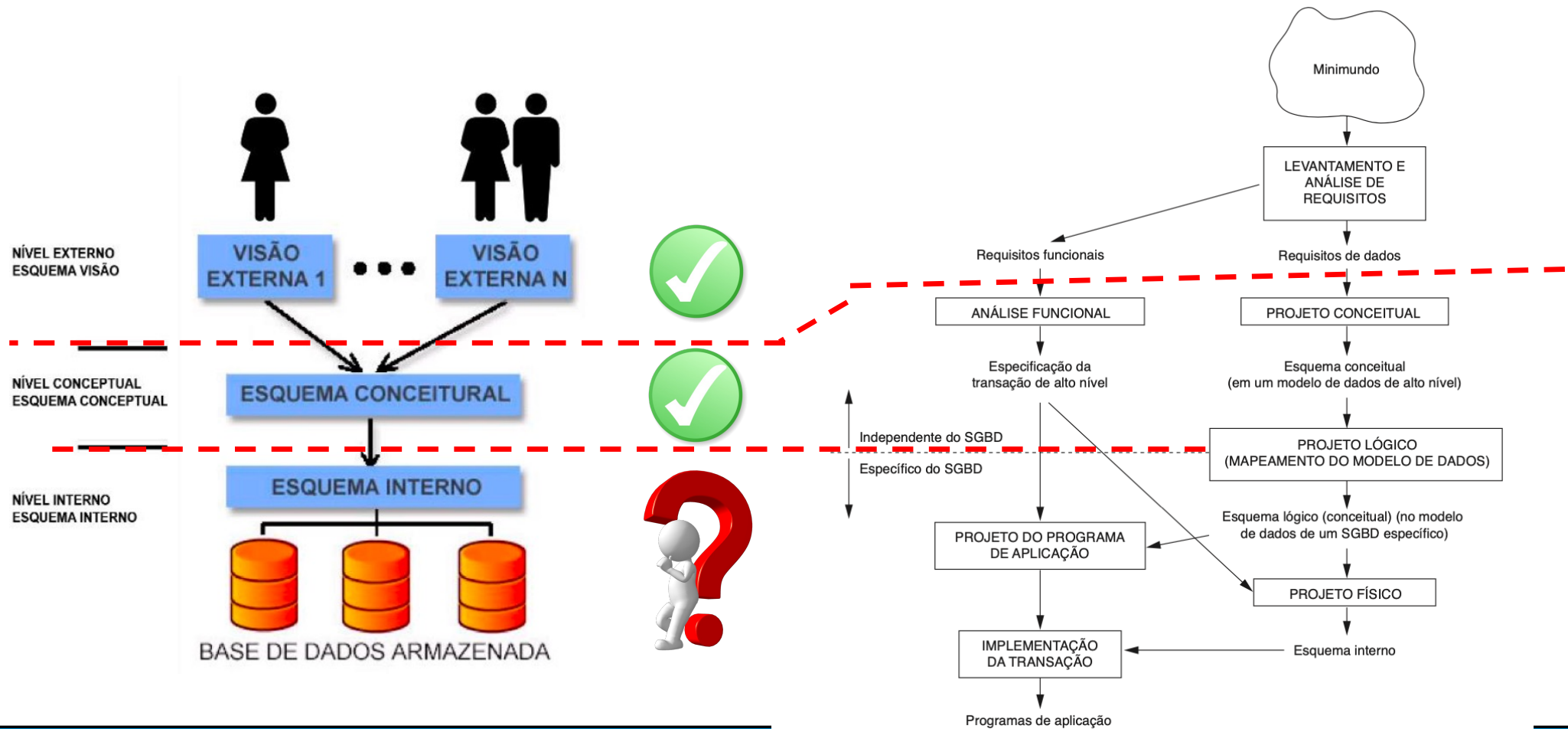
• One-to-many

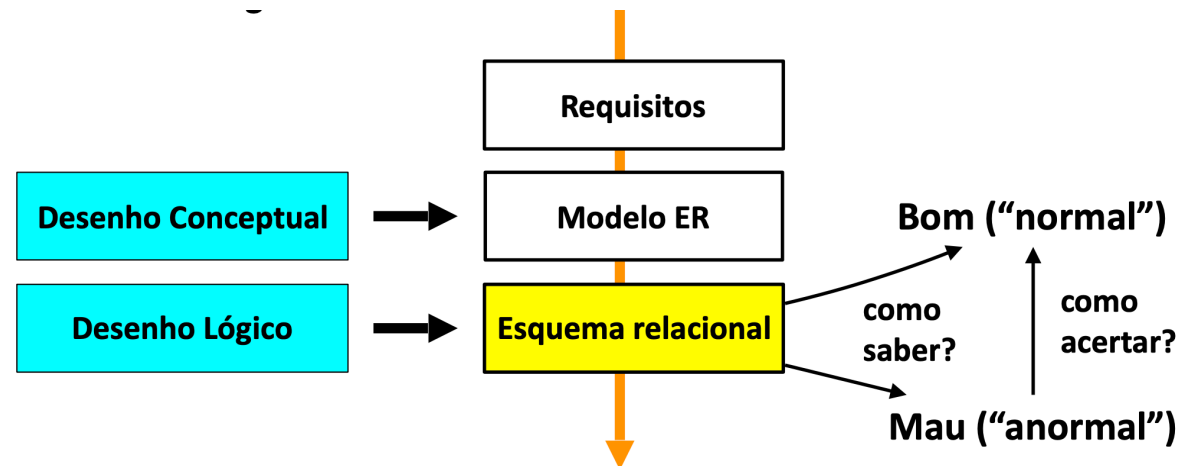


• Many-to-many



# Ciclo de vida de uma Base de Dados

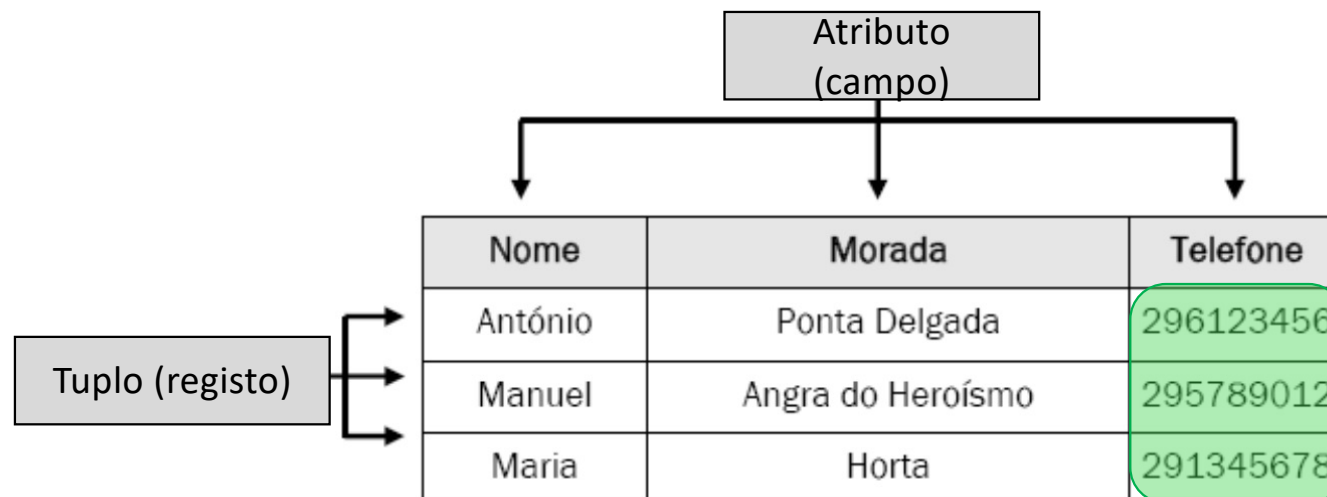




Qual é o critério para aferir se o **desenho de uma BD**, expresso por um esquema relacional, é “bom” ou “mau”? E se for “mau”, como resolver?

Um “mau” desenho tem **“anomalias”** (é **“anormal”**) que deveremos conseguir detetar de forma objetiva, e estas devem poder ser eliminadas por um processo de **normalização** do esquema.

# MR – Conceitos (Relembrar)



- Uma **Entidade** é representada por uma tabela, que no modelo Relacional se designa por **Relação**;
- Os Atributos (**campos**) são as colunas dessa **tabela**;
- Às linhas da tabela dá-se o nome de **registos ou tuplos** (designação formal num MR).
- Ao conjunto de valores permitidos para um atributo designamos de **Domínio** Todos os números de telefone

A nomenclatura dos atributos (colunas das tabelas) nem sempre são legíveis ou facilmente entendíveis. Como raramente é necessário mostrar uma tabela nua e crua, sem filtragem ou formatação a um utilizador, por vezes não existe a preocupação de usar nomes “apresentáveis” ou que sejam autoexplicativos a qualquer um que aceda. A escolha dos nomes dos campos de uma tabela, numa base de dados, obedece a outros critérios e regras.

## **Regras para Nomes dos Atributos (Campos):**

- Para siglas utilizar todas as letras em maiúsculo;
- Usar palavras no singular e sem acentuação;
- Usar nome que identifique e individualize os dados dentro da tabela; - Dar nomes distintos para dados distintos;
- Utilizar o nome sempre no singular;
- Acrescentar sempre comentários sobre a informação da coluna

**EXEMPLOS:** cd\_pessoa - código da pessoa; nm\_pessoa - nome da pessoa; dt\_nascimento - data de nascimento; vl\_salario - valor do salário

- **Chave candidata de uma relação**: atributo ou conjunto de atributos que permitem identificar de forma inequívoca qualquer tuplo dessa relação. O conjunto não pode ser reduzido sem perder essa qualidade.

De entre as possíveis chaves candidatas é escolhida uma que será declarada como chave Primária

- **A Chave Primária** terá que ser:
  - ✓ **Unívoca**: o atributo (ou atributos) da chave primária têm um valor único para qualquer tuplo da relação
  - ✓ **Não nula**: Não pode haver tuplos da relação que tenham o atributo (ou atributos) da chave primária nulos (sem qualquer valor)
  - ✓ **Não redundante**: Se algum dos atributos que a constituem for retirado os restantes deixam de identificar univocamente o tuplo.



- Quais as chaves candidatas?

Nome	B.I	N_contribuinte	N_eleitor	Freguesia	Concelho
Maria	1234567	123456722	2222	S. Pedro	Covilhã
Manuel	3377229	234156233	3333	Conceição	Covilhã
Paulo	2233337	233333567	3456	S. Maria	Covilhã
Paula	2876909	222333333	6782	S. Tiago	Covilhã

## EXEMPLO:

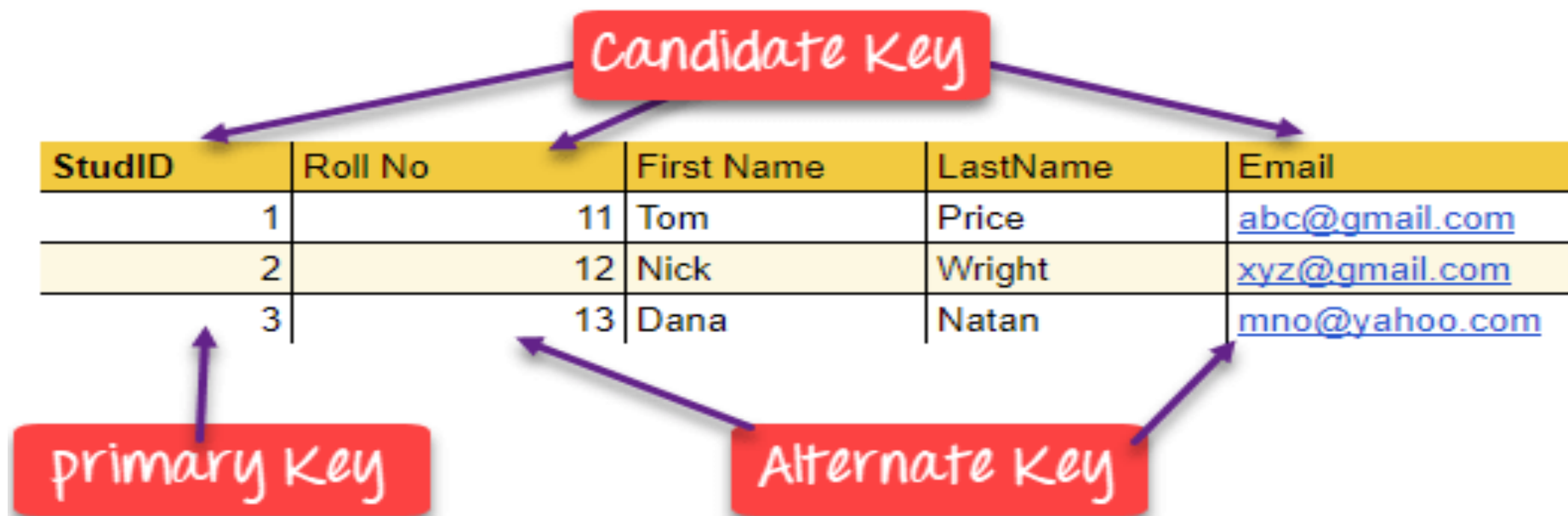
- {B.I.}
- {N\_Contribuinte}
- {N\_Eleitor, Freguesia, Concelho}

- ✓ **Unívoca:** o atributo (ou atributos) da chave primária têm um valor único para qualquer tuplo da relação
- ✓ **Não nula:** Não pode haver tuplos da relação que tenham o atributo (ou atributos) da chave primária nulos (sem qualquer valor)
- ✓ **Não redundante:** Se algum dos atributos que a constituem for retirado os restantes deixam de identificar univocamente o tuplo.

- **Chave Primária:** É a super-chave de uma relação, qualquer subconjunto de atributos que identifique univocamente qualquer tuplo da relação. No limite (não recomendável) o conjunto de todos os atributos da relação é uma super-chave.

Nome	B.I	N_contribuinte	N_eleitor	Freguesia	Concelho
Maria	1234567	123456722	2222	S. Pedro	Covilhã
Manuel	3377229	234156233	3333	Conceição	Covilhã
Paulo	2233337	233333567	3456	S. Maria	Covilhã
Paula	2876909	222333333	6782	S. Tiago	Covilhã

Melhor escolha será, no caso de Portugal, o **N\_contribuinte**, até porque é único para cada cidadão, todos possuem um. Já nem todos possuem BI (☺ exemplo antiquado)



**Chave secundária:** (SK – Secondary Key) é aquela chave candidata que não é primaria.

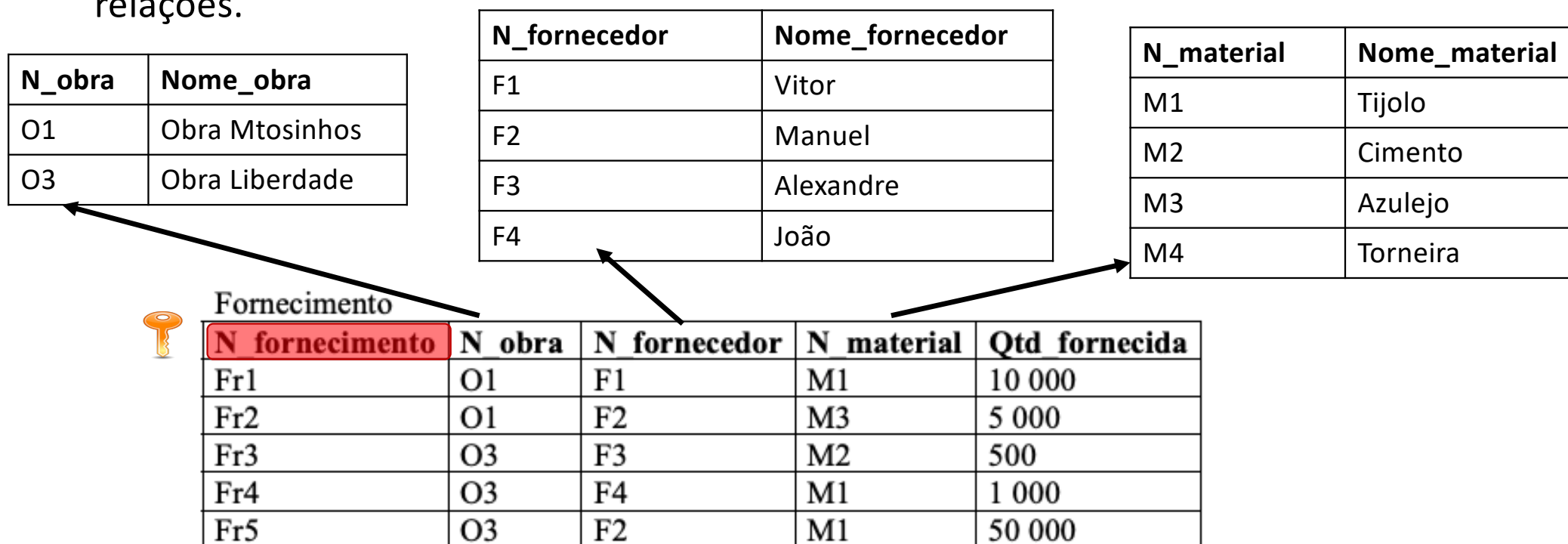
- **Chave Estrangeira:** Subconjunto de atributos que constituem a chave primária de uma outra relação permitindo estabelecer a associação entre tuplos de diferentes relações.

Fornecimento

N fornecimento	N obra	N fornecedor	N material	Qtd fornecida
Fr1	O1	F1	M1	10 000
Fr2	O1	F2	M3	5 000
Fr3	O3	F3	M2	500
Fr4	O3	F4	M1	1 000
Fr5	O3	F2	M1	50 000

- **N\_fornecimento** é chave primária da relação Fornecimento
- **N\_obra** é chave estrangeira da relação Fornecimento porque é chave primária na relação Obra
- **N\_fornecedor** é chave estrangeira da relação Fornecimento porque é chave primária na relação Fornecedor
- **N\_material** é chave estrangeira da relação Fornecimento porque é chave primária na relação Material

- **Chave Estrangeira:** Subconjunto de atributos que constituem a chave primária de uma outra relação permitindo estabelecer a associação entre tuplos de diferentes relações.



- **Uma tabela deve representar apenas um conceito do universo da BD**, em correspondência a uma entidade ou um relacionamento entre entidades, e manter a informação completa associada a este.
- **Informação de uma tabela não deve ser duplicada em outra**, com exceção da chave primaria através de referências por chaves externas (chave estrangeira).
- As tabelas devem relacionar-se, de igual modo, **apenas** via ligações **chave-estrangeira/chave primaria** e não por outros atributos.
- Devemos evitar atributos que sejam **NULL** em um grande número de casos.

## A violação destes princípios leva geralmente a:

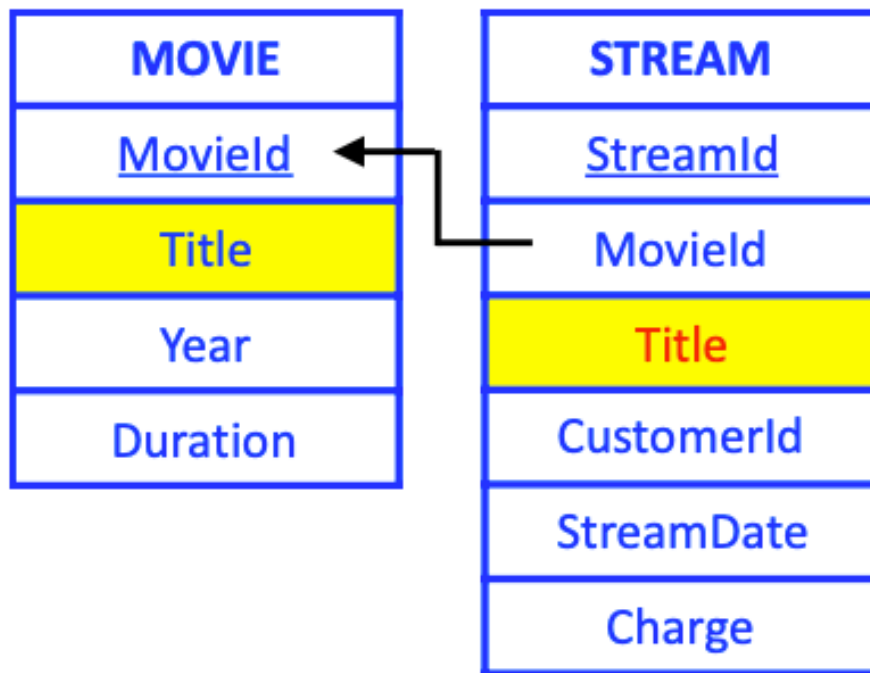
- a um **significado semântico pouco claro** do esquema da BD;
- **anomalias** em operações de dados e/ou **inconsistências** resultantes das mesmas
- **desperdício e redundância** no armazenamento;

MOVIE	STREAM
<u>Movied</u>	<u>StreamId</u>
Title	Title
Year	CustomerId
Duration	StreamDate
	Charge

### Mau desenho:

- Relação entre **STREAM** e **MOVIE** feita através de **TITLE** sem relação expressa no esquema.
- Embora **TITLE** seja um atributo chave em **MOVIE**, a chave primaria é **Movied**.
- **Possível ambiguidade semântica**: será que **STREAM** tem um titulo afinal e não se relaciona com **MOVIE** ?!

## Mais exemplos:



### Mau desenho ????

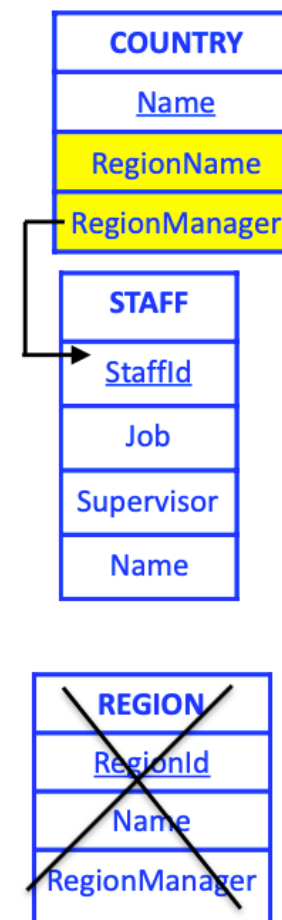
- Relação via chave externa agora OK.
- Contudo **TITLE continua** duplicado
  - Redundância de informação e anomalias do exemplo anterior ainda possíveis, exceto no caso da remoção de um filme.
  - Possível ambiguidade semântica mantém-se.



## Mais exemplos:

### A opção de apagar Relação REGION????

- Significado semântico do esquema não é de todo claro! COUNTRY corresponde a “duas entidades”: REGION “existe escondida” na tabela COUNTRY. Além disso, RegionManager depende na pratica de RegionName mas o esquema não captura essa dependência.
- Anomalias na manipulação de dados: podemos alterar RegionName sem qualquer restrição ou RegionManager de forma independente de RegionName levando a inconsistência dos dados.
- Mesmo que a consistência da BD de alguma forma sobreviva às “intempéries” o esquema é também pouco económico em termos de espaço: RegionName e RegionManager são repetidos para vários países



## Mais exemplos:

### Supondo que:

- Um cliente poderia atribuir opcionalmente uma valorização a um filme refletindo se gostou/não gostou de um filme (escala de 0 a 5) e também varias “tags” a um filme (por exemplo em texto separado por vírgulas como em 'classic, Hitchcock, thriller’).
- que estes dados são gravados na tabela STREAM
- Que grande parte dos clientes não atribuem “ratings” ou “tags” deixando esses atributos a NULL.

**Tudo bem com esta abordagem????????**

STREAM
<u>StreamId</u>
MovieId
CustomerId
StreamDate
Charge
Rating
Tags

## Mais exemplos:

### Tudo bem com esta abordagem???????

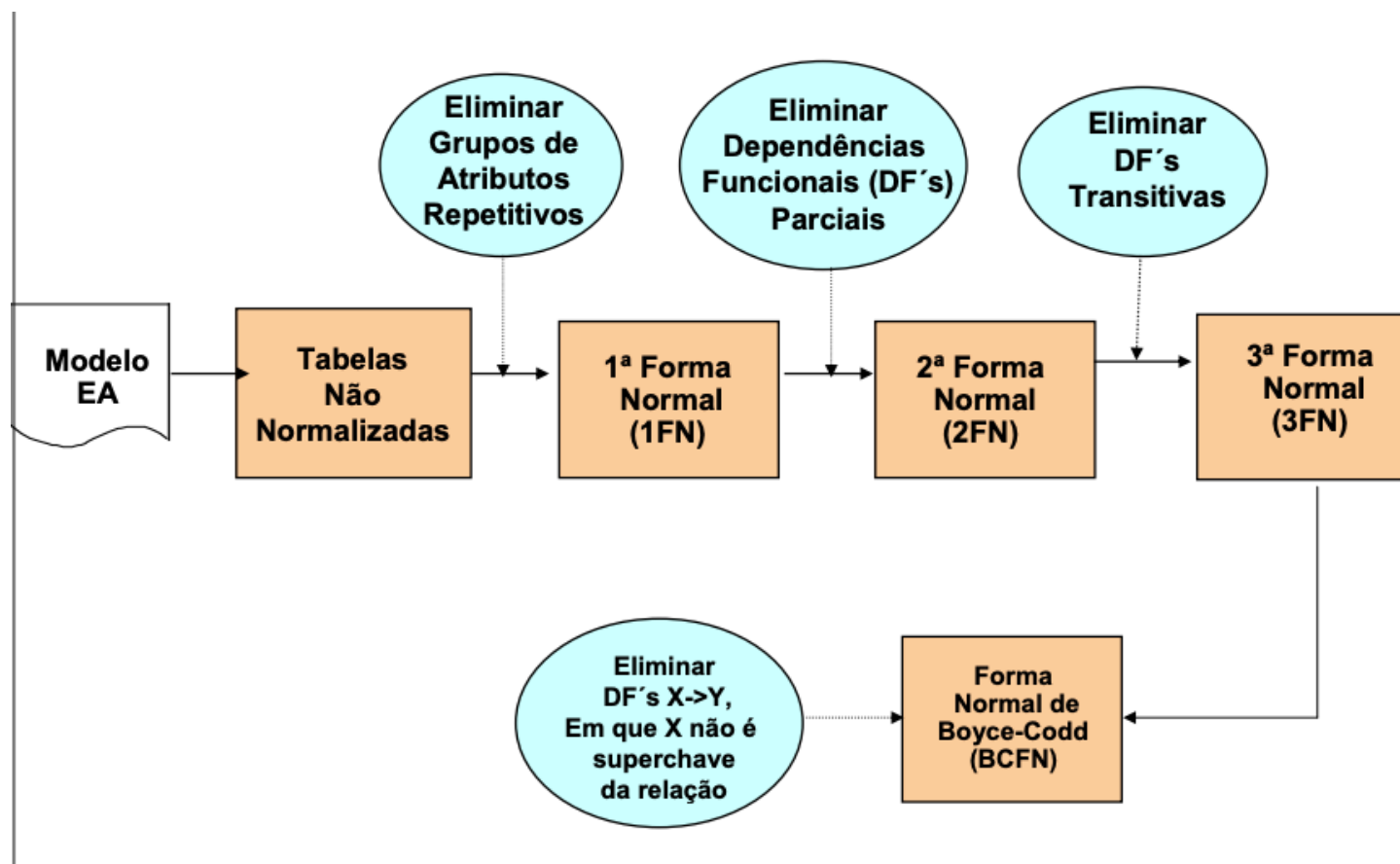
- Demasiados registos com entradas **NULL** levam a desperdício de espaço.
- Dados de “ratings” e “tags” parecem depender do “stream” em vez do cliente o que seria mais natural: faz pouco sentido que o mesmo cliente possa dar vários “ratings” diferentes para o mesmo filme.
- “Ratings” e “tags” podem ser vistas como entidades que merecem existência concreta na BD. Tags é também implicitamente um atributo multivalorado.



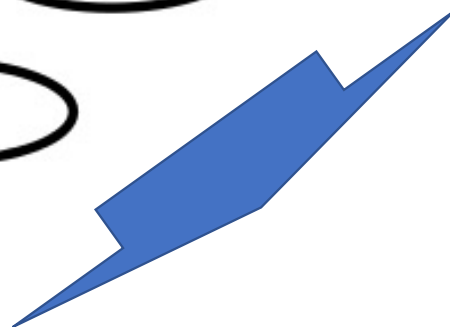
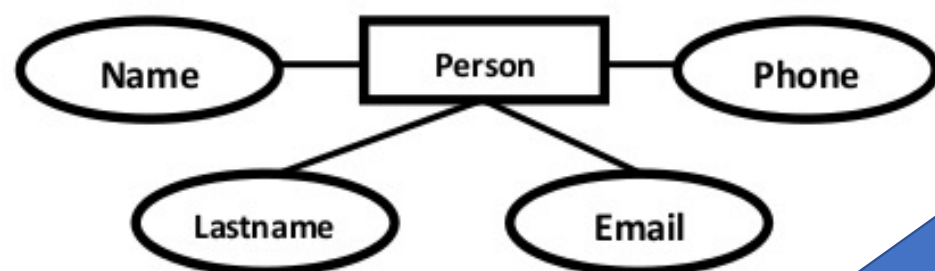
<b>STREAM</b>
<u>StreamId</u>
Movied
CustomerId
StreamDate
Charge
Rating
Tags

**1ª Regra - Integridade de Entidade:** nenhum valor de uma chave primaria pode ser nulo. Em termos de Diagrama Entidade-Associação (ER), o atributo chave primaria deve ser sempre obrigatório, nunca opcional.

**2ª Regra - Integridade Referencial:** numa entidade que possui uma chave estrangeira, cada valor desta chave só pode ser nulo ou igual a algum valor da chave primaria correspondente no relacionamento. A alteração dos valores constituintes da chave primaria ou a remoção de uma instancia que contenha uma chave primaria com uma chave estrangeira associada noutra entidade pode causar problemas de integridade referencial.



## Entidades e Atributos Simples:



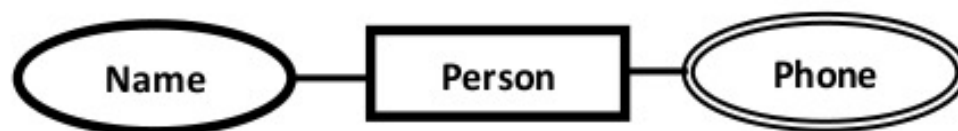
Person( personid , name, lastname, email, phone )

No caso dos atributos compostos, cada elemento será um campo

Person	
PK	<u>personid</u>
	name
	lastname
	email
	phone

<u>personid</u>	name	lastname	email	phone
-----------------	------	----------	-------	-------

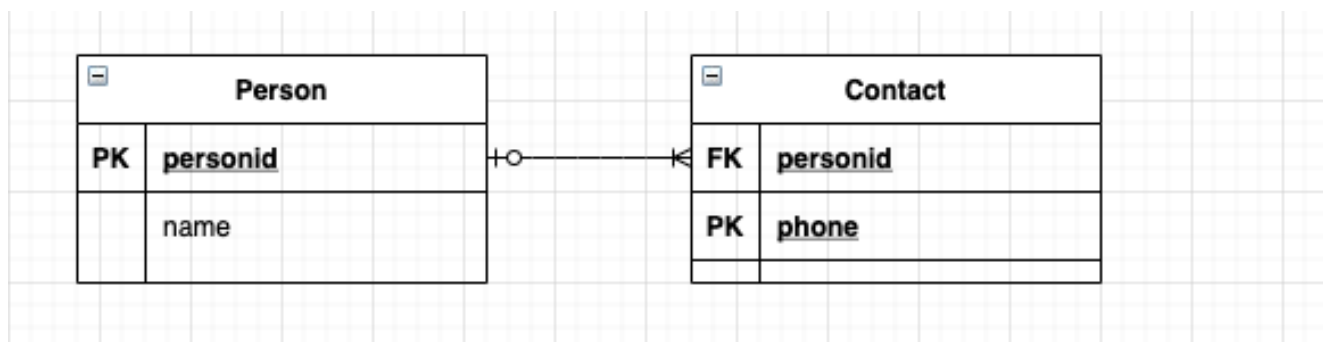
## Atributos Multivalorados:



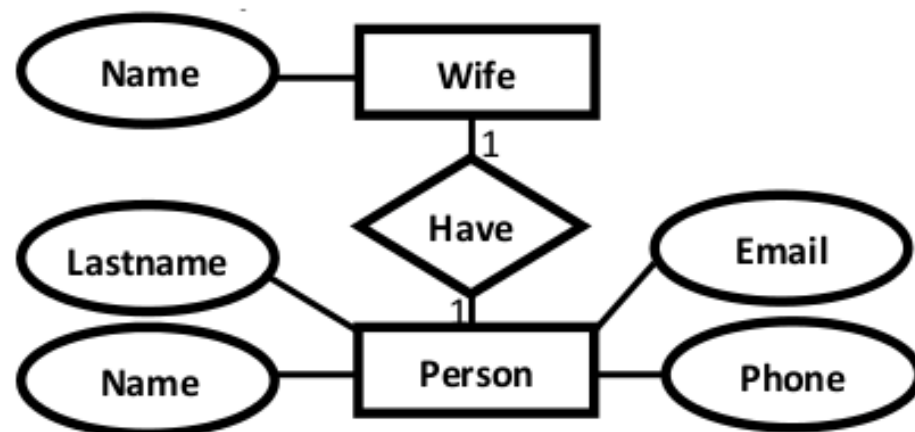
Um atributo multivalorado, dá origem a uma nova tabela, estabelecendo uma relação de 1:N



Person( personid , name)  
 Phones ( *personid*, phone )



## Relações 1:1

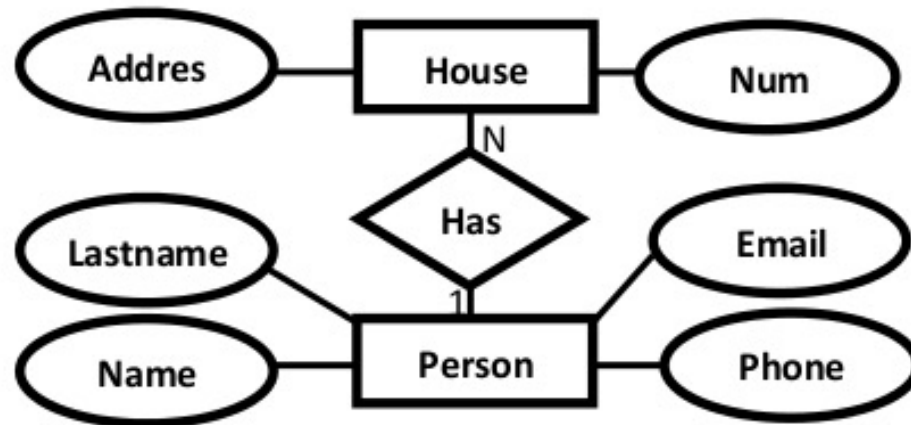


Person( personid , name, lastname, email , **wifeid** )

Wife ( wifeid , name )

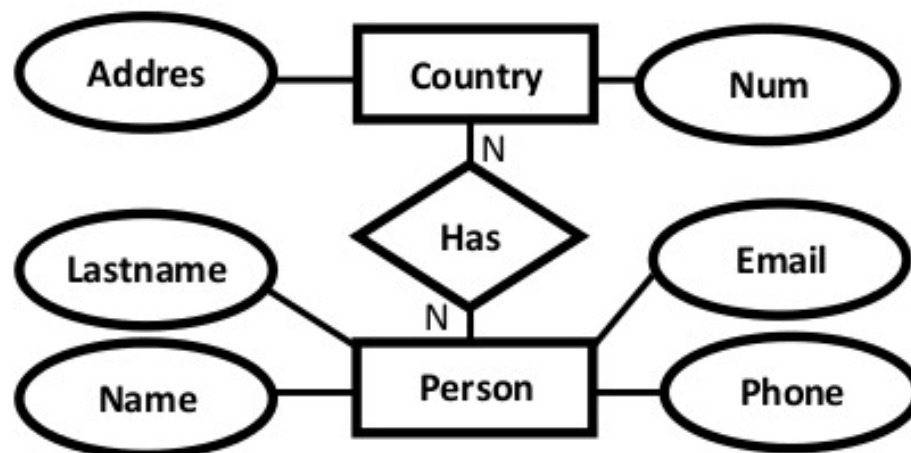


## Relações 1:N



Persons( personid , name, lastname, email )  
 House ( houseid , num , address, **personid** )

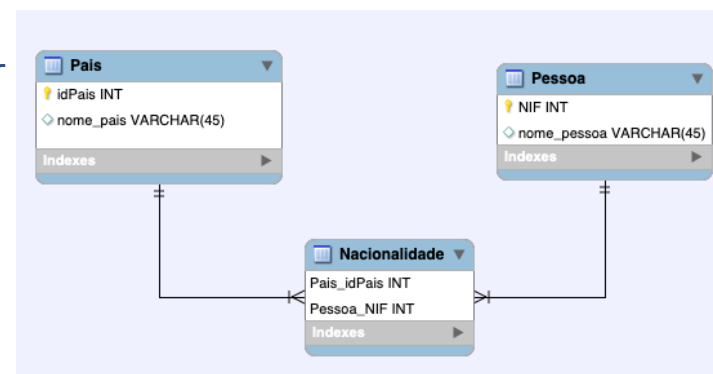
## Relações N:N



Uma relação de N para N terá sempre que criar uma Relação extra (Tabela) que implemente duas relações 1:N e N:1



Person( personid , name, lastname, email )  
 Countrie ( countryid , name, code)  
 HasRelat ( hasrelatid , **personid** , **countryid**)



## Algumas considerações prévias:

Na construção do esquema lógico (não confundir com conceptual), é importante ter em consideração as seguintes diretivas (informais):

- Garantir que a semântica (sentido/significado) dos atributos seja clara no esquema.
- Reduzir a informação redundante nos tuplos (Anomalias de atualização).
- Reduzir os valores NULL nos tuplos.
- Reprovar a possibilidade de gerar tuplos falsos (Anomalias de inserção)

## Conceitos

Processo que visa **corrigir deficiências (Anomalias)** de uma esquema de uma BD por forma a transformá-lo numa Forma Normal (FN).

Ou seja, o processo de identificação dos agrupamentos necessários, e da localização correta de cada atributo consiste **num conjunto de técnicas designadas por normalização**.

**Uma forma normal tem associados:**

- um conjunto de restrições que a definem;
- um processo de transformação de um esquema que não verifica a forma normal, recorrendo a decomposição e/ou transformação de relações.

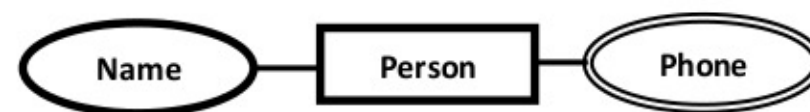
## Modelo Não Normalizado - Exemplo

RELATÓRIO DE ALOCAÇÃO A PROJETO					
CÓDIGO DO PROJETO: LSC001			TIPO: Novo Desenv.		
DESCRIÇÃO: Sistema de Estoque					
CÓDIGO DO EMPREGADO	NOME	CATEGORIA FUNCIONAL	SALÁRIO	DATA DE INÍCIO NO PROJETO	TEMPO ALOCADO AO PROJETO
2146	João	A1	4	1/11/91	24
3145	Sílvio	A2	4	2/10/91	24
6126	José	B1	9	3/10/92	18
1214	Carlos	A2	4	4/10/92	18
8191	Mário	A1	4	1/11/92	12
CÓDIGO DO PROJETO: PAG02			TIPO: Manutenção		
DESCRIÇÃO: Sistema de RH					
CÓDIGO DO EMPREGADO	NOME	CATEGORIA FUNCIONAL	SALÁRIO	DATA DE INÍCIO NO PROJETO	TEMPO ALOCADO AO PROJETO
8191	Mário	A1	4	1/05/93	12
4112	João	A2	4	4/01/91	24
6126	José	B1	9	1/11/92	12

Códproj	Tipo	Descr	Emp					
			CodEmp	Nome	Cat	Sal	DataIni	TempAl
LSC001	Novo Desenvolvimento	Sistema de Estoque	2146	João	A1	4	01/11/91	24
			3145	Sílvio	A2	4	02/10/91	24
			6126	José	B1	9	03/10/92	18
			1214	Carlos	A2	4	04/10/92	18
			8191	Mário	A1	4	01/11/92	12
PAG02	Manutenção	Sistema de RH	8191	Mário	A1	4	01/05/93	12
			4112	joão	A2	4	04/01/91	24
			6126	José	B1	9	01/11/92	12


## Modelo Não Normalizado - Exemplo

codigo_cliente	nome	telefone	endereço
C001	Jorge Jesus	239 990 370 239 990 587	Rua das Camélias, 168 Lousã 3201-909
C002	Rui Vitória	239 991 169	Rua Direita, 478 Lousã 3201-547
C003	Vítor Ferreira	239 067 899 239 067 900	Rua da Formação, 399 Coimbra 3030-481



## Porquê normalizar?

- Minimização de redundâncias e inconsistências;
- Facilidade de manipulações do Banco de Dados;
- Facilidade de manutenção do Sistema de Informação.

1FN	2FN	3FN	BCFN	4FN	5FN
<ul style="list-style-type: none"> <li>• Menos relações</li> <li>• Mais redundância</li> </ul>		 Ponto de equilíbrio		<ul style="list-style-type: none"> <li>• Mais relações</li> <li>• Menos redundância</li> </ul>	

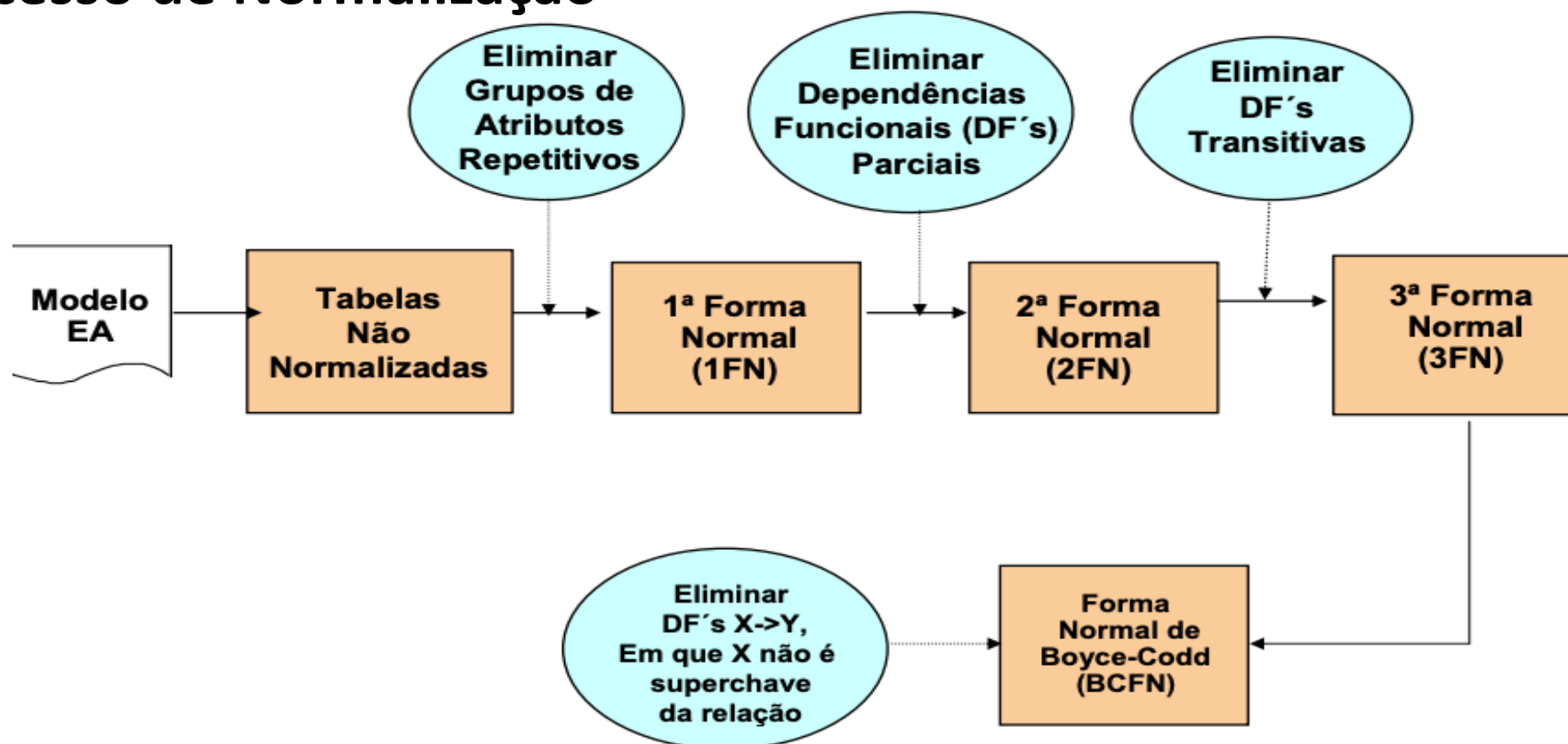
## Anomalias em BD não Normalizadas:

- Anomalias de Inserção;
- Anomalias de exclusão;
- Anomalias de modificação (alterar informação existente);

FUNC_DEP					Redundância	
Fnome	Cpf	Datanasc	Endereco	Dnumero	Dnome	Cpf_gerente
Silva, João B.	12345678966	09-01-1965	Rua das Flores, 751, São Paulo, SP	5	Pesquisa	33344555587
Wong, Fernando T.	33344555587	08-12-1955	Rua da Lapa, 34, São Paulo, SP	5	Pesquisa	33344555587
Zelaya, Alice J.	99988777767	19-01-1968	Rua Souza Lima, 35, Curitiba, PR	4	Administração	98765432168
Souza, Jennifer S.	98765432168	20-06-1941	Av. Arthur de Lima, 54, Santo André, SP	4	Administração	98765432168
Lima, Ronaldo K.	66688444476	15-09-1962	Rua Rebouças, 65, Piracicaba, SP	5	Pesquisa	33344555587
Leite, Joice A.	45345345376	31-07-1972	Av. Lucas Obes, 74, São Paulo, SP	5	Pesquisa	33344555587
Pereira, André V.	98798798733	29-03-1969	Rua Timbira, 35, São Paulo, SP	4	Administração	98765432168
Brito, Jorge E.	88866555576	10-11-1937	Rua do Horto, 35, São Paulo, SP	1	Matriz	88866555576



## Processo de Normalização



## Dependência Funcional – Conceito Formal

Dados dois conjuntos de atributos A e B, de uma entidade, diz-se que:

- B é **funcionalmente dependente** de A, ou
- A **determina** B, ou
- B **depende** de A,

Se, a cada valor de A estiver associado um, e um só, valor de B

Se A determina B, então B não é dependente de nenhum subconjunto de A.

Uma dependência formal é representada formalmente por :

**A → B** sendo A designado por **determinante**

## Dependência Funcional – Exemplos

Considere-se a seguinte Relação (tabela) Funcionário:

N_funcionário	Nome_Próprio	Apelido	Departamento
1021	Sofia	Reis	900
1022	Afonso	Reis	700
1023	António	Cardoso	900

**N\_Funcionário → Departamento ?**

**Sim**, N\_Funcionário é determinante de Departamento, ou Departamento depende funcionalmente de N\_Funcionário **porque é possível determinar o Departamento a partir do Número de Funcionário** (*Considera-se, neste exemplo que um funcionário só pode pertencer a um departamento* )

## Dependência Funcional – Exemplos

Considere-se a seguinte Relação (tabela) Funcionário:

N_funcionário	Nome_Próprio	Apelido	Departamento
1021	Sofia	Reis	900
1022	Afonso	Reis	700
1023	António	Cardoso	900

**Departamento → N\_funcionário ?**

**Não**, pois, por exemplo o Departamento 900 temos dois funcionários  
**900 => {1021; 1023}**

## Dependência Funcional – Exemplos

Considere-se a seguinte Relação (tabela) Funcionário:

N_funcionário	Nome_Próprio	Apelido	Departamento
1021	Sofia	Reis	900
1022	Afonso	Reis	700
1023	António	Cardoso	900

**N\_Funcionário → Apelido ?**

**Sim**, pois, mesmo havendo mais que um funcionário com o mesmo apelido, o N\_funcionário **determina** um, e um só, Apelido.

**1021 => “Reis”**

## Dependência Funcional – Exemplos

Considere-se a seguinte Relação (tabela) Funcionário:

N_funcionário	Nome_Próprio	Apelido	Departamento
1021	Sofia	Reis	900
1022	Afonso	Reis	700
1023	António	Cardoso	900

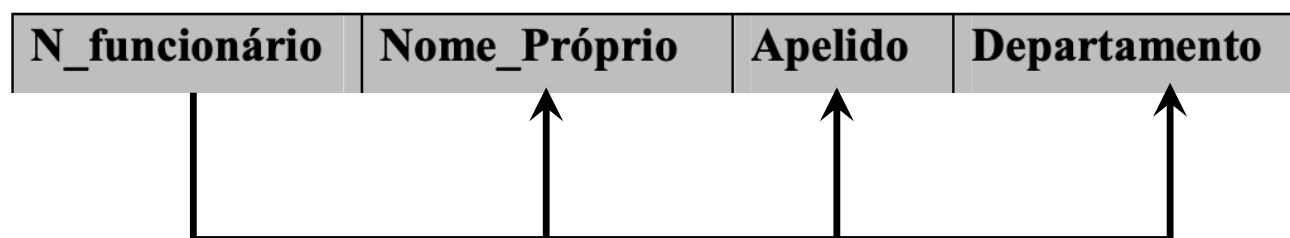
**Nome\_Próprio → N\_Funcionário ?**

**Não**, pois, podem haver vários Funcionários com o mesmo nome próprio, o que levaria a serem determinados vários N\_Funcionário através do atributo Nome\_Próprio

“Luís” => {1025; 1367;etc}

## Dependência Funcional – Exemplos

Considere-se a seguinte Relação (tabela) Funcionário:



**N\_Funcinário → Todos os outros atributos?**

**Sim**, na realidade através do atributo N\_Funcionário pode-se **determinar** qualquer um dos restantes atributos, logo os restantes atributos **são Funcionalmente Dependentes** de N\_Funcionário

Graficamente pode-se representar como na imagem acima:

## Dependência Funcional – Mais Exemplos

Considere-se a seguinte Relação denominada Inventário (Mapeamento do preço de diferentes artigos em diferentes papelarias):

Papelaria	Artigo	Preço
Colmeia	Caneta bic fina	150
Central	Fita cola	300
Aguarela	Borracha	215
Silva	Caneta bic fina	175

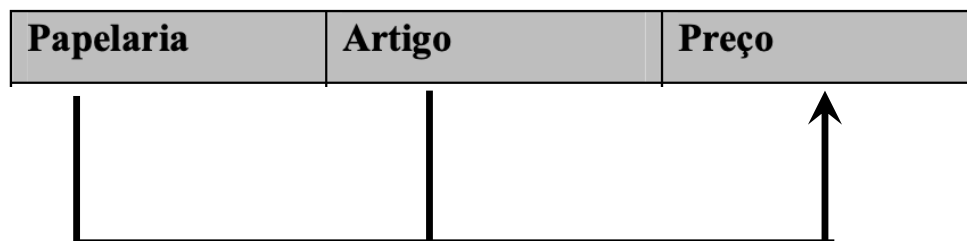
**Artigo → Preço** (O preço é funcionalmente dependente de Artigo?)

**Não**, o preço do mesmo artigo pode ser distinto em Papelarias distintas



## Dependência Funcional – Mais Exemplos

Considere-se a seguinte Relação denominada Inventário (Mapeamento do preço de diferentes artigos em diferentes papelarias):




**{Papelaria,Artigo} → Preço** (O preço é funcionalmente dependente de Papelaria e Artigo?)

**Sim**, para cada produto vendido numa Papelaria existe um só preço, ou seja o valor de **Papelaria** e **Artigo**, em conjunto, **determinam** o preço

## Formas Normais

Formas normais, com restrições progressivamente mais fortes:

- **1NF**: 1ª forma normal (“1st Normal Form”)
- **2NF**: 2ª forma normal (“2nd Normal Form”)
- **3NF**: 3ª forma normal (“3rd Normal Form”)
- **BCNF**: Forma normal de Boyce–Codd (“Boyce-Codd Normal Form”)
- Formas normais mais fortes que BCNF (4NF, 5NF, 6NF, 7NF!) são, normalmente, pouco consideradas/práticas (Complexidade crescente).
- Esquemas 3NF são quase sempre também BCNF.

1FN	2FN	3FN	BCFN	4FN	5FN
<ul style="list-style-type: none"> <li>• Menos relações</li> <li>• Mais redundância</li> </ul>		 Ponto de equilíbrio		<ul style="list-style-type: none"> <li>• Mais relações</li> <li>• Menos redundância</li> </ul>	

## 1ª Forma Normal

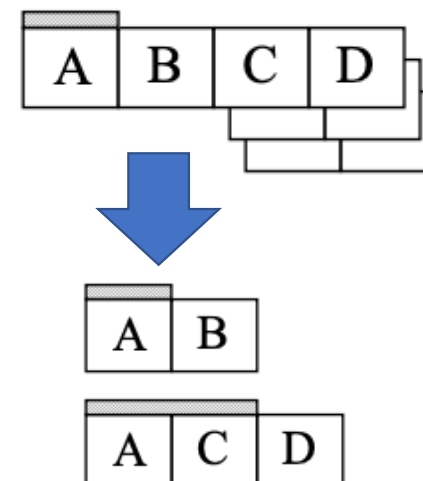
Uma relação estará na primeira forma normal (1NF ou 1FN) se:

- Possuir uma **chave primária** identificada;
- **Não há conjuntos de atributos repetidos** descrevendo a mesma característica
- Todos os atributos de relações **forem atômicos**.

*Não são permitidos atributos que implicitamente codificam subatributos compostos ou atributos multi-valor (ou multivalorados).*

### Processo de Conversão para a 1ª Forma Normal

- Uma das chaves candidatas é escolhida para chave primaria.
- Atributos multi-valor implícitos convertidos em nova relação (*tabela*) com chave externa referindo a chave primaria da tabela original.
- Cada atributo composto implícito é mapeado em vários subatributos atômicos.



## 1ª Forma Normal - Exemplos

PessoaCursos1

Nome	Endereço	NIF	Cursos
Artur	Covilhã	123456789	Programador
Ana	Fundão	222222222	Operador, Programador
Carlos	Covilhã	222333444	Analista, Programador, Operador
Paulo	Guarda	555666777	Operador, Analista

**Está na 1NF ?**

**Não**, pois existem atributos não atómicos. Uma pessoa pode ter mais que um curso

## 1ª Forma Normal - Exemplos

PessoaCursos2

Nome	Endereço	<u>NIF</u>	Curso1	Curso2	Curso3
Artur	Covilhã	123456789	Programador		
Ana	Fundão	222222222	Operador	Programador	
Carlos	Covilhã	222333444	Analista	Programador	Operador
Paulo	Guarda	555666777	Operador	Analista	

Está na 1NF ?

- Possuir uma **chave primaria** identificada;
- **Não há conjuntos de atributos repetidos** descrevendo a mesma característica
- Todos os atributos de relações **forem atômicos**

**Não**, pois:

- São repetidos atributos do mesmo tipo, curso1, curso2, curso3. (Diz-se que a relação tem um grupo repetitivo)
- Os tuplos correspondentes a alunos com apenas 1 ou dois cursos vão ter valores nulos para alguns atributos
- **Como representar uma pessoa com mais do que três cursos?**

atri

A	B	C
---	---	---

## 1ª Forma Normal - Exemplos

Nome	Endereço	<u>NIF</u>
Artur	Covilhã	123456789
Ana	Fundão	222222222
Carlos	Covilhã	222333444
Paulo	Guarda	555666777

<u>NIF</u>	<u>Curso</u>
123456789	Programador
222222222	Operador
222222222	Programador
555666777	Analista
222333444	Operador
****	****

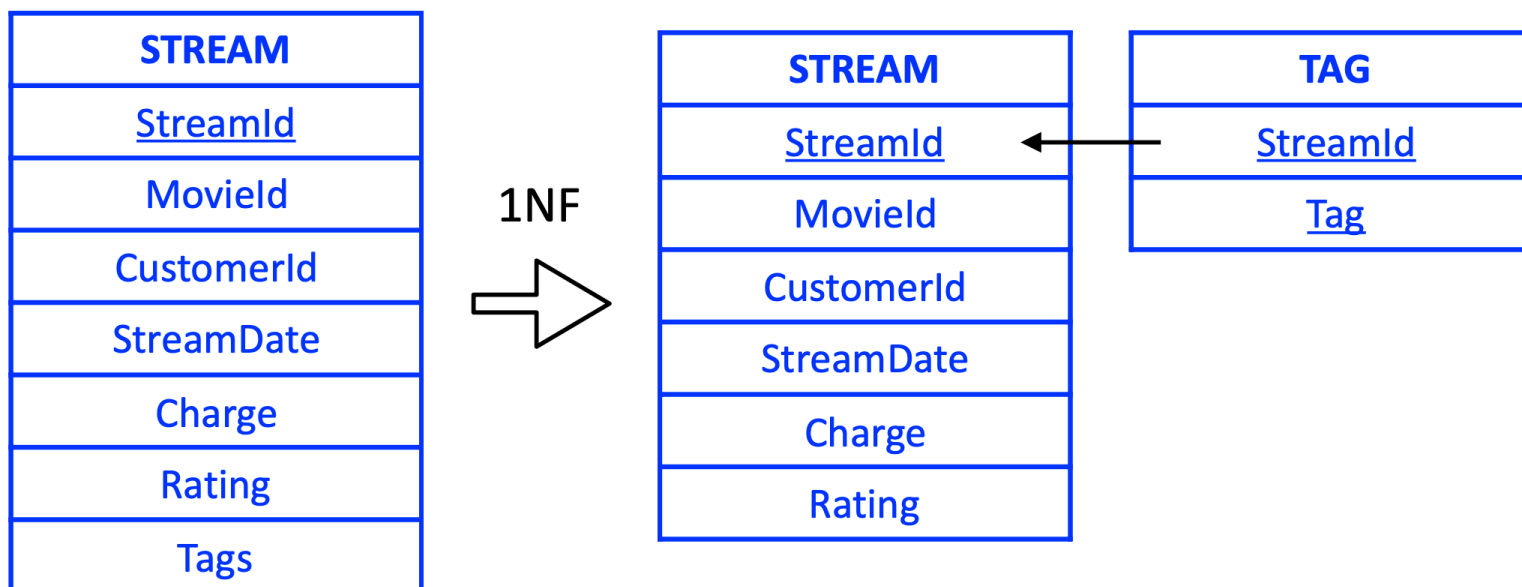
PessoaCursos1

Nome	Endereço	NIF	Cursos
Artur	Covilhã	123456789	Programador
Ana	Fundão	222222222	Operador, Programador
Carlos	Covilhã	222333444	Analista, Programador, Operador
Paulo	Guarda	555666777	Operador, Analista

Está na 1NF ?

**Sim.** Atributos multi-valor dão origem a uma nova tabela estabelecendo uma relação de 1:N entre ambas

## 1ª Forma Normal - Exemplos



Relação original decomposta em duas em função do atributo “multi-valor” implícito (tags).

## 1ª Forma Normal - Exemplos

codigo_cliente	nome	telefone	endereço
C001	Jorge Jesus	239 990 370 239 990 587	Rua das Camélias, 168 Lousã 3201-909
C002	Rui Vitória	239 991 169	Rua Direita, 478 Lousã 3201-547
C003	Vitor Ferreira	239 067 899 239 067 900	Rua da Formação, 399 Coimbra 3030-481

**Está na 1NF ?**

**Não.** Atributos multi-valor e subatributos aninhados (endereço)



## 1ª Forma Normal - Exemplos

codigo_cliente	p_nome	apelido	telefone	rua	cidade	codigo_postal
C001	Jorge	Jesus	239 990 370 239 990 587	Rua das Camélias, 168	Lousã	3201-909
C002	Rui	Vitória	239 991 169	Rua Direita, 478	Lousã	3201-547
C003	Vítor	Ferreira	239 998 899 239 998 900	Rua da Formação, 399	Coimbra	3030-481

**Está na 1NF ?**

**Não.** Atributos multi-valor

## 1ª Forma Normal - Exemplos

codigo_cliente	p_nome	apelido	rua	cidade	codigo_postal
C001	Jorge	Jesus	Rua das Camélias, 168	Lousã	3201-909
C002	Rui	Vitória	Rua Direita, 478	Lousã	3201-547
C003	Vítor	Ferreira	Rua da Formação, 399	Coimbra	3030-481

codigo_cliente	telefone
C001	239 990 370
C001	239 990 587
C002	239 991 169
C003	239 998 899
C003	239 998 900

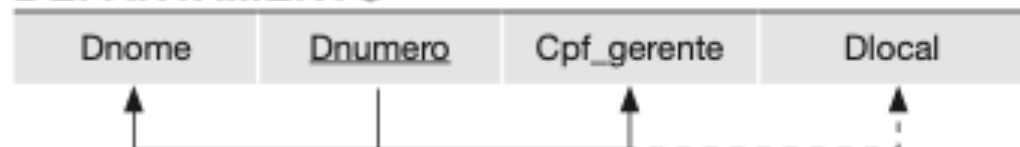


Está na 1NF ?

Sim

## 1ª Forma Normal - Exemplos

### DEPARTAMENTO



### DEPARTAMENTO

Dnome	<u>Dnumero</u>	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André, Itu, São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

Sim, está na 1FN (mas não é uma boa opção...)

### DEPARTAMENTO

Dnome	<u>Dnumero</u>	Cpf_gerente	Dlocal
Pesquisa	5	33344555587	Santo André
Pesquisa	5	33344555587	Itu
Pesquisa	5	33344555587	São Paulo
Administração	4	98765432168	Mauá
Matriz	1	88866555576	São Paulo

## 1ª Forma Normal – Problemas. Porquê continuar a normalizar...

NºEstudante	Nome Estudante	Curso	Nº Disciplina	Nome Disciplina	Cod Profe	Nº Professor	Nome Professor	Nota
-------------	----------------	-------	---------------	-----------------	-----------	--------------	----------------	------

↓

**Relação ESTUDANTE**

NºEstud.	Nome Estudante	Curso	Nº Disciplina	Nome Disciplina	Cod Profe	Nº Professor	Nome Professor	Nota
21934	...	...	08	Topologia	32	Nuno Neves	10	10
42346	...	...	04	Álgebra	21	Gil Alves	12	12
42346	...	...	12	Geometria	21	Gil Alves	18	18
42346	...	...	16	Lógica	32	Nuno Neves	13	13
54323	...	...	04	Álgebra	21	Gil Alves	11	11
54323	...	...	08	Topologia	32	Nuno Neves	10	10
...	...	...	...	...	...	...	...	...

**Relação NOTAS**

Nº Disciplina	Nome Disciplina	Cod Profe	Nº Professor	Nome Professor	Nota
08	Topologia	32	Nuno Neves	10	10
04	Álgebra	21	Gil Alves	12	12
12	Geometria	21	Gil Alves	18	18
16	Lógica	32	Nuno Neves	13	13
04	Álgebra	21	Gil Alves	11	11
08	Topologia	32	Nuno Neves	10	10
...	...	...	...	...	...

Assumimos que uma Disciplina tem um só Professor (Responsável) (Nº\_Disciplina → Cod\_Professor), mas um Professor pode ser responsável por várias Disciplinas

## 1ª Forma Normal – Problemas. Porquê continuar a normalizar ....

As tabelas normalizadas apenas na 1FN podem apresentar problemas de redundância de informação, bem como outros tipos de problemas ou anomalias, nomeadamente:

- **anomalias de inserção** – em certas situações, a inserção de um novo registo pode implicar um ou mais campos em branco;
- **anomalias de atualização** – dado que numa tabela pode existir repetição de informação sobre um mesmo elemento, se quisermos atualizar um dado relativo a um determinado elemento (por exemplo a morada de um cliente), teremos de efetuar tantas atualizações quantas as vezes que esse elemento aparecer na tabela.
- **anomalias de eliminação** – pela mesma razão anteriormente apontada, ou seja, a possibilidade de existir informação repetida sobre um mesmo elemento, se quisermos apagar um determinado elemento da nossa base de dados, teremos de o fazer a todos os registos em que ele figurar

## 2ª Forma Normal

Uma tabela está na **Segunda Forma Normal (2FN)** se:

- Estiver na 1FN;
- Todos os atributos não chave forem totalmente dependentes da chave primaria (*dependente de toda a chave e não apenas de parte dela*);

### Processo de Conversão para a 2ª Forma Normal

- Identificar os atributos que não são funcionalmente dependentes de toda a chave primaria;
- Remover da Entidade todos os atributos identificados e criar uma nova entidade com eles.

## 2ª Forma Normal: Exemplos

Uma tabela está na **Segunda Forma Normal (2FN)** se:

- Estiver na 1FN;
- Todos os atributos não chave forem totalmente dependentes da chave primária (*dependente de toda a chave e não apenas de parte dela*);



n_pedido	c_produto	produto	quant	valor_unitario	subtotal
1005	239-782	Leitão à Bairrada	16	17,50	280,00
1006	239-902	Chanfana	18	15,50	279,00
1007	239-908	Vitela à Lafões	10	10,20	102,00
1008	239-451	Posta à Mirandesa	25	14,40	360,00

**n\_pedido** → {c\_produto , produto , quant, valor\_unitário , subtotal}

**c\_produto** → **produto**

## 2ª Forma Normal: Exemplos

c_produto	produto
239-782	Leitão à Bairrada
239-902	Chanfana
239-908	Vitela à Lafões
239-451	Posta à Mirandesa

Tabela 6 - Tabela Produto (Normalizada – 2FN)

**c\_produto → produto**

n_pedido	c_produto	quant	valor_unitario	subtotal
1005	239-782	16	17,50	280,00
1006	239-902	18	15,50	279,00
1007	239-908	10	10,20	102,00
1008	239-451	25	14,40	360,00

Tabela 7 - Tabela Vendas (Normalizada – 2FN)

**n\_pedido → {c\_produto , produto , quant, valor\_unitário , subtotal}**

Resumindo está na **Segunda Forma Normal(2FN)** quando está na 1FN e não existem dependências funcionais entre os atributos não chave, ou seja, cada atributo deve depender apenas da chave primaria da relação.



## 3ª Forma Normal

Uma tabela está na **Terceira Forma Normal (3FN)** se:

- Estiver na 2FN;
- Se nenhum atributo (coluna) não chave depender de outra coluna não chave

Na terceira forma normal temos de eliminar todos aqueles campos que podem ser obtidos pela equação de outros campos da mesma tabela.

### Processo de Conversão para a 3ª Forma Normal

- Identificar todos os atributos que são funcionalmente dependentes de outros atributos não chave;
- Remover esses atributos (*Os tais atributos derivados dos Diagramas EA*)

## 3ª Forma Normal: Exemplos

c_produto	produto
239-782	Leitão à Bairrada
239-902	Chanfana
239-908	Vitela à Lafões
239-451	Posta à Mirandesa

Tabela 6 - Tabela Produto (Normalizada – 2FN)

n_pedido	c_produto	quant	valor_unitario	subtotal
1005	239-782	16	17,50	280,00
1006	239-902	18	15,50	279,00
1007	239-908	10	10,20	102,00
1008	239-451	25	14,40	360,00

Tabela 7 - Tabela Vendas (Normalizada – 2FN)



n_pedido	c_produto	quant	valor_unitario
1005	239-782	16	17,50
1006	239-902	18	15,50
1007	239-908	10	10,20
1008	239-451	25	14,40

Resumindo uma tabela na **Terceira Forma Normal (3FN)** é uma tabela em que as relações, além de estar na 2FN, não existem dependências funcionais entre os atributos não chave. Ou seja, **cada atributo deve depender apenas da chave primaria** da relação. A solução é decompor a relação em duas relações de acordo com as dependências funcionais existentes.

## Exemplo Completo

ORDEM_COMPRA
cod_ordem_compra
data_emissão
cod_fornecedor
num_fornecedor
endereço_fornecedor
cod_material
descricao_material
qtd_comprada
valor_total_item
valor_totalordem

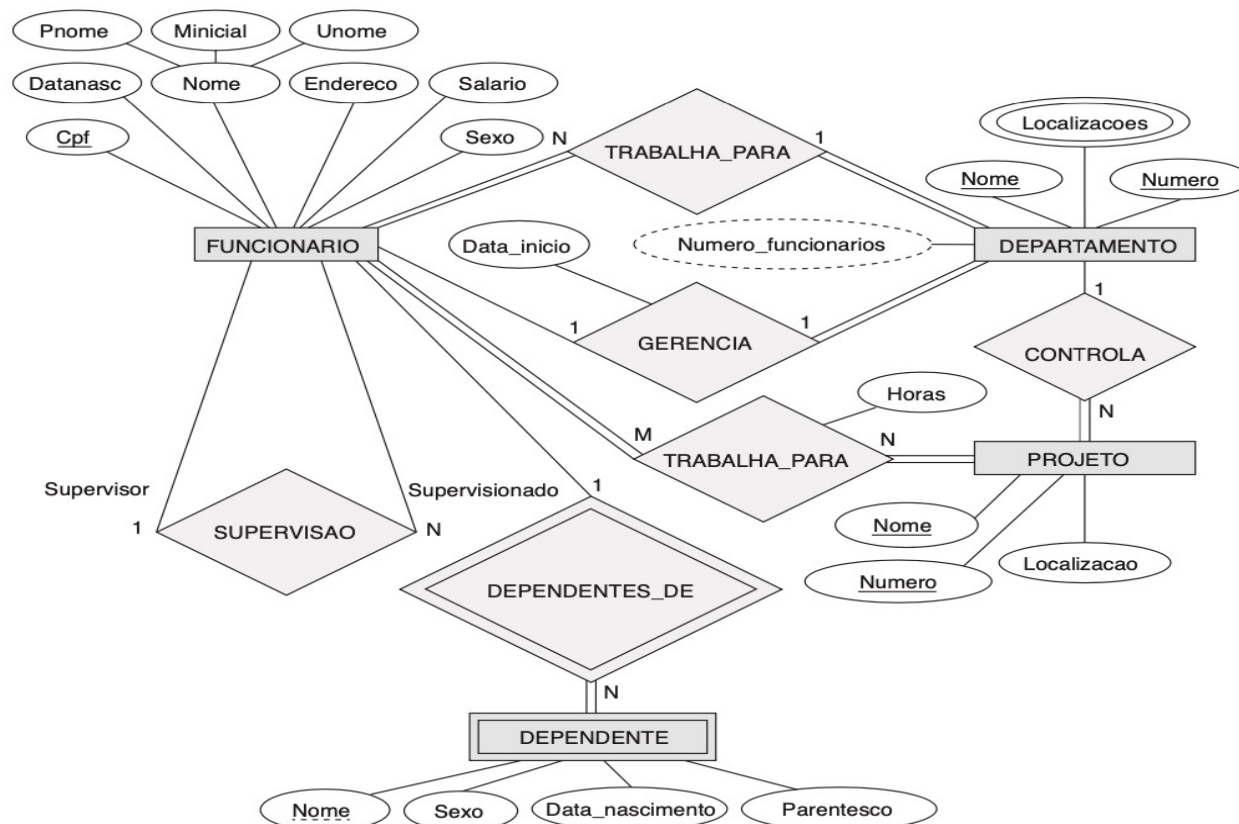


1FN
ORDEM_COMPRA
<u>cod_ordem_compra</u>
data_emissão
cod_fornecedor
num_fornecedor
endereço_fornecedor
valor_totalordem
MATERIAL
<u>cod_material</u>
descricao_material
qtd_comprada
valor_unitario
valor_total_item
<u>cod_ordem_compra</u>

2FN (DF da Chave)
ORDEM_COMPRA
<u>cod_ordem_compra</u>
data_emissão
cod_fornecedor
num_fornecedor
endereço_fornecedor
valor_totalordem
MATERIAL
<u>cod_material</u>
qtd_comprada
valor_total_item
<u>cod_ordem_compra</u>
DETALHES_MATERIAL
<u>cod_material</u>
descricao_material
valor_unitario

3FN (DF Transitiva)
ORDEM_COMPRA
<u>cod_ordem_compra</u>
data_emissão
cod_fornecedor
FORNECEDOR
<u>cod_fornecedor</u>
num_fornecedor
endereço_fornecedor
MATERIAL
<u>cod_material</u>
qtd_comprada
valor_total_item
<u>cod_ordem_compra</u>
DETALHES_MATERIAL
<u>cod_material</u>
descricao_material
valor_unitario

## Diagrama EA da Empresa



## Modelo Lógico (3FN)

