

# Álgebra Relacional

---

Carina F. Dorneles  
dorneles@inf.ufsc.br

Parte II



# Relembrando...

- **Seleção** de linhas, usando condições
    - Seleção -  $\sigma$
  - **Projeção** de colunas
    - Projeção -  $\pi$
  - **Junção** de tabelas
    - Produto Cartesiano -  $\times$
  - **Alteração dos nomes** de tabelas e atributos
    - Renomeação -  $\rho$
-

# Operadores

- Seleção -  $\sigma$
  - Projeção -  $\pi$
  - Produto Cartesiano -  $\times$
  - Renomeação -  $\rho$
  - Junção -  $\bowtie$
  - Diferença -  $-$
  - União -  $\cup$
  - Intersecção -  $\cap$
  - Divisão -  $\div$
-

# Banco de dados Exemplo



## Livro

<u>Codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700

## Pessoa

<u>Codigo</u>	Nome	Idade	fone	CodEsposa	Sexo
PE02	Aninha	23	9999.9999	NULL	F
PE10	Paulinho	20	8888.8888	NULL	M
PE87	Juca	34	7777.7777	PE02	M
PE23	Luana	30	6666.6666	NULL	F
PE54	Beto	28	5555.5555	PE23	M

## Empréstimo

<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>	<u>Responsavel</u>
LI005	PE02	10/10/2000	8:00	PE02
LI670	PE02	10/10/2000	8:00	PE02
LI340	PE23	01/11/2000	11:50	NULL
LI003	PE54	20/11/2000	10:00	NULL
LI005	PE10	11/11/2000	14:00	PE10
LI670	PE87	23/05/2001	16:15	PE10

# Junção – $\bowtie_{\theta}$

- Junta tuplas das relações, dada uma condição de junção – operação **binária**
- Sintaxe:

`<relação_1>  $\bowtie_{\theta}$  <relação_2>`

- Onde:
    - `<relação_i>`: nome da relação que se deseja recuperar dados
    - $\Theta$  = `<condicao de junção>`, que é uma expressão booleana que envolve literais e valores de atributos da tabela
  - O parâmetro `<relação>` pode ser outra expressão algébrica, pois uma expressão algébrica retorna uma relação
-

# Junção - exemplo

- Q1: Recuperar título e ano dos livros emprestados

Livro

<u>Codigo</u>	<u>Título</u>	<u>Ano</u>	<u>NrPaginas</u>
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700
LI999	Introdução à Computação	2010	200

Empréstimo

<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	PE02	10/10/2000	8:00
LI670	PE02	10/10/2000	8:00
LI340	PE23	01/11/2000	11:50
LI003	PE54	20/11/2000	10:00
LI005	PE10	11/11/2000	14:00
LI670	PE87	23/05/2001	16:15

## Junção - exemplo

- Q1: Recuperar título e ano dos livros emprestados

$\pi$  titulo, ano (Livro  `codigo = codLivro` Emprestimo)

# Junção vs Produto Cartesiano - exemplo

- Q1: Recuperar título e ano dos livros emprestados

$\pi$  titulo, ano ( $\sigma$  codigo = codLivro (Livro  $\times$  Emprestimo))

$\pi$  titulo, ano (Livro  $\bowtie$  codigo = codLivro Emprestimo)

São necessários  
dois operadores



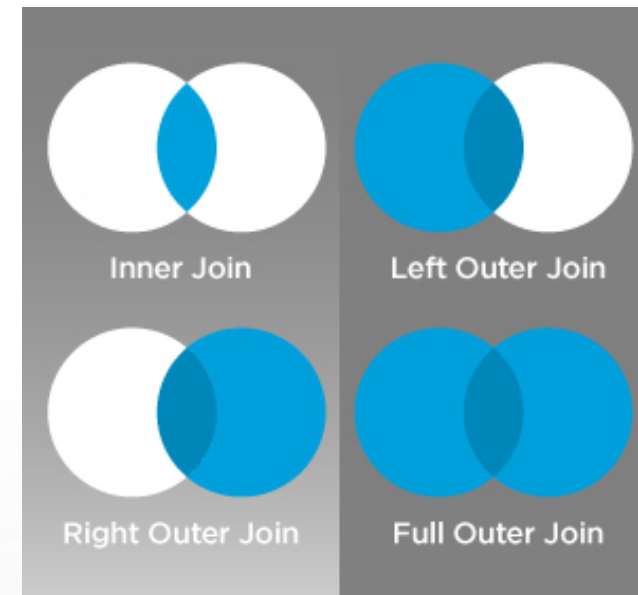
# Tipos de Junção

- Inner Join

- [Inner] Join  $\bowtie_{\theta}$
- Natural Join  $\bowtie$

- Outer Join

- Left [Outer] Join  $\bowtie_{\leftarrow}$
- Right [Outer] Join  $\bowtie_{\rightarrow}$
- Full [Outer] Join  $\bowtie_{\overline{\phantom{x}}}$



# Junção natural – ⋈

- **Junta** tuplas das relações, pela igualdade dos valores de atributos de mesmo nome – operação **binária**
- Sintaxe:

`<relação_1> ⋈ <relação_2>`

- Onde:
    - `<relação_i>`: **nome da relação** que se deseja recuperar dados
  - O parâmetro `<relação>` **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação
-

# Junção natural - exemplo

- Q1: Recuperar titulo e ano dos livros emprestados

Livro

<u>CodLivro</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700
LI999	Introdução à Computação	2010	200

CodLivro

# Junção natural - exemplo

- Q1: Recuperar título e ano dos livros emprestados

Atributos de mesmo nome

Livro

<u>CodLivro</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700
LI999	Introdução à Computação	2010	200

Empréstimo

<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	PE02	10/10/2000	8:00
LI670	PE02	10/10/2000	8:00
LI340	PE23	01/11/2000	11:50
LI003	PE54	20/11/2000	10:00
LI005	PE10	11/11/2000	14:00
LI670	PE87	23/05/2001	16:15

## Junção natural - exemplo

- Q1: Recuperar título e ano dos livros emprestados

$\pi$  titulo, ano (Livro  $\bowtie$  Emprestimo)

---

# Junção natural - exemplo

- Q1: Recuperar titulo e ano dos livros emprestados

$\pi$  titulo, ano (Livro  Empréstimo)

Usará como **condição de junção** a igualdade entre os **atributos de mesmo nome**



## Junção natural - exemplo

- Q1: Recuperar titulo e ano dos livros emprestados

Se as tabelas tiverem mais de um atributo de mesmo nome, todos serão considerados

mesmo nome



# INNER JOIN

- Recupera tuplas que satisfaçam a condição de junção
- No caso do Natural join, a condição de junção é dada pela igualdade dos valores de atributos de mesmo nome
  - Q2: Livros emprestados:

Livro

<u>Codig o</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700
LI888	Computação e Engenharia	20112	1050
LI777	Arquitetura de Computadores	2000	890
LI999	Introdução à Computação	2010	200

Empréstimo

<u>CodLivr o</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	PE02	10/10/2000	8:00
LI670	PE02	10/10/2000	8:00
LI340	PE23	01/11/2000	11:50
LI003	PE54	20/11/2000	10:00
LI005	PE10	11/11/2000	14:00
LI670	PE87	23/05/2001	16:15



# INNER JOIN

- Recupera tuplas que satisfaçam a condição de junção
- No caso do Natural join, a condição de junção é dada pela igualdade dos valores de atributos de mesmo nome
  - Q2: Livros emprestados:

Livro

<u>codigo</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
LI003	Algoritmos e Lógica	2000	700
LI835	Computação e Engenharia	2011	1050
LI777	Arquitetura de Computadores	2000	890
LI999	Introdução à Computação	2010	200

Empréstimo

<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>
LI005	PE02	10/10/2000	8:00
LI670	PE02	10/10/2000	8:00
LI340	PE23	01/11/2000	11:50
LI003	PE54	20/11/2000	10:00
LI005	PE10	11/11/2000	14:00
LI670	PE87	23/05/2001	16:15




Não estarão no resultado do INNER JOIN  
pois a condição para junção seria `codigo=codlivro`

# Junção Externa

- Algumas vezes, também é necessário saber **quais as tuplas que não satisfazem a condição**
  - Q3: **titulo de todos os livros e datas** de empréstimo daqueles que foram **emprestados**

Título	Data_Emprestimo
Web e Banco de dados	10/10/2000
Web e Banco de dados	11/11/2000
Introdução a Banco de Dados	10/10/2000
Introdução a Banco de Dados	23/05/2001
Programação C	01/11/2000
Algoritmos e Lógica	20/11/2000
Computação e Engenharia	NULL
Arquitetura de Computadores	NULL
Introdução à Computação	NULL

# Junção Externa

- Outer Join
    - Left [Outer] Join 
    - Right [Outer] Join 
    - Full [Outer] Join 
-

# LEFT [OUTER] JOIN

- Obtém como resultado uma relação que possui:
  - as tuplas que obedecem a condição de junção
  - as tuplas da relação à esquerda do operador que não estão na relação à direita
- Sintaxe:

`<relação_1>  <relação_2>`

- Onde:
    - `<relação_i>`: nome da relação que se deseja recuperar dados
-

# LEFT JOIN - Exemplo

- Q3: titulo de todos os livros e datas de empréstimo daqueles que foram emprestados

Título	Data_Emprestimo
Web e Banco de dados	10/10/2000
Web e Banco de dados	

# RIGHT [OUTER] JOIN $\theta$

- Obtém como resultado uma relação que possui:
  - as tuplas que obedecem a condição de junção
  - as tuplas da relação à direita do operador que não estão na relação à esquerda
- Sintaxe:

`<relação_1>  <relação_2>`  
 $\theta$

- Onde:
    - `<relação_i>`: nome da relação que se deseja recuperar dados
-

# RIGHT JOIN - Exemplo

- Q3: titulo de todos os livros e datas de empréstimo daqueles que foram emprestados

$\pi$  titulo, data (Emprestimo  $\bowtie$  `codigo = codLivro` Livro)

Título	Data_Emprestimo
Web e Banco de dados	10/10/2000
Web e Banco de dados	11/11/2000
Introdução a Banco de Dados	10/10/2000
Introdução a Banco de Dados	23/05/2001
Programação C	01/11/2000
Algoritmos e Lógica	20/11/2000
Computação e Engenharia	NULL
Arquitetura de Computadores	NULL
Introdução à Computação	NULL

# FULL [OUTER] JOIN

- Obtém como resultado uma relação que possui:
  - as tuplas que obedecem a condição de junção
  - as tuplas da relação à direita do operador que não estão na relação à esquerda
  - as tuplas da relação à esquerda do operador que não estão na relação à direita
- Sintaxe:

`<relação_1>  <relação_2>`

- Onde:
    - `<relação_i>`: nome da relação que se deseja recuperar dados
-



# FULL JOIN - Exemplo

- Q3: Datas de empréstimos e nomes dos responsáveis. Aqueles empréstimos sem responsáveis, e pessoas não responsabilizadas por empréstimos também devem aparecer no resultado

Livro


<u>CodLivro</u>	<u>CodPessoa</u>	<u>Data</u>	<u>Hora</u>	<u>Responsavel</u>
LI005	PE02	10/10/2000	8:00	PE02
LI670	PE02	10/10/2000	8:00	PE02
LI340	PE23	01/11/2000	11:50	NULL
LI003	PE54	20/11/2000	10:00	NULL
LI005	PE10	11/11/2000	14:00	PE10
LI670	PE87	23/05/2001	16:15	PE10

Pessoa

<u>Codig o</u>	Nome	Idad e	fone	CodEsposa	Sexo
PE02	Aninha	23	9999.9999	NULL	F
PE10	Paulinho	20	8888.8888	NULL	M
PE87	Juca	34	7777.7777	PE02	M
PE23	Luana	30	6666.6666	NULL	F
PE54	Beto	28	5555.5555	PE23	M

# FULL JOIN - Exemplo


- Q3: Datas de empréstimos e nomes dos responsáveis. Aqueles empréstimos sem responsáveis, e pessoas não responsabilizadas por empréstimos também devem aparecer no resultado

$\pi$  data, nome (emprestimo  codigo = responsavel pessoa)

---

# FULL JOIN - Exemplo

- Q3: Datas de empréstimos e nomes dos responsáveis. Aqueles empréstimos sem responsáveis, e pessoas não responsabilizadas por empréstimos também devem aparecer no resultado

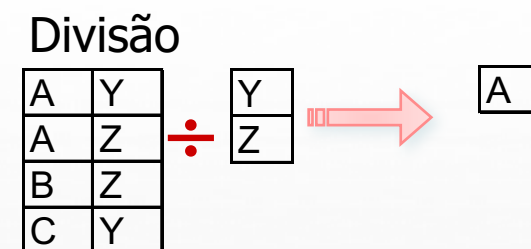
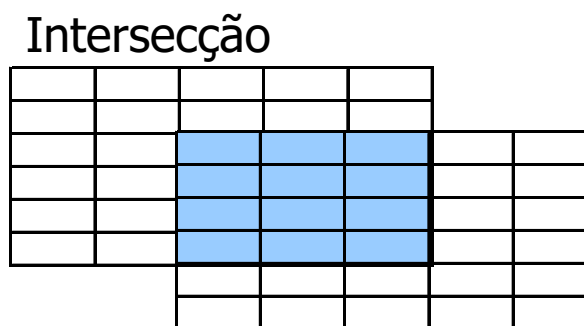
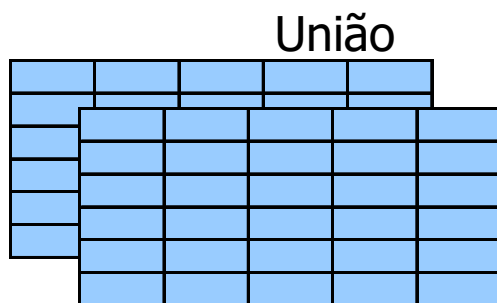
$\pi$  data, nome (emprestimo  codigo = responsavel pessoa)

<u>Data</u>	<u>Nome</u>
10/10/2000	Aninha
10/10/2000	Aninha
01/11/2000	NULL
20/11/2000	NULL
11/11/2000	Paulinho
23/05/2001	Pauinho
NULL	Juca
NULL	Luana
NULL	Beto

empréstimos sem responsáveis

pessoas não responsabilizadas por empréstimos

# União, Intersecção, Diferença e Divisão



*Tabelas resultantes não tem tuplas duplicadas*

# União, Intersecção e Diferença

- Operam somente em **relações compatíveis**
    - Devem ter o **mesmo número de atributos**
    - O **tipo** da n-ésima coluna da primeira relação **deve ser igual** ao tipo da n-ésima coluna da segunda relação.
-

# União, Intersecção e Diferença

- Operam somente em **relações compatíveis**
  - Devem ter o **mesmo número de atributos**
  - O **tipo** da n-ésima coluna da primeira relação **deve ser igual** ao tipo da n-ésima coluna da segunda relação.

## *Livro*

<u>CodLiv</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250

## *Revista*

<u>CodRev</u>	Nome	Ano	NrPaginas
RE005	IEEE Computer	2013	100
RE670	SIGMOD Record	2013	90
RE340	Very Large Databases	2013	85

# União, Intersecção e Diferença

- Operam somente em **relações compatíveis**
    - Devem ter o **mesmo número de atributos**
-

# União, Intersecção e Diferença

- Operam somente em **relações compatíveis**
  - Devem ter o **mesmo número de atributos**
  - O **tipo** da n-ésima coluna da primeira relação **deve ser igual** ao tipo da n-ésima coluna da segunda relação.

*Livro*

<u>CodLiv</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250

int	varchar	year	int
-----	---------	------	-----

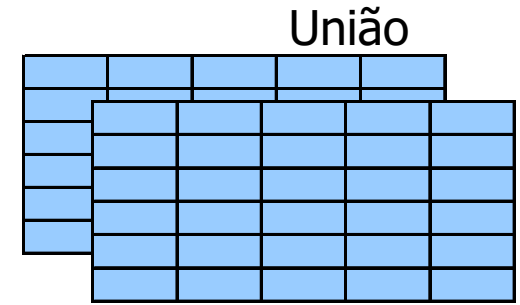
*Revista*

<u>CodRev</u>	Nome	Ano	NrPaginas
RE005	IEEE Computer	2013	100
RE670	SIGMOD Record	2013	90
RE340	Very Large Databases	2013	85

int	varchar	year	int
-----	---------	------	-----



# União - U



- Produz como resultado uma relação que possui todas as tuplas das duas relações – operador **binário**.
- Sintaxe

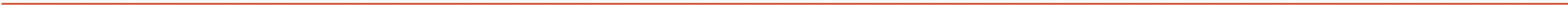
$$\langle \text{relação}_1 \rangle \cup \langle \text{relação}_2 \rangle$$

- O parâmetro  $\langle \text{relação} \rangle$  **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação
-

## União - exemplo

- Q4: Recuperar os dados de todos os livros e revistas

livro  $\cup$  revista



# União - exemplo

- Q4: Recuperar os dados de todos os livros e revistas

livro  $\cup$  revista

*Livro*

<u>CodLiv</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250

*Revista*

<u>CodRev</u>	Nome	Ano	NrPaginas
RE005	IEEE Computer	2013	100
RE670	SIGMOD Record	2013	90
RE340	Very Large Databases	2013	85

$\cup$

=

<u>CodLiv</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250
RE005	IEEE Computer	2013	100
RE670	SIGMOD Record	2013	90
RE340	Very Large Databases	2013	85

# Intersecção - $\cap$

Intersecção


- Produz como resultado uma relação que possui todas as tuplas que são comuns às duas relações – operador **binário**.
- Sintaxe

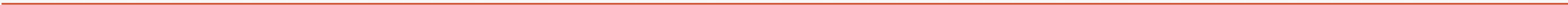
$$\langle \text{relação}_1 \rangle \cap \langle \text{relação}_2 \rangle$$

- O parâmetro  $\langle \text{relação} \rangle$  **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação

# Intersecção - exemplo

- Q4: Recuperar os dados de todos os livros e revistas

livro  $\cap$  revista



# Intersecção - exemplo

- Q4: Recuperar os dados de todos os livros e revistas

livro  $\cap$  revista

*Livro*

<u>CodLiv</u>	Título	Ano	NrPaginas
LI005	Web e Banco de dados	2013	330
LI670	Introdução a Banco de Dados	2000	500
LI340	Programação C	2012	250

*Revista*

<u>CodRev</u>	Nome	Ano	NrPaginas
RE005	IEEE Computer	2013	100
RE670	SIGMOD Record	2013	90
RE340	Very Large Databases	2013	85

$\cap$

=

<u>CodLiv</u>	Título	Ano	NrPaginas
---------------	--------	-----	-----------

**Não há dados em comum nas duas tabelas**

# Intersecção - exemplo

- Q4: Recuperar dados dos funcionários que são clientes

$$\left( \pi_{\text{nome, CPF}} (\text{funcionário}) \right) \cap \left( \pi_{\text{nome, CPF}} (\text{cliente}) \right)$$

*Funcionário*

Nome	CPF
Joana	99999999-9
Pedro	88888888-8
Nuno	77777777-7

$\cap$

*Cliente*

Nome	CPF
Joana	99999999-9
Pedro	66666666-6
Nuno	77777777-7

$=$

Nome	CPF
Joana	99999999-9
Nuno	77777777-7


## Diferença - $-$

- Produz como resultado uma relação que possui todas as tuplas que estão na primeira relação e não estão na segunda relação— operador **binário**.
- Sintaxe

`<relação_1>     $-$     <relação_2>`

- O parâmetro `<relação>` **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação



## Diferença - exemplo

- Q4: Recuperar nomes dos funcionários que não são clientes

$$\left( \pi_{\text{nome, CPF}} (\text{funcionário}) \right) - \left( \pi_{\text{nome, CPF}} (\text{cliente}) \right)$$

*Funcionário*

Nome	CPF
Joana	99999999-9
Pedro	88888888-8
Nuno	77777777-7

—

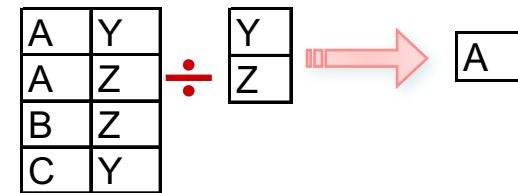
*Cliente*

Nome	CPF
Joana	99999999-9
Pedro	66666666-6
Nuno	77777777-7

=

Nome	CPF
Pedro	88888888-8

Divisão



## Divisão - $\div$

- Retorna as tuplas de um atributo x, em um par  $\langle x, y \rangle$  da primeira relação que está associado com todas as tuplas de um atributo y da segunda relação
- Sintaxe

$$\langle \text{relação}_1 \rangle \div \langle \text{relação}_2 \rangle$$

- O parâmetro  $\langle \text{relação} \rangle$  **pode ser outra expressão algébrica**, pois uma expressão algébrica retorna uma relação

## Divisão - exemplo

- Q4: Recuperar nomes dos funcionários que estão associados a todas as tarefas

$$\left( \pi_{\text{nome}, \text{codTarefa}} (\mathbf{execucao}) \right) \div \left( \pi_{\text{codTarefa}} (\mathbf{tarefa}) \right)$$

---

## Divisão - exemplo

- Q4: Recuperar nomes dos funcionários que estão associados a todas as tarefas

$$\left( \pi_{\text{nome}, \text{codTarefa}} (\text{execucao}) \right) \div \left( \pi_{\text{codTarefa}} (\text{tarefa}) \right)$$

*Execução*

Nome	codTarefa
Joana	1
Joana	2
Joana	3
Pedro	2
Pedro	3
Nuno	1
Nuno	3

÷

*Tarefa*

codTarefa
1
2
3

## Divisão - exemplo

- Q4: Recuperar nomes dos funcionários que estão associados a todas as tarefas

$$\left( \pi_{\text{nome}, \text{codTarefa}} (\text{execucao}) \right) \div \left( \pi_{\text{codTarefa}} (\text{tarefa}) \right)$$

*Execução*

Nome	codTarefa
Joana	1
Joana	2
Joana	3
Pedro	2
Pedro	3
Nuno	1
Nuno	3

÷

*Tarefa*

codTarefa
1
2
3

=

Nome
Joana