



Universidade de Brasília

Departamento de Ciência da Computação



Bancos de Dados

CIC0097



Prof. Pedro Garcia Freitas

<https://pedrogarcia.gitlab.io/>

pedro.garcia@unb.br

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciências da Computação



Este conjunto de slides não deve ser utilizado ou republicado sem a expressa permissão do autor.

This set of slides should not be used or republished without the author's express permission.



Módulo 7
Modelo Relacional
- (MR) -
Parte 1:
Características, formalização e
restrições
CIC0097/2023.1
T1/T2



1. Objetivos

Esta aula apresenta uma introdução sucinta ao principal modelo de dados que é utilizado nos principais SGBDs do tipo relacional. Não é uma introdução completa à abordagem relacional, mas apresenta um conjunto mínimo de conceitos, com o objetivo de permitir que o aluno compreenda o projeto de bancos de dados relacionais



2. Motivação

Além dos SGBD relacionais, existem outros tipos no mercado. Entretanto, hoje, há um claro predomínio dos SGBD relacionais. Além disso, os SGBD relacionais estão gradativamente substituindo os SGBD de outras abordagens (hierárquica, rede, etc).



3. Modelo Relacional: Conceito

- Modelo de dados que **representa a realidade** modela como uma **coleção de relações**.



3. Modelo Relacional: Conceito

- Modelo de dados que **representa a realidade** modela como uma **coleção de relações**.
- Uma relação remete a uma **tabela de valores**, onde cada linha representa uma coleção de **valores relacionados** (colunas)



3. Modelo Relacional: Conceito

- Nesse modelo, uma linha representa uma instância (ocorrência) de uma entidade ou relacionamento do mundo real.



3. Modelo Relacional: Conceito

- Nesse modelo, uma linha representa uma instância (ocorrência) de uma entidade ou relacionamento do mundo real.
- Analogamente, as **tabelas são como as entidades do MER** e as linhas as correspondem às ocorrências de entidades.



3. Modelo Relacional: Conceito

- Um banco de dados relacional é composto de *tabelas* ou *relações*.



3. Modelo Relacional: Conceito

- Um banco de dados relacional é composto de *tabelas* ou *relações*.
- A terminologia “*tabela*” é mais popularmente conhecida comercialmente.



3. Modelo Relacional: Conceito

- Um banco de dados relacional é composto de *tabelas* ou *relações*.
- A terminologia “*tabela*” é mais popularmente conhecida comercialmente.
- Já a terminologia “*relação*” foi utilizada na literatura original (daí que surge a denominação “relacional”) e é mais comum na área acadêmica.



3. Modelo Relacional: Glossário

Nomenclatura formal

Relação

Tupla

Atributo

Domínio

→

→

→

→

Nomenclatura usual
(comercial)

Tabela

Linha

Coluna

Tipo de dado



4. Características

- Exemplo: tabela/relação Funcionário

Funcionário

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

4. Características

- Exemplo: tabela/relação Funcionário

Funcionário

Nome da relação

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

4. Características

- Exemplo: tabela/relação Funcionário

Nome da relação

Atributo (coluna)

Funcionário

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

4. Características

- Exemplo: tabela/relação Funcionário

Nome da relação

Funcionário

Atributo (coluna)

Nome do atributo (Nome do campo)

<u>id funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

Valor de atributo (Campo)

4. Características

- Exemplo: tabela/relação Funcionário

Funcionário

Nome da relação

Atributo
(coluna)

Nome do atributo
(Nome do campo)

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

Tupla (linha)

Valor de atributo (Campo)



4. Características

Tabelas/Relação

- Uma relação (tabela) é um conjunto não ordenado de tuplas (linhas).



4. Características

Tabelas/Relação

- Uma relação (tabela) é um conjunto não ordenado de tuplas (linhas).
- Cada tupla é composta por uma série de valores de atributos (Campos).



4. Características

Tabelas/Relação

- Uma relação (tabela) é um conjunto não ordenado de tuplas (linhas).
- Cada tupla é composta por uma série de valores de atributos (Campos).
- Cada atributo é identificado por um nome de atributo (nome de campo).



4. Características

Tabelas/Relação

- Uma relação (tabela) é um conjunto não ordenado de tuplas (linhas).
- Cada tupla é composta por uma série de valores de atributos (Campos).
- Cada atributo é identificado por um nome de atributo (nome de campo).
- O conjunto de campos das linhas de uma tabela formam uma coluna.



5. Formalização

Tabelas/Relação

- Numa descrição mais formal, chamamos um *esquema de relação* $\mathbf{R}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ uma *representação* composta de uma relação de nome \mathbf{R} e da lista de atributos $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$.

5. Formalização

Tabelas/Relação

- Numa descrição mais formal, chamamos um *esquema de relação* $\mathbf{R}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ uma *representação* composta de uma relação de nome \mathbf{R} e da lista de atributos $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$.
- Cada atributo \mathbf{a}_i é o nome do papel desempenhado por um domínio \mathbf{D} no esquema da relação \mathbf{R} .

5. Formalização

Tabelas/Relação

- Numa descrição mais formal, chamamos um *esquema de relação* $\mathbf{R}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ uma *representação* composta de uma relação de nome \mathbf{R} e da lista de atributos $\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n$.
- Cada atributo \mathbf{a}_i é o nome do papel desempenhado por um domínio \mathbf{D} no esquema da relação \mathbf{R} .
- O grau de uma relação é o número n de atributos.



5. Formalização

- Exemplo: tabela/relação Funcionário

```
Funcionário(id_funcionario, Nome,  
            CategoriaFuncional, Salário)
```



5. Formalização

- Exemplo: tabela/relação Funcionário

```

R
Funcionário(a1id_funcionario, a2Nome,
            a3CategoriaFuncional, a4Salário)

```


5. Formalização

- Exemplo: tabela/relação Funcionário

R
Funcionário(id_funcionario: **longint**, **a**₁
Nome: **varchar**, **a**₂
CategoriaFuncional: **enum**, **a**₃
Salário: **int**) **a**₄

O *esquema textual* pode aparecer **tipado** também em alguns casos.



5. Formalização

Tabelas/Relação

- Um estado de uma relação \mathbf{r} , é definido sobre o esquema $\mathbf{R}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ é um conjunto de ***m-tuplas*** tal que $\mathbf{r} = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_m\}$.



5. Formalização

Tabelas/Relação

- Um estado de uma relação \mathbf{r} , é definido sobre o esquema $\mathbf{R}(\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ é um conjunto de ***m-tuplas*** tal que $\mathbf{r} = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_m\}$.
- Cada tupla da m-tupla é uma lista ordenada de n valores $\mathbf{t} = \langle \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \rangle$.



5. Formalização

Tabelas/Relação

- Um estado de uma relação \mathbf{r} , é definido sobre o esquema $\mathbf{R} (\mathbf{a}_1, \mathbf{a}_2, \dots, \mathbf{a}_n)$ é um conjunto de ***m-tuplas*** tal que $\mathbf{r} = \{\mathbf{t}_1, \mathbf{t}_2, \dots, \mathbf{t}_m\}$.
- Cada tupla da m-tupla é uma lista ordenada de n valores $\mathbf{t} = \langle \mathbf{v}_1, \mathbf{v}_2, \dots, \mathbf{v}_n \rangle$.
- O j^{th} valor da tupla \mathbf{t} , que ***corresponde*** ao atributo \mathbf{a}_j , é diferenciado como $\mathbf{t}[\mathbf{a}_j]$ ou $\mathbf{t} . \mathbf{a}_j$.



5. Formalização

Exemplo

R

Funcionário

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00



5. Formalização

Exemplo

R

Funcionário

a₁

a₂

a₃

a₄

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

5. Formalização

Exemplo

R

Funcionário

a₁**a₂****a₃****a₄**

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

m-tupla com m=2



5. Formalização

Exemplo

R

Funcionário

a_1

a_2

a_3

a_4

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

m-tupla com $m=2$

5. Formalização

Exemplo

R

Funcionário **a₁** **a₂** **a₃** **a₄**

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00 t₁
1518	Liev Bronstein	CL3	2.200,00

v₁ **v₂** **v₃** **v₄**

5. Formalização

Exemplo

R

Funcionário **a₁** **a₂** **a₃** **a₄**

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00

v₁ **v₂** **v₃** **v₄**

t₁ [Nome] = Maria Marta



5. Formalização

Ordenação de tuplas em uma relação

- Uma relação é definida como um conjunto de **tuplas**.
- Elementos em um conjunto **não são ordenados**.
- Portanto, **as tuplas não possuem** uma ordenação.



5. Formalização

Ordenação de tuplas em uma relação

Relação_A

<u>id_funcionario</u>	Nome
1516	João Silva
1517	Maria Marta
1518	Liev Bronstein

=

Relação_B

<u>id_funcionario</u>	Nome
1517	Maria Marta
1518	Liev Bronstein
1516	João Silva



5. Formalização

Ordenação dos valores de uma tupla

- Como uma tupla é uma lista de valores, então a ordem dos valores na tupla é importante.



5. Formalização

Ordenação dos valores de uma tupla

Relação_A

<u>id_funcionario</u>	Nome
1516	João Silva
1517	Maria Marta
1518	Liev Bronstein

≠

Relação_C

Nome	<u>id_funcionario</u>
João Silva	1516
Maria Marta	1517
Liev Bronstein	1518



5. Formalização

Valores e nulidade nas tuplas

- Cada valor numa tupla é **atômico**.



5. Formalização

Valores e nulidade nas tuplas

- Cada valor numa tupla é **atômico**.
- Portanto, atributos compostos e **atributos multivalorados não são permitidos**.



5. Formalização

Valores e nulidade nas tuplas

- Cada valor numa tupla é **atômico**.
- Portanto, atributos compostos e **atributos multivalorados não são permitidos**.
- Valores nulos (**Null**) são usados para representar **valores desconhecidos** ou que não se aplicam a um dado atributo de uma dada tupla.



6. Restrições

- O modelo relacional possui uma série de **restrições**.



6. Restrições

- O modelo relacional possui uma série de **restrições**.
- Todas essas restrições implicam em **regras** que precisam ser seguidas na **manipulação dos dados** via SGBD.



6. Restrições

- O modelo relacional possui uma série de **restrições**.
- Todas essas restrições implicam em **regras** que precisam ser seguidas na **manipulação dos dados** via SGBD.
- Essas regras de manipulações são garantidas pelo SGBD.



6. Restrições

Restrições de domínio:

- As restrições de domínio devem especificar que valor de cada atributo **a** deve ser **atômico** dentro de um domínio **dom (A)** em todas as tuplas da relação.
- Noutras palavras: cada atributo deve **possuir um tipo**.



6. Restrições


Restrições de chave:

- Por definição, **todos os elementos** de um conjunto são **distintos**.
- Portanto, **todas as tuplas** em uma relação devem ser **distintas**.
- Ou seja, duas tuplas distintas **não podem ter a mesma combinação de valores** em todos os seus atributos!

$$t_1 \neq t_2$$

6. Restrições

Restrições de chave:

$$t_1 \neq t_2$$

$$t_1[SK] \neq t_2[SK]$$

The diagram shows the relationship between a general tuple inequality and a specific key inequality. A grey arrow points from $t_1 \neq t_2$ to $t_1[SK] \neq t_2[SK]$. Red circles highlight the $[SK]$ parts in the second equation, and a red arrow points from the first $[SK]$ to the explanatory text below.

SK: Conjunto de atributos onde os **valores** são diferentes

6. Restrições

Restrições de chave:

- Surge então o conceito de **superchave**.
- Superchave é a combinação dos atributos/colunas **SK** que identificam unicamente uma tupla/linha na relação/tabela.
- Toda relação possui **pelo menos uma** superchave (todos os seus atributos).



6. Restrições

Exemplo: Superchave

Funcionário

<u>id funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00
1519	João Silva	CL1	1.500,00

6. Restrições

Exemplo: Superchave

Funcionário

Não formam uma superchave

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00
1519	João Silva	CL1	1.500,00



6. Restrições

Exemplo: Superchave

Funcionário

Formam uma superchave

<u>id funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00
1519	João Silva	CL1	1.500,00

6. Restrições

Exemplo: Superchave

Formam uma superchave

Funcionário

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00
1519	João Silva	CL1	1.500,00

Apenas 1 atributo diferencia as tuplas de facto (Chave). 57

6. Restrições

Exemplo: Superchave

Funcionário

Formam uma superchave

<u>id_funcionario</u>	Nome	CategoriaFuncional	Salário
1516	João Silva	CL1	1.500,00
1517	Maria Marta	CL1	1.500,00
1518	Liev Bronstein	CL3	2.200,00
1519	João Silva	CL1	1.500,00

Atributos redundantes

Apenas 1 atributo diferencia as tuplas de facto (**Chave**).



6. Restrições

Restrições de chave:

- Uma chave de uma relação é uma **superchave sem atributos redundantes**.
- Ou seja, é o conjunto mínimo de atributos que **garantem a restrição de unicidade** das tuplas.



6. Restrições

Restrições de chave:

- A partir do **conceito de chave**, podemos derivar 3 outros conceitos:
 - Chave candidata;
 - Chave primária;
 - Chave única;



6. Restrições

Restrições de chave:

- Como um esquema pode ter mais de uma chave, cada uma delas é uma **chave candidata**.



6. Restrições

Restrições de chave:

- Como um esquema pode ter mais de uma chave, cada uma delas é uma **chave candidata**.
- As chaves candidatas são identificadas para a escolha da **chave primária** da relação.



6. Restrições

Restrições de chave:

- Como um esquema pode ter mais de uma chave, cada uma delas é uma **chave candidata**.
- As chaves candidatas são identificadas para a escolha da **chave primária** da relação.
- Como diversas chaves podem ser construídas a partir do **conjunto mínimo de atributos** que garantem a restrição de unicidade, cada **combinação** desse conjunto é uma **chave única**.

6. Restrições

Restrições de integridade:

- **Restrição de integridade de entidade:** o valor da chave primária deve ser único (da definição) e **não pode ser nulo.**
- **Restrição de integridade referencial:** é especificada entre duas relações e usadas para manter a **consistência entre as tuplas.**
 - “Se uma tupla de uma tabela X referencia um campo da tabela Y, essa referencia deve apontar para uma **linha existente** de Y.”



6. Restrições

Restrições de integridade:

- Surge, portanto, o conceito de **chave estrangeira**.
- A **chave estrangeira** é um atributo (ou combinação de atributos) que se refere à chave primária de outra relação.



6. Restrições

Restrições de integridade:

- Uma **chave estrangeira** é uma coluna ou uma combinação de colunas, cujos **valores aparecem necessariamente na chave primária** de uma tabela.
- A chave estrangeira é o **mecanismo que permite a implementação de relacionamentos** em um banco de dados relacional.

6. Restrições

Restrições de integridade:

- Chave estrangeira: Um conjunto de atributos \mathbf{FK} no esquema \mathbf{R}_1 é um chave estrangeira que referencia \mathbf{R}_2 se:
 - Os atributos em \mathbf{FK} tem o mesmo domínio dos atributos da chave primária \mathbf{PK} de \mathbf{R}_2 ;
 - O valor de \mathbf{FK} numa tupla \mathbf{t}_1 de um estado $\mathbf{r}_1 \in \mathbf{R}_1$ ocorre como valor de \mathbf{PK} para alguma tupla \mathbf{t}_2 de algum estado $\mathbf{r}_2 \in \mathbf{R}_2$.



6. Restrições

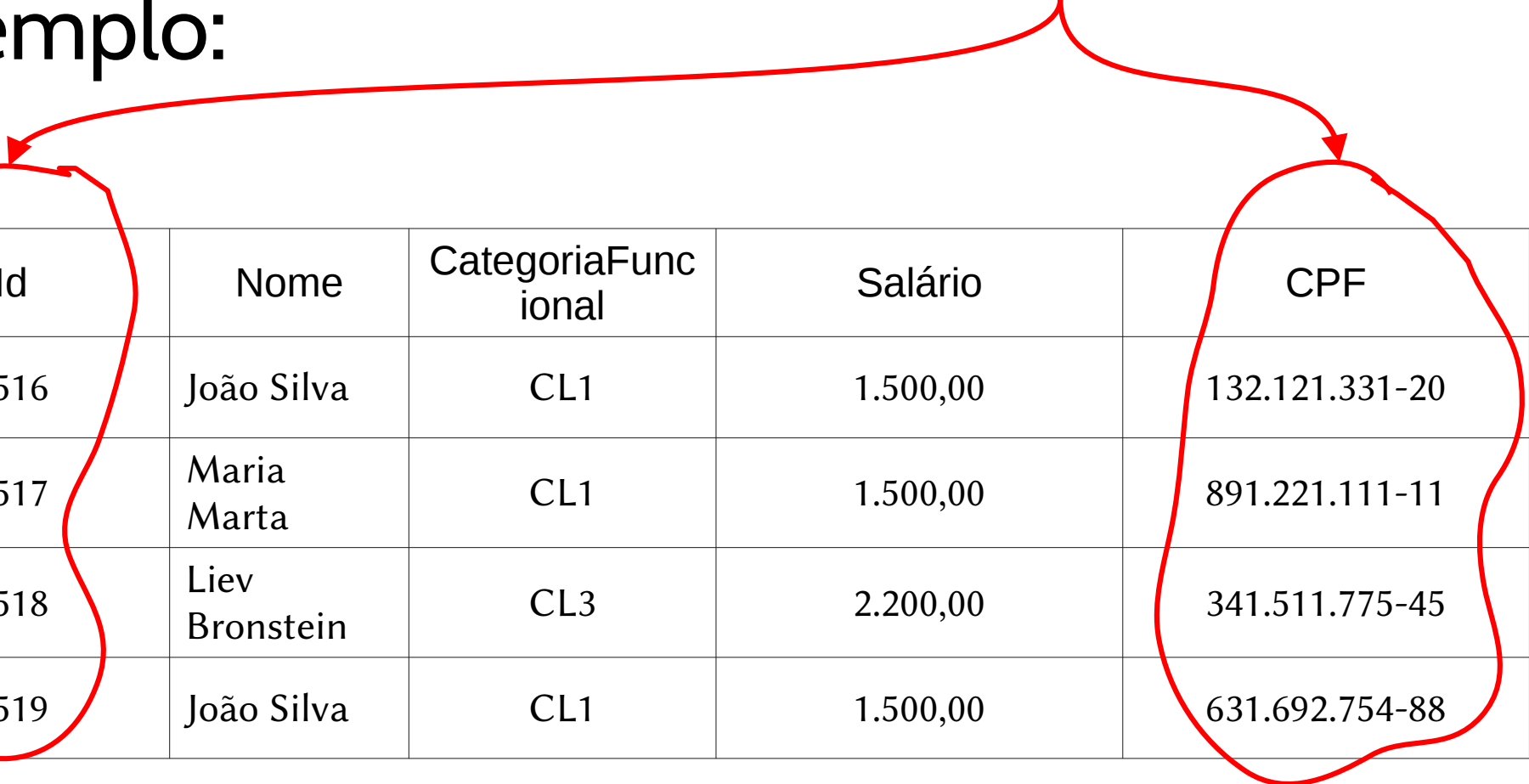
Restrições de integridade:

$$t_1 [FK] \neq t_2 [PK]$$

6. Restrições

Esses dois atributos permitem identificar as linhas unicamente.

Exemplo:

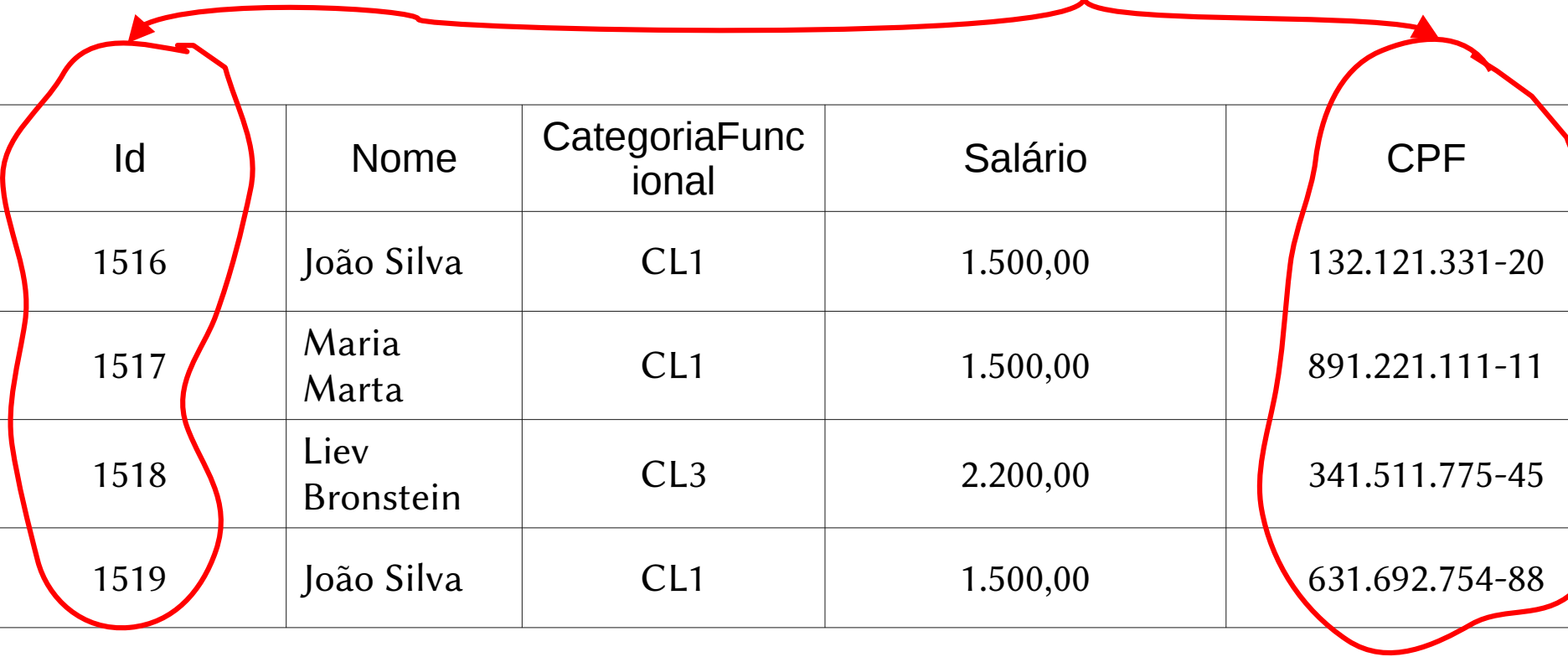


Id	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88

6. Restrições

Exemplo:

Portanto, {Id e CPF} formam o conjunto mínimo de atributos que garantem a restrição de unicidade das tuplas.



Id	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88

6. Restrições

Exemplo:

Assim, a combinação desses atributos formam as **chaves candidatas** {**Id**, **CPF**, **Id** \cup **CPF**}, cada uma delas sendo uma **chave única**.

Id	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88

6. Restrições

Escolhendo Id como PK, usualmente destacamos dos outros atributos.

Exemplo:

<u>Id</u>	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88



6. Restrições

Exemplo:

Funcionário

CategoriaFuncional

<u>Codigo</u>	Descrição
CL1	Júnior
CL2	Gerente

<u>Id</u>	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88

6. Restrições

Exemplo:

Funcionário

<u>Id_fun</u>	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88

CategoriaFuncional

PK₂

<u>Codigo</u>	Descrição
CL1	Júnior
CL2	Gerente

6. Restrições

Exemplo:

Funcionário

<u>Id_fun</u>	Nome	CategoriaFuncional	Salário	CPF
1516	João Silva	CL1	1.500,00	132.121.331-20
1517	Maria Marta	CL1	1.500,00	891.221.111-11
1518	Liev Bronstein	CL3	2.200,00	341.511.775-45
1519	João Silva	CL1	1.500,00	631.692.754-88

CategoriaFuncional

<u>Codigo</u>	Descrição
CL1	Júnior
CL2	Gerente

PK

FK

[illegible]



Dúvidas?

