

Prática - SQL

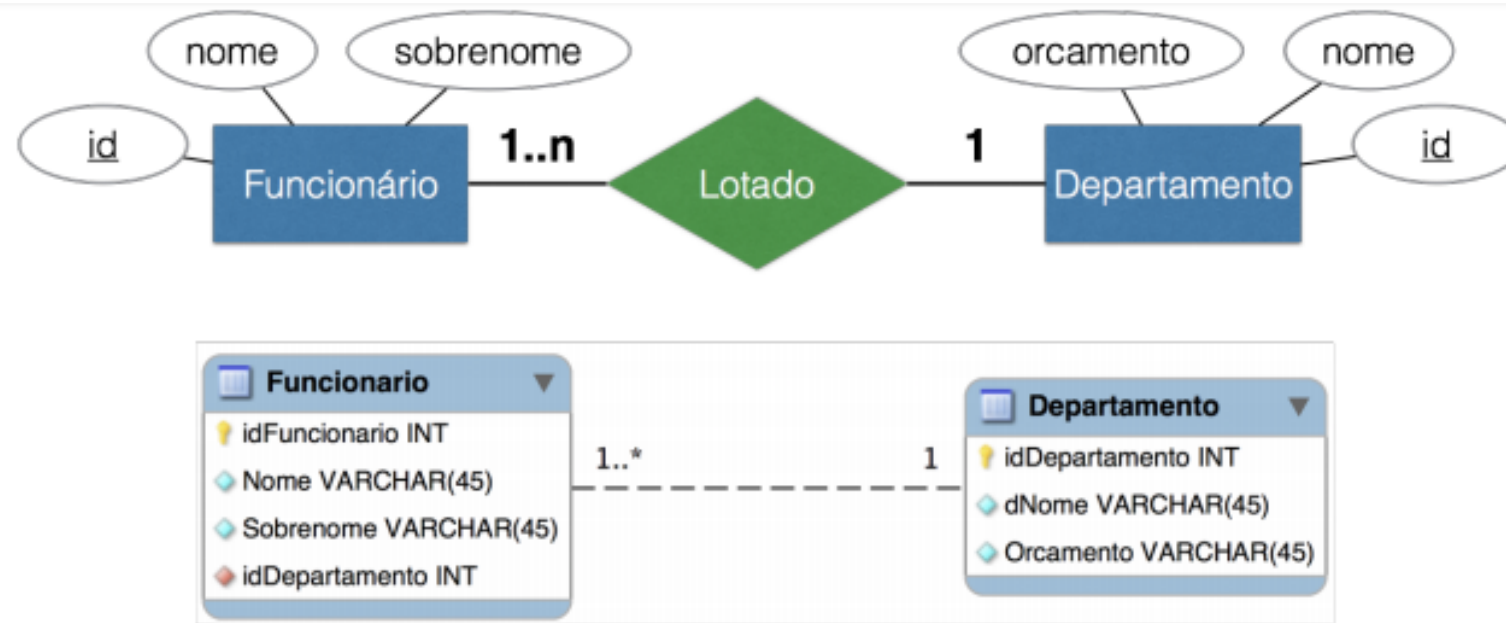


Figura 1: Esquema de banco de dados com as relações Funcionário e Departamento

```
1 CREATE TABLE Funcionario (  
2     idFuncionario INT NOT NULL,  
3     Nome VARCHAR(45) NOT NULL,  
4     Sobrenome VARCHAR(45) NOT NULL,  
5     idDepartamento INT NOT NULL,  
6     PRIMARY KEY (idFuncionario),  
7     CONSTRAINT fk_Funcionario_Departamento FOREIGN  
8         KEY (idDepartamento)  
9     REFERENCES Departamento (idDepartamento));
```

```
9 CREATE TABLE Departamento (  
10     idDepartamento int(11) NOT NULL,  
11     dNome varchar(255) NOT NULL,  
12     Orcamento decimal(10,0) NOT NULL,  
13     PRIMARY KEY (idDepartamento));
```

DML – Selecionar Dados (SELECT)

Cláusula WHERE, AND e OR

- Comando utilizado para selecionar tuplas de uma ou mais tabelas.

```
SELECT coluna1, coluna2, coluna3  
FROM tabela_nome1, tabela_nome2  
WHERE coluna1=valor1  
AND   coluna2=valor2  
OR    coluna2=valor3;
```

DML – Selecionar Dados (SELECT)

Operadores Lógicos

- Realiza operações do tipo booleano (verdadeiro/falso).

Operador	Significado	Apresenta resultado se:
AND	E	2 condições verdadeira
OR	OU	Apenas 1 das condições verdadeira
NOT ou !	Negação	Inverte o resultado da expressão lógica

```
SELECT * FROM departamento WHERE dNome='TI' OR dNome='Financeiro';
```

```
SELECT * FROM departamento WHERE NOT dNome='Juridico';
```

DML – Selecionar Dados (SELECT)

Operadores Relacionais

- Utilizados para realizar comparações entre valores.

Operador	Significado	Exemplo
=	Igual	Codigo_Autor = 2
<	Menor que	Preco_Venda < 10
<=	Menor ou igual a	Preco_Venda <= 10
>	Maior que	Preco_Venda > 10
>=	Maior ou igual a	Preco_Venda >= 10
!= ou <>	Diferente	Codigo_Autor != 2

```
SELECT * FROM departamento WHERE Orcamento <=50000;
```

idFuncionario	Nome	Sobrenome	idDepartamento
123	Julio	Silva	1
152	Arnaldo	Coelho	1
222	Carol	Ferreira	2
326	João	Silveira	2
331	George	de la Rocha	3
332	José	Oliveira	1
546	José	Pereira	4
631	David	Luz	3
654	Zacarias	Ferreira	4
745	Eric	Estrada	4
845	Elizabeth	Coelho	1
846	Joaquim	Goveia	1

SELECT f.Nome,d.dNome FROM
departamento d, funcionario f where
f.idDepartamento=d.idDepartamento

+ Opções

Nome	dNome
Julio	Financeiro
Arnaldo	Financeiro
Carol	TI
João	TI
George	Gestão de Pessoas
José	Financeiro
José	Pesquisa e Desenvolvimento
David	Gestão de Pessoas
Zacarias	Pesquisa e Desenvolvimento
Eric	Pesquisa e Desenvolvimento
Elizabeth	Financeiro
Joaquim	Financeiro

SELECT f.Nome,d.dNome FROM departamento d, funcionario f
where d.idDepartamento=1 AND
d.idDepartamento=f.idDepartamento

Nome	dNome
Julio	Financeiro
Arnaldo	Financeiro
José	Financeiro
Elizabeth	Financeiro
Joaquim	Financeiro

DML – Selecionar Dados (SELECT)

Operadores Especiais

- Os principais operadores especiais são:
 - IS NULL ou IS NOT NULL;
 - BETWEEN;
 - LIKE;
 - IN.

```
SELECT * FROM `funcionario` WHERE Sobrenome LIKE '%a';
```

```
SELECT * FROM `funcionario` WHERE Sobrenome LIKE 's%';
```

DML – Selecionar Dados (SELECT)

Operadores Especiais

- IN compara o valor de uma coluna com um conjunto informado.
- Exemplo:

```
SELECT * from funcionario  
WHERE codigo IN (2, 4, 25, 30)
```

```
SELECT * FROM funcionario WHERE Nome IN ('Carol', 'Joaquim', 'Eric');
```

DML – Selecionar Dados (SELECT)

Funções de Agregação

- ❑ Na SQL existem algumas funções que agrupam valores.
- ❑ São elas:
 - ❑ COUNT: conta a quantidade de linhas
 - ❑ AVG: realiza a média aritmética da coluna
 - ❑ SUM: soma os valores da coluna
 - ❑ MIN: retorna o menor valor da coluna
 - ❑ MAX: retorna o maior valor da coluna

DML – Selecionar Dados (SELECT)

Funções de Agregação

ID	Nome	Valor	Quantidade
1	Lápis	1,00	10
2	Borracha	3,00	10
3	Caderno	15,00	10

SELECT COUNT(idFuncionario)FROM funcionario;

SELECT MIN(Orcamento)FROM departamento;

SELECT AVG(Orcamento)FROM departamento;

SELECT SUM(Orcamento * 0.25) AS acrescimo
FROM departamento;

SELECT COUNT(*) AS QTD FROM ITEM;

QTD
3

SELECT MIN(ID) as menor FROM ITEM;

menor
1

SELECT MAX(ID) as maior FROM ITEM;

maior
3

SELECT AVG(valor) as media FROM ITEM;

media
6.33

SELECT SUM(valor * quantidade) as valorTotal FROM ITEM;

valorTotal
190.0

DML – Selecionar Dados (SELECT)

Cláusulas GROUP BY, HAVING e ORDER BY

```
SELECT coluna1, coluna2, coluna3  
FROM tabela_nome1, tabela_nome2  
WHERE coluna1=valor1  
AND   coluna2=valor2  
OR   coluna2=valor3  
GROUP BY coluna1  
HAVING AVG(coluna1) > 100  
ORDER BY coluna2;
```

DML – Selecionar Dados (SELECT)

Ordenando valores

- Para ordenar o resultado de uma pesquisa utilizamos a cláusula `ORDER BY`.
- Pode ser ordenado de forma ascendente (`ASC`) ou descendente (`DESC`).
- O padrão é `ASC`.

```
SELECT * FROM departamento ORDER BY Orcamento
```

```
SELECT * FROM departamento ORDER BY Orcamento DESC
```