



Formação Desenvolvedor Moderno Módulo: Banco de Dados

Capítulo: Consultas SQL

<https://devsuperior.com.br>

1

Agenda

- Álgebra relacional
 - Projeção
 - Restrição
 - Produto cartesiano
 - Junção
- SELECT, FROM, WHERE
- Junções
- Operadores LIKE, IN, BETWEEN
- Funções "comuns"
- Funções de agregação: COUNT, SUM, MIN, MAX, AVG
- Cláusulas DISTINCT, ORDER BY, TOP, LIMIT, GROUP BY
- Subconsulta
- União e diferença

2

Motivação

- Como extrair informações de uma base de dados?
- Como responder questões de negócio?

É preciso saber CONSULTAR os dados de forma a obter as informações desejadas para responder as questões de negócio.

- Quais são os produtos da categoria dos eletrônicos?
- Qual foi o melhor vendedor do mês?
- Quanto foram os custos da empresa em cada categoria?
- Quais alunos já terminaram todas tarefas do curso?

3

Consultas SQL

A SQL possui um subconjunto dedicado a realizar consultas a uma base de dados. São os comandos "SELECT ...".

Alguns autores classificam esse subconjunto como DQL (Data Query Language), e outros autores o colocam dentro da DML (Data Manipulation Language).

<https://pt.stackoverflow.com/questions/262867/o-que-s%C3%A3o-as-siglas-ddl-dml-dql-dtl-e-dcl>

4

Referências

<https://www.w3schools.com/sql/default.asp>

<https://github.com/devsuperior/scripts-sql-cap7>

<https://www.beecrowd.com.br>

Álgebra relacional

As consultas a um banco de dados relacional estão fundamentadas em uma base teórica muito consolidada e elegante chamada Álgebra Relacional.

Da mesma forma que temos uma sólida álgebra na aritmética e nos conjuntos, também a temos no modelo relacional (tabelas).

- Operações elementares (adição, subtração, multiplicação, etc.)
- Operações de conjuntos (interseção, união, diferença)

Aritmética

$$5 + 4$$

Expressão composta por
NÚMEROS e **OPERADORES ARITMÉTICOS**

RESULTADO = **9**

O resultado é um **NÚMERO**

Álgebra relacional

tb_empregado					tb_departamento	
id	nome	dept_id			id	nome
1	Maria	1			1	Financeiro
2	João	2			2	Recursos Humanos
3	Ana	1			3	Comercial
4	Carlos	2				

Expressão composta por
RELAÇÕES (TABELAS) e **OPERADORES RELACIONAIS**

RESULTADO =

id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	2	Recursos Humanos
3	Ana	1	1	Financeiro
4	Carlos	2	2	Recursos Humanos

O resultado é uma **RELAÇÃO (TABELA)**

7

PROJEÇÃO

É uma operação que "filtra" as colunas da tabela.
Exemplo:

"Projeção das colunas id e nome da tabela tb_empregado"

Notação da álgebra relacional:

$\pi_{(id, nome)} tb_empregado$

Notação SQL:

```
SELECT id, nome  
FROM tb_empregado
```

tb_empregado

id	nome	dept_id
1	Maria	1
2	João	2
3	Ana	1
4	Carlos	2



id	nome
1	Maria
2	João
3	Ana
4	Carlos

8

Testando na prática

<https://github.com/devsuperior/scripts-sql-cap7/blob/main/empregados.sql>

tb_empregado			tb_departamento	
id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	2	Recursos Humanos
3	Ana	1	3	Comercial
4	Carlos	2		

9

RESTRIÇÃO (SELEÇÃO)

É uma operação que "filtra" as linhas da tabela conforme um predicado. Exemplo:

"Restrição na tb_empregado das linhas tal que: dept_id = 2"

Notação da álgebra relacional:

$\sigma_{(\text{dept_id} = 2)} \text{tb_empregado}$

Notação SQL:

```
SELECT *  
FROM tb_empregado  
WHERE dept_id = 2
```

tb_empregado		
id	nome	dept_id
1	Maria	1
2	João	2
3	Ana	1
4	Carlos	2



id	nome	dept_id
2	João	2
4	Carlos	2

10

Você pode combinar operações

$\pi(id, nome) (\sigma_{dept_id = 2} tb_empregado)$

```
SELECT id, nome  
FROM tb_empregado  
WHERE dept_id = 2
```

tb_empregado

id	nome	dept_id
1	Maria	1
2	João	2
3	Ana	1
4	Carlos	2



id	nome	dept_id
2	João	2
4	Carlos	2



RESULTADO FINAL =

id	nome
2	João
4	Carlos

11

PRODUTO CARTESIANO

É uma operação que faz o "cruzamento" de todos os registros entre duas tabelas. Exemplo:

"Produto cartesiano entre as tabelas tb_empregado e tb_departamento"

Notação da álgebra relacional:

$tb_empregado \times tb_departamento$

Notação SQL:

```
SELECT *  
FROM tb_empregado, tb_departamento
```

tb_empregado

id	nome	dept_id
1	Maria	1
2	João	2
3	Ana	1
4	Carlos	2

tb_departamento

id	nome
1	Financeiro
2	Recursos Humanos
3	Comercial



id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	1	Financeiro
3	Ana	1	1	Financeiro
4	Carlos	2	1	Financeiro
1	Maria	1	2	Recursos Humanos
2	João	2	2	Recursos Humanos
3	Ana	1	2	Recursos Humanos
4	Carlos	2	2	Recursos Humanos
1	Maria	1	3	Comercial
2	João	2	3	Comercial
3	Ana	1	3	Comercial
4	Carlos	2	3	Comercial

12

JUNÇÃO

Faz o "cruzamento" entre duas tabelas apenas dos registros correspondentes. Exemplo:

id	nome	dept_id
1	Maria	1
2	João	2
3	Ana	1
4	Carlos	2

id	nome
1	Financeiro
2	Recursos Humanos
3	Comercial



"Junção entre as tabelas tb_empregado e tb_departamento"

Notação da álgebra relacional:

`tb_empregado ⋈ tb_departamento`

Notação SQL:

```
SELECT *
FROM tb_empregado
INNER JOIN tb_departamento ON tb_empregado.dept_id = tb_departamento.id
```

id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	2	Recursos Humanos
3	Ana	1	1	Financeiro
4	Carlos	2	2	Recursos Humanos

13

Alternativa: produto cartesiado + restrição

$\sigma_{(tb_empregado.dept_id = tb_departamento.id)}(tb_empregado \times tb_departamento)$

```
SELECT *
FROM tb_empregado, tb_departamento
WHERE tb_empregado.dept_id = tb_departamento.id
```

id	nome	dept_id
1	Maria	1
2	João	2
3	Ana	1
4	Carlos	2

id	nome
1	Financeiro
2	Recursos Humanos
3	Comercial

id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	1	Financeiro
3	Ana	1	1	Financeiro
4	Carlos	2	1	Financeiro
1	Maria	1	2	Recursos Humanos
2	João	2	2	Recursos Humanos
3	Ana	1	2	Recursos Humanos
4	Carlos	2	2	Recursos Humanos
1	Maria	1	3	Comercial
2	João	2	3	Comercial
3	Ana	1	3	Comercial
4	Carlos	2	3	Comercial

id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	2	Recursos Humanos
3	Ana	1	1	Financeiro
4	Carlos	2	2	Recursos Humanos

14

Resumo

Principais formas de obter a junção entre duas tabelas usando SQL:

```
SELECT *  
FROM tb_empregado  
INNER JOIN tb_departamento ON tb_empregado.dept_id = tb_departamento.id
```

```
SELECT *  
FROM tb_empregado, tb_departamento  
WHERE tb_empregado.dept_id = tb_departamento.id
```

15

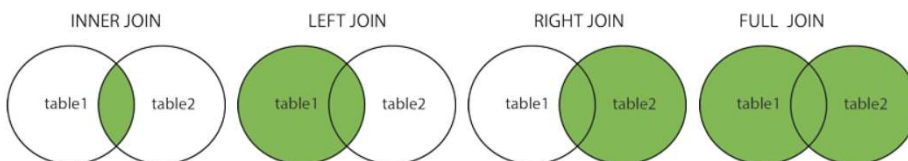
Tipos de junção

https://www.w3schools.com/sql/sql_join.asp

Different Types of SQL JOINS

Here are the different types of the JOINS in SQL:

- **(INNER) JOIN** : Returns records that have matching values in both tables
- **LEFT (OUTER) JOIN** : Returns all records from the left table, and the matched records from the right table
- **RIGHT (OUTER) JOIN** : Returns all records from the right table, and the matched records from the left table
- **FULL (OUTER) JOIN** : Returns all records when there is a match in either left or right table



16

Testando na prática

<https://github.com/devsuperior/scripts-sql-cap7/blob/main/joins.sql>

tb_empregado			tb_departamento	
id	nome	dept_id	id	nome
1	Maria	1	1	Financeiro
2	João	2	2	Recursos Humanos
3	Ana	1	3	Comercial
4	Carlos	null		

17

Renomeação: AS

Usos comuns:

- Remover ambiguidades
- Dar nome a campos calculados
- Dar nome a tabelas resultantes de subconsultas

18

Operadores LIKE, IN, BETWEEN

https://www.w3schools.com/sql/sql_like.asp

https://www.w3schools.com/sql/sql_in.asp

https://www.w3schools.com/sql/sql_ref_between.asp

Testando na prática:

<https://github.com/devsuperior/scripts-sql-cap7/blob/main/sales.sql>

Nota: o operador IN pode ser usado tanto para valores como para tabelas.

19

Funções comuns

UPPER, LOWER

CAST, ROUND

DAY, MONTH, YEAR, EXTRACT

CONCAT

CASE

REPLACE

CHAR_LENGTH

MD5

Atenção: as funções variam um pouco entre os SGBDs

20

Funções de agregação

COUNT, SUM, AVG, MIN, MAX

https://www.w3schools.com/sql/sql_count_avg_sum.asp

https://www.w3schools.com/sql/sql_min_max.asp

21

DISTINCT

https://www.w3schools.com/sql/sql_distinct.asp

Exemplo: id e nome dos vendedores que venderam produtos mais caros que 500.00

22

ORDER BY, TOP, LIMIT

https://www.w3schools.com/sql/sql_orderby.asp

https://www.w3schools.com/sql/sql_top.asp

23

GROUP BY

https://www.w3schools.com/sql/sql_groupby.asp

Exemplo: quantidade de vendas por dia

Exemplo: total em dinheiro vendido por vendedor

24

Subconsultas

Usos comuns:

- Diferença / União
- Resolução de consultas complexas

O resultado de uma consulta é uma tabela. Este resultado pode naturalmente ser usado como parâmetro de uma cláusula FROM ou qualquer outra cláusula que receba uma tabela como argumento.

Exemplo: data da venda e nome do vendedor para vendas cujo preço unitário seja menor que 500

25

UNIÃO

https://www.w3schools.com/sql/sql_union.asp

Exemplo: id e data das vendas cujo preço unitário é maior que 800.0, ou que sejam do vendedor "Joaquim Silva Borges"

26

DIFERENÇA

https://www.w3schools.com/sql/sql_in.asp

Exemplo: id, data, e quantidade de todas vendas que não sejam nos mesmos dias em que o vendedor "Ivan Reis" vendeu.

Base de dados:

https://github.com/devsuperior/scripts-sql-cap7/blob/main/sales_diff.sql