



por: Marília S. C. Santos



com ChatGPT



EDUK
4ALL

APRENDA **COMANDOS BÁSICOS SQL** **(CONCEITO) aplicados na PRÁTICA**

<https://www.linkedin.com/in/maliarte/>



Imagine que SQL (Structured Query Language) é como uma linguagem mágica que ajuda você a falar com um grande baú de informações (o banco de dados). Esse baú guarda listas de coisas, como uma lista de alunos, de cursos, de notas, etc.



Agora, pense que esse banco de dados é como um grande caderno de escola cheio de tabelas. Cada tabela tem várias linhas (dados) e colunas (informações diferentes, como nome, idade, curso, etc.).


Com SQL, você pode fazer perguntas para esse banco de dados. Por exemplo:

- "Quem está matriculado no curso de Matemática?"
- "Qual aluno tirou a maior nota?"
- "Quantos alunos existem na escola?"

Agora, vamos aprender os comandos que usamos para falar com esse "caderno de informações"! 📖✨

SELECT

Pega informações da tabela

 Imagine que você tem uma lista com o nome de todos os alunos e quer ver só alguns nomes. O SELECT é como pegar uma lupa e olhar apenas as partes importantes da tabela.



Exemplo:

"Quero ver o nome e o email de todos os alunos!"

<https://www.linkedin.com/in/maliarte/>



SELECT

sql


```
SELECT nome, email FROM aluno;
```



FROM

<https://www.linkedin.com/in/maliarte/>

Diz de qual tabela os dados vêm

 O FROM fala para o SQL:
"Olhe nessa tabela específica!"



Exemplo:

"Busque os nomes dos alunos da
tabela chamada 'aluno'!"

sql

```
SELECT nome FROM aluno;
```



WHERE

<https://www.linkedin.com/in/maliarte/>

📌 O WHERE é como perguntar algo específico.

"Mostre só os alunos que têm e-mail do Gmail!"

👉 Exemplo:

sql

📄 Copiar

```
SELECT nome, email FROM aluno WHERE email LIKE '%@gmail.com';
```



Explicação: Esse comando busca alunos que tenham "@gmail.com" no e-mail.

JOIN

<https://www.linkedin.com/in/maliarte/>

📌 O JOIN é como juntar duas listas.

Imagine que você tem uma lista de alunos e outra de cursos. Se quiser ver em qual curso cada aluno está matriculado, precisa juntar essas tabelas.

👉 Exemplo:

sql


```
SELECT aluno.nome, curso.titulo  
FROM aluno  
JOIN matricula ON aluno.aluno_id = matricula.aluno_id  
JOIN curso ON matricula.curso_id = curso.curso_id;
```

🔍 Explicação: Estamos conectando três tabelas para mostrar quais alunos estão em quais cursos!



GROUP BY

<https://www.linkedin.com/in/maliarte/>

 O GROUP BY é como organizar uma lista por categoria. Se queremos ver a média de notas por curso, primeiro precisamos agrupar os alunos que pertencem ao mesmo curso.

 Exemplo:



GROUP BY

<https://www.linkedin.com/in/maliarte/>

sql

```
SELECT curso.titulo, AVG(matricula.nota) AS media_notas
FROM curso
JOIN matricula ON curso.curso_id = matricula.curso_id
WHERE matricula.nota IS NOT NULL
GROUP BY curso.titulo;
```

Aqui, estamos pegando todas as notas, agrupando pelo nome do curso e calculando a média das notas

HAVING → Filtro depois do GROUP BY

📌 O HAVING é como um WHERE, mas ele só funciona depois que os dados foram agrupados.

👉 Exemplo:

"Quero ver apenas cursos onde a média de notas seja maior ou igual a 8!"

sql

```
SELECT curso.titulo, AVG(matricula.nota) AS media_notas
FROM curso
JOIN matricula ON curso.curso_id = matricula.curso_id
WHERE matricula.nota IS NOT NULL
GROUP BY curso.titulo
HAVING AVG(matricula.nota) >= 8.0;
```



ORDER BY → ORGANIZA OS DADOS

O **ORDER BY** coloca os resultados em ordem crescente (A-Z, 0-9) ou decrescente (Z-A, 9-0).

👉 Exemplo:

"Quero ver os alunos ordenados por nome, do A ao Z!"

sql

```
SELECT nome FROM aluno ORDER BY nome ASC;
```

📌 Se quisermos ver do Z ao A, usamos **DESC** :

sql

```
SELECT nome FROM aluno ORDER BY nome DESC;
```



COUNT → Conta quantos resultados existem

O **COUNT** conta quantas coisas aparecem no banco de dados.

👉 Exemplo:

"Quantos alunos estão matriculados em cada curso?"

sql

📋 Copiar

✎ Editar

```
SELECT curso.titulo, COUNT(matricula.aluno_id) AS total_matriculados
FROM curso
JOIN matricula ON curso.curso_id = matricula.curso_id
GROUP BY curso.titulo;
```



AVG, MAX, MIN

<https://www.linkedin.com/in/maliarte/>

Calcula médias e encontra valores extremos
📌 Esses comandos fazem cálculos nos dados:

👉 Exemplo:

"Quero saber a maior nota que existe no banco de dados!"

sql

```
SELECT MAX(nota) AS maior_nota FROM matricula;
```

- **AVG** → Calcula a média
- **MAX** → Encontra o maior valor
- **MIN** → Encontra o menor valor

👉 Outro exemplo:

"Quero saber a menor nota de um aluno!"

sql

```
SELECT MIN(nota) AS menor_nota FROM matricula;
```


INSERT INTO

<https://www.linkedin.com/in/maliarte/>



Adiciona novos dados

O INSERT coloca novas informações dentro da tabela.



Exemplo:

sql

```
INSERT INTO aluno (aluno_id, nome, email)
VALUES (6, 'Pedro Santos', 'pedro@gmail.com');
```



update

<https://www.linkedin.com/in/maliarte/>

📌 O UPDATE modifica uma informação dentro da tabela.

👉 Exemplo:

👉 Exemplo:

"Quero mudar o e-mail do aluno Pedro para *'pedro@hotmail.com'!*"

sql

📄 Copiar

```
UPDATE aluno
SET email = 'pedro@hotmail.com'
WHERE nome = 'Pedro Santos';
```



delete

📌 O DELETE apaga informações do banco de dados.

👉 Exemplo:

👉 Exemplo:

"Quero apagar o aluno chamado Pedro!"

sql

```
DELETE FROM aluno WHERE nome = 'Pedro Santos';
```

⚠️ Cuidado! Se esquecer o WHERE, você pode apagar todos os alunos por acidente!

<https://www.linkedin.com/in/maliarte/>





por: Marília S. C. Santos



com ChatGPT

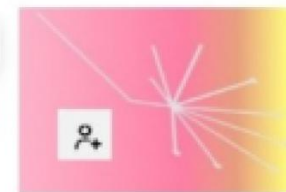


EDUK
4ALL

APRENDA



@maliarte21



É real, o contraditório e o fantástico! Tudo sobre
uma Sociedade descentralizada.

SIGA PARA +

<https://www.linkedin.com/in/maliarte/>

Agora você entende SQL de um jeito fácil e divertido!
Qual comando você quer testar primeiro? 🚀

Obrigado