



SQL Básico: Informação Descomplicada

Miriam Ribeiro

COMANDOS BÁSICOS SQL

Manipulando dados

Este ebook é uma introdução prática aos comandos SQL: SELECT, INSERT, DELETE e UPDATE. Cada seção explica a função de cada comando, com exemplos básicos.



SQL



SELECT

Recuperando dados de uma ou mais colunas de
uma tabela

SELECIONANDO COLUNAS

```
SELECT nome, email  
FROM clientes;
```

Selecionar as colunas "nome" e "email" da tabela "clientes"

```
SELECT nome AS Nome_Cliente, email AS Email_Cliente  
FROM clientes;
```

Utilizando aliases para renomear as colunas no resultado da consulta



FILTRANDO DADOS

Comando WHERE

O comando WHERE é usado para filtrar registros com base em condições específicas.

```
SELECT nome, email
FROM clientes
WHERE data_cadastro > TO_DATE('2023-01-01', 'YYYY-MM-DD');
```

Seleciona clientes cadastrados após 1º de janeiro de 2023.

```
SELECT nome, email
FROM clientes
WHERE email LIKE '%@example.com' AND data_cadastro BETWEEN TO_DATE('2023-01-01', 'YYYY-MM-DD') AND SYSDATE;
```

Aqui, filtramos clientes com emails de um domínio específico e cadastrados em um intervalo de datas.



ORDENANDO RESULTADOS

Comando ORDER BY

O comando ORDER BY é usado para ordenar os resultados de uma consulta

```
SELECT nome, email  
FROM clientes  
ORDER BY nome;
```

Os clientes são ordenados pelo nome em ordem alfabética.

```
SELECT nome, email, data_cadastro  
FROM clientes  
ORDER BY data_cadastro DESC, nome ASC;
```

Os resultados são ordenados primeiro pela data de cadastro em ordem decrescente e, em seguida, pelo nome em ordem crescente.



AGRUPANDO DADOS

Comando GROUP BY

O comando GROUP BY é usado para agrupar registros com valores comuns

```
SELECT data_cadastro, COUNT(*)  
FROM clientes  
GROUP BY data_cadastro;
```

Agrupar clientes pela data de cadastro e conta quantos clientes foram cadastrados em cada data.

```
SELECT data_cadastro, COUNT(*),  
AVG(LENGTH(nome)) AS Tamanho_Medio_Nome  
FROM clientes  
GROUP BY data_cadastro  
HAVING COUNT(*) > 1;
```

Retorna a quantidade de clientes por data apenas para datas com mais do que 1 cliente.



SUBCONSULTAS

```
SELECT nome, email
FROM clientes
WHERE id IN (SELECT cliente_id FROM
pedidos WHERE produto = 'Notebook');
```

Seleciona clientes que compraram um produto específico

```
SELECT nome, email
FROM clientes
WHERE id IN (
    SELECT cliente_id
    FROM pedidos
    GROUP BY cliente_id
    HAVING SUM(quantidade) > 5
);
```

Seleciona clientes que fizeram pedidos com um total de mais de 5 produtos



SQL



INSERT

Adicionando novos registros a uma tabela

INSERINDO DADOS SIMPLES

```
INSERT INTO clientes (id, nome, email,  
data_cadastro)  
VALUES (1, 'João Silva',  
'joao.silva@example.com',  
TO_DATE('2023-01-01', 'YYYY-MM-DD'));
```

Inserindo um novo cliente na tabela "clientes" com as informações fornecidas.



INSERINDO DADOS COM SUBCONSULTAS

```
INSERT INTO clientes_premium (id, nome,  
email, data_cadastro)  
SELECT id, nome, email, data_cadastro  
FROM clientes  
WHERE email LIKE '%@premium.com';
```

Inserindo clientes com emails de domínio
"premium.com" na tabela
"clientes_premium."



SQL



DELETE

Removendo registros de uma tabela

REMOVENDO REGISTROS SIMPLES

```
DELETE FROM clientes  
WHERE id = 1;
```

Remove da tabela "clientes" registro com o id=1.

```
DELETE FROM clientes;
```

Remove todos os registros da tabela "clientes".



REMOVENDO REGISTROS COM SUBCONSULTAS

```
DELETE FROM clientes  
WHERE id IN (SELECT cliente_id FROM  
pedidos WHERE data_pedido <  
TO_DATE('2023-01-01', 'YYYY-MM-DD'));
```

Remove clientes que fizeram pedidos antes
de 1º de janeiro de 2023



SQL



UPDATE

Atualizando registros de uma tabela

ATUALIZANDO REGISTROS SIMPLES

```
UPDATE clientes  
SET email = 'joao.silva@novoemail.com',  
    nome = 'João Pedro Silva'  
WHERE id = 1;
```

Atualizamos o email e o nome do cliente com o id=1.



ATUALIZANDO REGISTROS COM SUBCONSULTAS

```
UPDATE clientes  
SET status = 'vip'  
WHERE id IN (SELECT cliente_id FROM  
pedidos WHERE total_pedido > 500);
```

Atualizamos o status para 'vip' de clientes que fizeram pedidos com valor total acima de 500.



SQL



AGRADECIMENTOS

OBRIGADA POR LER ATÉ AQUI

Esse Ebook foi gerado por IA e diagramado por humano



<https://github.com/miriammrbeiro/ebook>

