


## Lista dos Comandos Mais Utilizados em SQL

Este documento apresenta uma lista dos comandos SQL mais utilizados, essenciais para manipulação e consulta de dados em bancos de dados relacionais. Cada comando é acompanhado de uma breve descrição e exemplos de uso, fornecendo uma referência prática para desenvolvedores, analistas de dados e estudantes interessados em aprimorar suas habilidades em SQL. Explore esta lista para entender e aplicar os conceitos fundamentais de SQL em seus projetos.

### 1. **SELECT:** Seleciona dados de uma tabela.Exemplo:


sql

 Copy code

```
SELECT * FROM clientes;
```

### 2. **INSERT INTO:** Insere novos registros em uma tabela.Exemplo:


sql

 Copy code

```
INSERT INTO clientes (nome, idade) VALUES ('João', 30);
```

### 3. **UPDATE:** Atualiza registros em uma tabela.Exemplo:


sql

 Copy code

```
UPDATE clientes SET idade = 31 WHERE nome = 'João';
```

### 4. **DELETE:** Remove registros de uma tabela.Exemplo:


sql

 Copy code

```
DELETE FROM clientes WHERE nome = 'João';
```

### 5. **CREATE TABLE:** Cria uma nova tabela no banco de dados.Exemplo:

sql


 Copy code

```
CREATE TABLE produtos (  
    id INT PRIMARY KEY,  
    nome VARCHAR(50),  
    preco DECIMAL(10,2)  
);
```



6. **ALTER TABLE:** Altera a estrutura de uma tabela existente.Exemplo:


sql

 Copy code

```
ALTER TABLE produtos ADD estoque INT;
```

7. **DROP TABLE:** Remove uma tabela do banco de dados.Exemplo:


sql

 Copy code

```
DROP TABLE clientes;
```

8. **JOIN:** Combina dados de duas ou mais tabelas com base em uma condição.Exemplo:


sql

 Copy code

```
SELECT clientes.nome, pedidos.produto  
FROM clientes  
INNER JOIN pedidos ON clientes.id = pedidos.cliente_id;
```

9. **GROUP BY:** Agrupa registros com base em uma coluna específica.Exemplo:


sql

 Copy code

```
SELECT departamento, COUNT(*) AS total_funcionarios  
FROM funcionarios  
GROUP BY departamento;
```

10. **ORDER BY:** Ordena os resultados por uma ou mais colunas.Exemplo:

sql


 Copy code

```
SELECT * FROM produtos ORDER BY preco DESC;
```



### 1. SELECT com LIKE para correspondência parcial:

sql


 Copy code

```
SELECT * FROM produtos WHERE nome LIKE 'Camiseta%';
```

Este comando seleciona todos os produtos cujo nome começa com "Camiseta".

### 2. SELECT com LIKE para correspondência em qualquer posição:

sql


 Copy code

```
SELECT * FROM clientes WHERE nome LIKE '%Silva%';
```

Este comando seleciona todos os clientes cujo nome contém "Silva" em qualquer posição.

### 3. SELECT com LIKE para correspondência exata:

sql


 Copy code

```
SELECT * FROM livros WHERE titulo LIKE 'O Senhor dos Anéis';
```

Este comando seleciona todos os livros com o título exato "O Senhor dos Anéis".

### 4. SELECT com LIKE para correspondência de caracteres específicos:

sql


 Copy code

```
SELECT * FROM produtos WHERE nome LIKE 'B%e';
```

Este comando seleciona todos os produtos cujo nome começa com "B" e termina com "e".

### 5. SELECT com LIKE para correspondência de um único caractere:

sql

 Copy code


```
SELECT * FROM palavras WHERE palavra LIKE '_rato';
```

Este comando seleciona todas as palavras que têm cinco letras, onde a segunda letra é "r" e a última letra é "o".



1. **COUNT**: Retorna o número de linhas que correspondem a um critério específico.Exemplo:


sql

 Copy code

```
SELECT COUNT(*) FROM clientes;
```

2. **SUM**: Retorna a soma dos valores em uma coluna.Exemplo:


sql

 Copy code

```
SELECT SUM(preco) FROM vendas;
```

3. **AVG**: Retorna a média dos valores em uma coluna.Exemplo:


sql

 Copy code

```
SELECT AVG(idade) FROM funcionarios;
```

4. **MAX**: Retorna o valor máximo em uma coluna.Exemplo:


sql

 Copy code

```
SELECT MAX(valor) FROM transacoes;
```

5. **MIN**: Retorna o valor mínimo em uma coluna.Exemplo:


sql

 Copy code

```
SELECT MIN(quantidade) FROM estoque;
```

6. **HAVING**: Permite filtrar resultados de um **GROUP BY** com base em uma condição.Exemplo:


sql

 Copy code

```
SELECT departamento, COUNT(*) AS total_funcionarios
FROM funcionarios
GROUP BY departamento
HAVING COUNT(*) > 5;
```

7. **DISTINCT**: Retorna valores únicos em uma coluna.Exemplo:


sql

 Copy code

```
SELECT DISTINCT categoria FROM produtos;
```

8. **LIMIT**: Limita o número de linhas retornadas em um resultado.Exemplo:


sql

 Copy code

```
SELECT * FROM clientes LIMIT 10;
```

9. **OFFSET**: Especifica o número de linhas a serem ignoradas antes de começar a retornar os resultados.Exemplo:

sql

 Copy code

```
SELECT * FROM clientes LIMIT 10 OFFSET 20;
```