

✧ SQL NINJA ✧

ARTE E TÉCNICA DAS
CONSULTAS BÁSICAS



GUSTAVO DINIZ

Introdução ao SQL

Conceitos e Consultas Essenciais

SQL (Structured Query Language) é a linguagem padrão utilizada para gerenciar e manipular bancos de dados relacionais. Com SQL, você pode criar, ler, atualizar e excluir dados, tornando-se uma ferramenta essencial para qualquer profissional que lida com dados. Este ebook foi criado para fornecer uma introdução clara e simplificada ao SQL, apresentando os principais conceitos e métodos de consultas. Cada capítulo traz exemplos práticos e contextos reais para ajudar você a entender e aplicar os fundamentos do SQL em suas atividades diárias. Se você é um iniciante em SQL, este guia é o ponto de partida perfeito para começar a explorar o poder desta linguagem.





01

Estrutura Básica de um Banco de Dados

A estrutura básica de um banco de dados é composta por tabelas, cada uma organizada em linhas (registros) e colunas (campos). As tabelas possuem chaves primárias para identificar unicamente cada registro e chaves estrangeiras para estabelecer relacionamentos entre tabelas. Este design permite a organização, armazenamento e recuperação eficiente dos dados. A integridade referencial e a normalização garantem a consistência e a minimização de redundâncias nos dados.

Bancos de Dados e Tabelas

1.1 Bancos de Dados e Tabelas

Um banco de dados é uma coleção organizada de dados. Dentro de um banco de dados, os dados são armazenados em tabelas, que são compostas por linhas e colunas.

***Banco de Dados:** Como um arquivo de planilha.

***Tabela:** Como uma planilha dentro desse arquivo.

1.2 Colunas e Linhas

Colunas: Campos ou atributos dos dados (ex: nome, idade, email).

Linhas: Cada linha representa um registro único.

1.3 Tipos de Dados

Os tipos de dados definem que tipo de informação pode ser armazenada em uma coluna.

INT: Números inteiros.

VARCHAR: Cadeias de caracteres.

DATE: Datas.



02

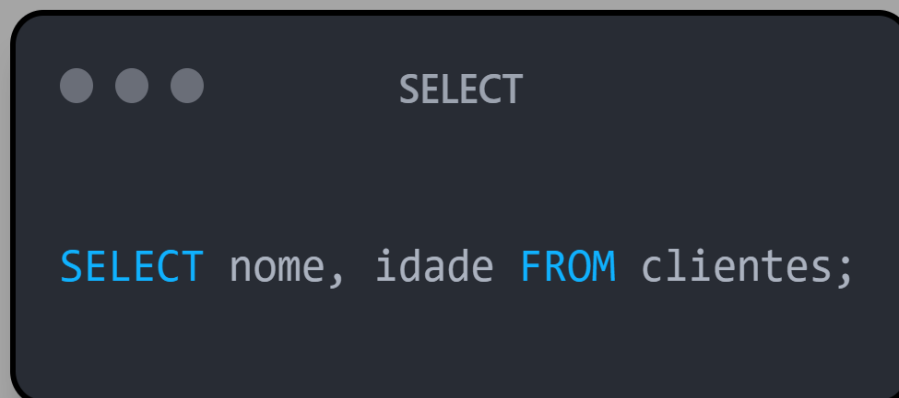
Consultas Básicas em SQL

Consultas básicas em SQL permitem realizar operações essenciais em bancos de dados, como selecionar, inserir, atualizar e excluir dados. A instrução `SELECT` é usada para recuperar dados de tabelas, enquanto `INSERT` adiciona novos registros. A instrução `UPDATE` modifica dados existentes, e `DELETE` remove registros específicos. Essas operações são fundamentais para a manipulação e gestão de dados.

Consultas Básicas em SQL

2.1 Selecionando Dados: *SELECT*

A consulta 'SELECT' é usada para buscar dados de uma tabela.

A terminal window with a dark background and rounded corners. It has three small gray circles in the top-left corner. The title bar in the top-right corner says "SELECT" in white. The main area contains the SQL query "SELECT nome, idade FROM clientes;" in a light blue monospace font.

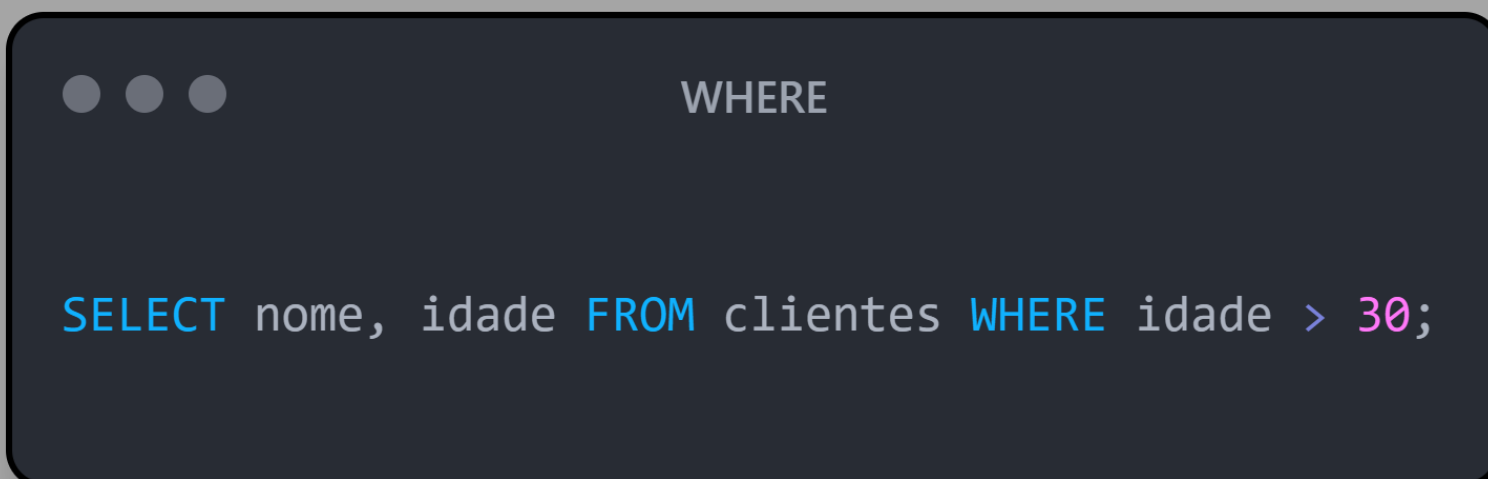
```
SELECT
```

```
SELECT nome, idade FROM clientes;
```

Retorna as colunas "nome" e "idade" da 'tabela' 'clientes'.

2.2 Filtrando Dados: *WHERE*

A cláusula 'WHERE' filtra os registros que atendem a uma condição específica.

A terminal window with a dark background and rounded corners. It has three small gray circles in the top-left corner. The title bar in the top-right corner says "WHERE" in white. The main area contains the SQL query "SELECT nome, idade FROM clientes WHERE idade > 30;" in a light blue monospace font, with the number 30 highlighted in pink.

```
WHERE
```

```
SELECT nome, idade FROM clientes WHERE idade > 30;
```

Retorna os nomes e idades de clientes com mais de 30 anos.

2.3 Ordenando Resultados: *ORDER BY*

A cláusula 'ORDER BY' ordena os resultados em ordem crescente ('ASC') ou decrescente ('DESC').



ORDER BY

```
SELECT nome, idade FROM clientes ORDER BY idade DESC;
```

Ordena os clientes pela idade, do mais velho ao mais novo.

2.4 Limitando resultados: 'LIMIT'

A cláusula LIMIT limita o número de registros retornados.



LIMIT

```
SELECT nome, idade FROM clientes ORDER BY idade DESC LIMIT 5;
```

Retorna os 5 clientes mais velhos.



03

Manipulação de Dados

A manipulação de dados em SQL envolve a inserção, atualização e exclusão de registros em um banco de dados. Esses comandos são fundamentais para manter os dados atuais, corretos e úteis. Neste capítulo, vamos explorar como adicionar novos registros, modificar informações existentes e remover dados desnecessários de uma tabela, sempre com exemplos práticos para facilitar a compreensão.

3.1 Inserindo Dados: 'INSERT INTO'

A consulta 'INSERT INTO' insere novos registros em uma tabela.

```
INSERT INTO clientes (nome, idade, email) VALUES ('João Silva', 28, 'joao@example.com');
```

Insere um novo cliente chamado *João Silva* na tabela 'clientes'.

3.2 Atualizando Dados: 'UPDATE'

A consulta 'UPDATE' modifica registros existentes em uma tabela.

```
UPDATE clientes SET email = 'joao.novoemail@example.com' WHERE nome = 'João Silva';
```

Atualiza o email do cliente João Silva.

3.3 Deletando Dados: 'DELETE'

A consulta 'DELETE' remove registros de uma tabela.

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

Remove o cliente João Silva da tabela 'clientes'.



04

Estruturas e Relacionamentos de Tabelas

A estrutura e os relacionamentos de tabelas são fundamentais no design de bancos de dados relacionais. Esses conceitos determinam como os dados são organizados, armazenados e interconectados, permitindo a criação de sistemas eficientes e escaláveis.

4.1 Criando Tabelas: 'CREATE TABLE'

A consulta 'CREATE TABLE' cria uma nova tabela.

```
CREATE TABLE clientes (  
    id INT PRIMARY KEY,  
    nome VARCHAR(100),  
    idade INT,  
    email VARCHAR(100)  
);
```

Cria a tabela "clientes" com colunas para "id", "nome", "idade" e "email".

4.2 Alterando Tabelas: 'ALTER TABLE'

A consulta 'ALTER TABLE' modifica a estrutura de uma tabela existente.

```
ALTER TABLE clientes ADD COLUMN telefone VARCHAR(20);
```

Adiciona uma nova coluna 'telefone' à tabela 'clientes'.

4.3 Relacionamentos entre Tabelas: 'JOIN'

As consultas 'JOIN' combinam registros de duas ou mais tabelas com base em uma condição relacionada

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

Retorna os IDs dos pedidos, nomes dos clientes e datas dos pedidos, combinando dados das tabelas 'pedidos' e 'clientes'.



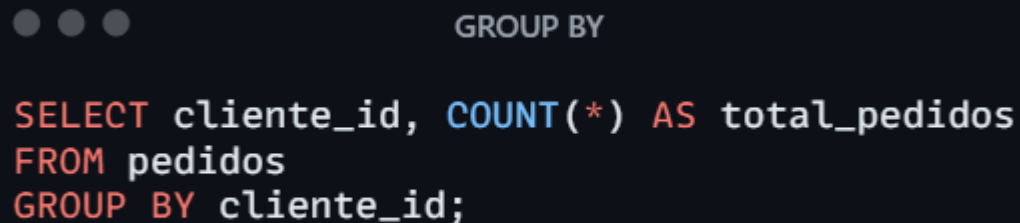
05

Consultas Avançadas

Consultas avançadas em bancos de dados relacionais são técnicas sofisticadas usadas para recuperar, manipular e analisar dados de forma eficiente e específica. Essas consultas vão além das operações básicas de seleção, inserção, atualização e exclusão, utilizando comandos e funções mais complexas do SQL (Structured Query Language).

5.1 Agregação de Dados: 'GROUP BY' e Funções Agregadas

A cláusula '*GROUP BY*' agrupa registros que têm valores em comum em colunas específicas. Funções agregadas como 'COUNT', 'SUM', 'AVG', 'MAX' e 'MIN' são usadas para realizar cálculos em grupos de dados.

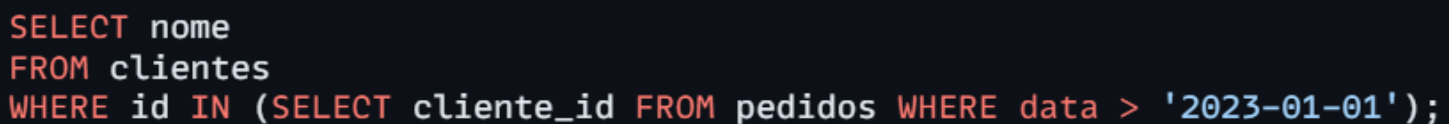
A screenshot of a code editor window titled "GROUP BY". It contains a SQL query that selects the client ID and the count of orders, grouped by client ID.

```
SELECT cliente_id, COUNT(*) AS total_pedidos
FROM pedidos
GROUP BY cliente_id;
```

Conta o número total de pedidos para cada cliente.

5.2 Subconsultas: 'Subquery'

Subconsultas são consultas aninhadas dentro de outra consulta.

A screenshot of a code editor window titled "SUBQUERY". It contains a SQL query that selects the names of clients whose IDs are in the list of orders placed after January 1, 2023.

```
SELECT nome
FROM clientes
WHERE id IN (SELECT cliente_id FROM pedidos WHERE data > '2023-01-01');
```

Retorna os nomes dos clientes que fizeram pedidos após 1º de janeiro de 2023.



AGRADECIMENTOS



✪ OBRIGADO POR LER ATÉ AQUI ✪

Este Ebook foi criado por IA, e diagramado por humano.

Este conteúdo foi gerado com a ajuda de inteligência artificial, especificamente o modelo GPT-4 desenvolvido pela OpenAI, para fornecer uma introdução clara e prática ao SQL. Espero que este material seja útil para os iniciantes em SQL e ajude a abrir portas no mundo da tecnologia da informação. Agradeço a todos os leitores e esperamos que vocês aproveitem e apliquem este conhecimento em seus estudos e carreiras.



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

<https://github.com/gudnz>