



A Era dos Comandos Fundamentais

Camila Cardozo



Desbravando o Mundo do SQL: Principais Comandos e Funções para Iniciantes

Se aventurar no universo do SQL pode parecer assustador inicialmente, mais com os comandos e funções certos, você pode dominar o poder do banco de dados. Neste pequeno ebook, exploraremos os principais comandos SQL de uma maneira simples, fornecendo exemplos práticos para consolidar seu entendimento.

01

SELECT

(Obtendo Dados Precisos).



SELECT - Obtendo Dados Precisos.

O comando SELECT é a base de qualquer consulta SQL, permitindo recuperar dados de uma tabela. Vejamos um exemplo:

```
SELECT nome, idade FROM usuarios WHERE cidade = 'São Paulo';
```

Neste exemplo, estamos selecionando os campos nome e idade da tabela usuarios apenas para registros onde a cidade é 'São Paulo'.



02

INSERT

(Adicionando Dados Novos).



INSERT: Adicionando Dados Novos.

O comando INSERT é essencial para adicionar novos registros à sua tabela. Vamos ver como podemos inserir um novo usuário:

```
INSERT INTO usuarios (nome, idade, cidade) VALUES ('Maria', 25, 'Rio de Janeiro');
```

Aqui, estamos adicionando uma nova entrada na tabela usuarios com o nome 'Maria', idade 25 e cidade 'Rio de Janeiro'.



03

UPDATE

(Modificando Dados Existente).



UPDATE: Modificando Dados Existentes.

Com o comando UPDATE, podemos fazer alterações nos dados já existentes. Suponhamos que precisamos corrigir a idade de um usuário:

```
UPDATE usuarios  
  SET idade = 26  
 WHERE nome = 'Maria';
```

Neste exemplo, estamos atualizando a idade de 'Maria' para 26.



04

DELETE

(Removendo Dados Indesejados).



DELETE: Removendo Dados Indesejados.

Para excluir registros, utilizamos o comando DELETE. Aqui está um exemplo:

```
DELETE FROM usuarios WHERE idade > 30;
```

Este comando remove todos os usuários com mais de 30 anos da tabela usuarios.



05

WHERE

(Refinando Resultados).



WHERE: Refinando Resultados

O operador WHERE é crucial para filtrar resultados. Vamos combinar com o SELECT para uma consulta mais detalhada:

```
SELECT nome,  
        cidade  
FROM usuarios  
WHERE idade BETWEEN 20 AND 30;
```

Este comando retorna nomes e cidades dos usuários com idades entre 20 e 30 anos.



06

LIKE

(Pesquisando com Flexibilidade).



LIKE: Pesquisando com Flexibilidade.

O operador LIKE é útil para pesquisas flexíveis. Por exemplo, para encontrar usuários com nomes que contenham "Ana":

```
SELECT nome,  
        cidade  
FROM usuarios  
WHERE nome LIKE '%Ana%';
```

Este comando retorna nomes e cidades dos usuários cujo nome contém "Ana", independentemente da posição.



07

Funções Agregadas (Analisando Dados de Forma Resumida).



Funções Agregadas: Analisando Dados de Forma Resumida

O SQL oferece diversas funções agregadas para análises mais simples e complexas, agora apresentaremos as funções mais importantes para desenvolvedores iniciantes:

```
--AVG: Calcula a média de uma coluna numérica.  
SELECT AVG(idade) FROM usuarios;  
--SUM: Soma os valores de uma coluna numérica.  
SELECT SUM(idade) FROM usuarios;  
--COUNT: Conta o número de linhas em uma tabela.  
SELECT COUNT(*) FROM usuarios;  
--MAX: Retorna o valor máximo em uma coluna.  
SELECT MAX(idade) FROM usuarios;  
--MIN: Retorna o valor mínimo em uma coluna.  
SELECT MIN(idade) FROM usuarios;  
--DISTINCT: Retorna valores únicos de uma coluna.  
SELECT DISTINCT cidade FROM usuarios;
```

Estas funções ajudam a extrair insights valiosos de seus dados de forma eficiente.



08

ORDER BY (Ordenando Resultados).



ORDER BY: Ordenando Resultados

O comando ORDER BY é essencial para organizar os resultados de acordo com critérios específicos. Por exemplo, para ordenar os usuários por idade de forma decrescente:

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

Este comando retorna nomes e idades dos usuários ordenados de forma decrescente pela idade.



09

JOIN

(Unindo Dados de Tabelas Relacionadas).



JOIN: Unindo Dados de Tabelas Relacionadas

O comando JOIN é usado para combinar linhas de duas ou mais tabelas com base em uma condição relacionada. Suponhamos que temos uma tabela pedidos relacionada a usuarios:

```
SELECT usuarios.nome,  
        pedidos.produto  
FROM usuarios  
JOIN pedidos  
    ON usuarios.id = pedidos.usuario_id;
```

Neste exemplo, estamos unindo as tabelas usuarios e pedidos para obter o nome do usuário e o produto do pedido.





Estes são apenas alguns dos principais comandos e funções SQL que você pode usar para interagir com seus dados.

Com a prática, você se sentirá cada vez mais confortável explorando e manipulando bancos de dados.

Continue experimentando e expandindo seu conhecimento neste fascinante mundo do SQL!



OBRIGADA POR LER ATÉ AQUI.

Esse ebook foi gerado por uma IA e diagramado por um humano.

O conteúdo foi gerado para fins didáticos de construção, não foi realizada uma avaliação humana cuidadosa, por isso o conteúdo pode apresentar erros gerados pela IA.