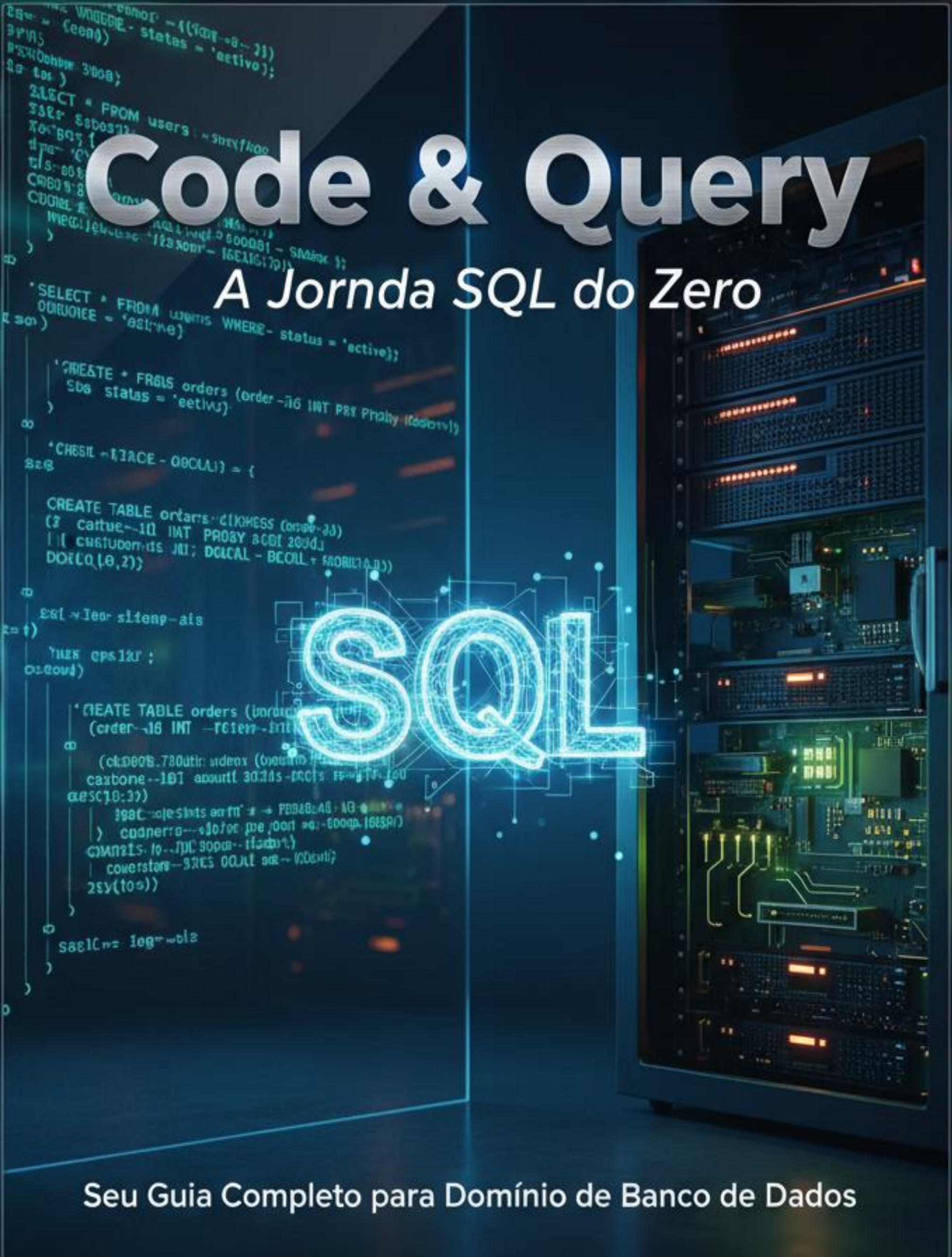
Code & Query

A Jornda SQL do Zero



Seu Guia Completo para Domínio de Banco de Dados

01

O COMEÇO DA JORNADA

Code & Query - Augusto Belussi

Entendendo a Linguagem SQL

O que é SQL?

SQL (*Structured Query Language*) é a linguagem usada para se comunicar com bancos de dados. Ela permite que você **peça informações**, **adicione novos dados**, **modifique registros existentes** e até **crie estruturas** inteiras dentro do banco.

Mesmo que pareça técnico, pense em SQL como “frases” que você diz para o banco de dados executar uma ação.

A Importância do SQL no Mundo Real

Por que SQL é tão importante?

Praticamente tudo que usamos no dia a dia depende de informação armazenada:

- Redes sociais usam bancos de dados para guardar perfis e posts.

- Apps de delivery armazenam pedidos, restaurantes e entregadores.

- Bancos registram transações, contas e clientes.

Sem SQL, esses sistemas não teriam como guardar, buscar e organizar essas informações. Aprender SQL significa entender como os dados se movimentam no “coração” de qualquer aplicação moderna.

02

ENTENDENDO O BANCO DE DADOS

Code & Query - Augusto Belussi

O que é um Banco de Dados?

A “Biblioteca de Dados”

Um banco de dados é um sistema criado para **armazenar informações de forma organizada**. Imagine uma grande biblioteca: ao invés de livros, temos **dados**. Cada parte desse sistema serve para facilitar o acesso rápido e seguro às informações..

Estrutura Fundamental

Tabelas

As tabelas são as estruturas principais do banco. Cada tabela armazena informações sobre um assunto específico .Exemplos de tabelas comuns: **clientes**, **produtos**, **pedidos**.

Colunas e Tipos de Dados

As colunas definem *o tipo de informação* que será armazenada em cada tabela. Exemplo de colunas de uma tabela de clientes:

- **nome** → texto
- **idade** → número
- **email** → texto
- **data_cadastro** → data e hora

Esses tipos de dados garantem que cada informação seja armazenada corretamente.

Linhas (Registros)

Cada linha é um item real armazenado na tabela. Se a tabela é **clientes**, cada linha representa *um cliente específico*.

03

CONSULTANDO DADOS

Code & Query - Augusto Belussi

SELECT – Buscando Informações

Comando SELECT

O comando **SELECT** é o mais usado no SQL, pois permite consultar informações de uma tabela.
Exemplo:

Code & Query - Augusto Belussi

```
SELECT * FROM clientes;
```

O *asterisco* significa “traga todas as colunas”.

Selecionando colunas específicas

Code & Query - Augusto Belussi

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

Assim, você vê apenas as informações de nomes da tabela clientes.

Filtrando com WHERE

Usando WHERE

O **WHERE** permite buscar dados que atendam a uma condição:

Code & Query - Augusto Belussi

```sql
SELECT * FROM clientes
WHERE idade > 18;
```

Combinando condições

Você pode combinar condições:

Code & Query - Augusto Belussi

```sql
WHERE idade > 18 AND cidade = 'São Paulo';
```

Buscando partes de um texto (LIKE)

Usando LIKE

Procura por pedaços de texto:

Code & Query - Augusto Belussi

```sql
SELECT * FROM produtos
WHERE nome LIKE '%mouse%';
```

O símbolo % significa “qualquer sequência de caracteres”.

04

Criando e Inserindo Dados

Code & Query - Augusto Belussi

Criando Tabelas com CREATE TABLE

Definindo a estrutura

Ao criar uma tabela, você define suas colunas:

Code & Query - Augusto Belussi

```sql
CREATE TABLE produtos (
  id INT PRIMARY KEY,
  nome VARCHAR(100),
  preco DECIMAL(10,2),
  estoque INT
);
```

- **PRIMARY KEY** → identifica cada item de forma única
- **VARCHAR** → texto variável
- **DECIMAL** → números com casas decimais
- **INT** → números inteiros

Inserindo Registros com INSERT INTO

Adicionando novos dados

Code & Query - Augusto Belussi

```sql
INSERT INTO produtos (id, nome, preco, estoque)
VALUES (1, 'Teclado Mecânico', 199.90, 50);
```

Inserindo vários valores de uma vez

Code & Query - Augusto Belussi

```sql
INSERT INTO produtos (id, nome, preco)
VALUES
(2, 'Mouse Gamer', 129.00),
(3, 'Headset', 249.90);
```

05

Atualizando e Deletando Dados

Code & Query - Augusto Belussi

Atualizando Informações com UPDATE

Modificando registros

Code & Query - Augusto Belussi

```sql
UPDATE produtos
SET preco = 179.90
WHERE id = 1;
```

Sem o WHERE, *todas* as linhas seriam alteradas — por isso, cuidado!

Removendo Dados com DELETE

Excluindo registros

Code & Query - Augusto Belussi

```sql
DELETE FROM produtos
WHERE id = 2;
```

Assim como no UPDATE, sempre inclua o WHERE para evitar apagar tudo sem querer.

06

Organizando e Classificando Dados

Code & Query - Augusto Belussi

Ordenação com ORDER BY

Ordenando resultados

Code & Query - Augusto Belussi

```sql
SELECT nome, preco
FROM produtos
ORDER BY preco ASC;
```

- ASC → crescente
- DESC → decrescente

Limitando Resultados com LIMIT

Mostrar poucos resultados

Code & Query - Augusto Belussi

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

Ótimo para visualizar apenas parte dos dados.

Removendo resultados duplicados (DISTINCT)

Evitar repetições

Code & Query - Augusto Belussi

```
SELECT DISTINCT cidade
FROM clientes;
```

Isso retorna cada cidade apenas uma vez.

07

Avançando um Pouco Mais

Code & Query - Augusto Belussi

Junções (JOINs)

Conectando tabelas

JOINs conectam dados de duas tabelas diferentes.
Exemplo de junção simples:

Code & Query - Augusto Belussi

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

Isso permite unir dados relacionados, como pedidos + nome do cliente.

Funções de agregação

Cálculos e resumos

Usadas para cálculos:

- **COUNT()** → conta registros
- **SUM()** → soma valores
- **AVG()** → média
- **MAX()** → maior valor
- **MIN()** → menor valor
- **AS** → definir um pseudônimo (ou alias) para a coluna resultante da operação

Exemplo:

Code & Query - Augusto Belussi

```sql
SELECT AVG(preco) AS preco_medio
FROM produtos;
```

08

Conclusão da Jornada

Revisão Geral

O que você aprendeu

Neste eBook você viu:

- Como funciona um banco de dados
- Estruturas como tabelas, colunas e registros
- Consultas com SELECT
- Filtros com WHERE
- Ordenação e organização dos resultados
- Como criar, inserir, atualizar e apagar dados
- Introdução a JOINs e funções avançadas

Mensagem Final

Encerramento e próximos passos

SQL é uma habilidade essencial e poderosa. Com ela, você domina a linguagem que controla os dados do mundo moderno.

Esse é apenas o primeiro passo — e o próximo capítulo dessa jornada é você quem escreve.