

Aula 13

Modificando e Excluindo Dados
(DML - UPDATE e DELETE)

(Slide 1 de 4) - Lembrando da Aula Anterior

- **O que fizemos?**
 - Demos nosso primeiro passo no DML (Data Manipulation Language).
 - Usamos o comando `INSERT INTO` para popular nossas tabelas com dados.
- **Resultado:** Nosso banco de dados agora tem registros, está "vivo".
- **A pergunta de hoje:**
 - O que fazer quando um dado está errado, desatualizado ou precisa ser removido?

(Slide 2 de 4) - Corrigindo Informações com UPDATE

- O comando UPDATE é usado para **modificar** registros que **já existem** na tabela.
- **Sintaxe básica:**
 - UPDATE nome_da_tabela
 - SET coluna1 = 'novo_valor', coluna2 = 123
- **O Papel do SET :**
 - A cláusula SET indica qual(is) coluna(s) você quer alterar e quais os novos valores.
- **Atenção:** Usar o comando UPDATE sozinho é **extremamente perigoso**, pois ele alteraria **TODAS AS LINHAS** da tabela.

(Slide 3 de 4) - A Cláusula Mais Importante do Dia: WHERE

- A cláusula `WHERE` é o nosso **filtro de segurança**. Ela especifica **quais registros** serão afetados pelo `UPDATE` ou `DELETE`.
- **Como funciona?**
 - `UPDATE tabela SET ... WHERE condicao;`
 - `DELETE FROM tabela WHERE condicao;`
- **A Regra de Ouro do Professor:**
 - Antes de rodar um `UPDATE` ou `DELETE`, sempre teste sua cláusula `WHERE` com um `SELECT`.
 - `SELECT * FROM tabela WHERE condicao;`. Se o `SELECT` retornar as linhas que você quer afetar, sua condição está correta.

(Slide 4 de 4) - Removendo Registros com DELETE

- O comando `DELETE` é usado para **remover permanentemente** uma ou mais linhas da tabela.
- **Sintaxe:**
 - `DELETE FROM nome_da_tabela WHERE condicao;`
- **PERIGO MÁXIMO:**
 - Um comando `DELETE FROM tabela` **sem a cláusula** `WHERE` irá apagar **TODOS OS DADOS DA TABELA** sem pedir confirmação.
- **Lembre-se:** A Regra de Ouro vale aqui também. Teste o `WHERE` com `SELECT` primeiro!

Prática

```
5      -- 1. SETUP: Criando um banco de dados novo e uma tabela para o exemplo.
6 •    CREATE DATABASE IF NOT EXISTS aula4_exemplo;
7 •    USE aula4_exemplo;
8
9 •    CREATE TABLE produtos (
10         produto_id INT PRIMARY KEY AUTO_INCREMENT,
11         nome VARCHAR(100),
12         quantidade_estoque INT
13     );
14
15 •    select * from produtos;
```

```
16
17 -- 2. POPULANDO A TABELA: Inserindo alguns dados para podermos trabalhar.
18 • INSERT INTO produtos (nome, quantidade_estoque) VALUES
19     ('Caneta Azul', 100),
20     ('Caderno 96fls', 50),
21     ('Borracha', 200);
22
23 -- Vamos conferir o que inserimos.
24 • SELECT * FROM produtos;
25
```



```
--
26  -- 3. PRÁTICA DE UPDATE: O estoque do 'Caderno' foi contado errado, o correto é 45.
27  -- Primeiro, a verificação de segurança com SELECT.
28 • SELECT * FROM produtos WHERE nome = 'Caderno 96fls';
29
30  -- Agora, o comando UPDATE com segurança.
31 • UPDATE produtos
32   SET quantidade_estoque = 45
33   WHERE nome = 'Caderno 96fls';
34
35  -- Vamos conferir o resultado.
36 • SELECT * FROM produtos;
```

```
38      --      Agora, o comando UPDATE com segurança.
39 •  UPDATE produtos
40     SET quantidade_estoque = 45
41     WHERE produto_id = 2;
42
```

```
39  -- 4. PRÁTICA DE DELETE: O produto 'Borracha' saiu de linha e será removido.
40  -- Verificação de segurança com SELECT, usando o ID.
41 • SELECT * FROM produtos WHERE produto_id = 3;
42
43  -- Comando DELETE com segurança.
44 • DELETE FROM produtos
45  WHERE produto_id = 3;
46
47  -- Verificação final. O produto 'Borracha' não existe mais.
48 • SELECT * FROM produtos;
```


Exercícios

Altere o primeiro produto da
tabela e exclua a última