

Lista de Exercícios: Comando UPDATE no MySQL

1. Atualizar Dados Específicos

- Crie uma tabela chamada **estudantes** com as seguintes colunas:
 - **id** (INT, auto incremento, chave primária)
 - **nome** (VARCHAR(100), não nulo)
 - **nota** (DECIMAL(5,2), não nulo)
- Insira ao menos 5 registros na tabela **estudantes**.
- Atualize a nota do estudante com **id** 3 para 9.50.

2. Atualização Condicional

- Crie uma tabela chamada **produtos** com as seguintes colunas:
 - **id** (INT, auto incremento, chave primária)
 - **nome** (VARCHAR(100), não nulo)
 - **preco** (DECIMAL(10,2), não nulo)
- Insira ao menos 5 registros na tabela **produtos**.
- Aumente o preço de todos os produtos com preço menor que 100 em 15%.

3. Atualização de Múltiplas Colunas

- Suponha que você tem uma tabela **clientes** com as seguintes colunas:
 - **id** (INT, auto incremento, chave primária)
 - **nome** (VARCHAR(100), não nulo)
 - **email** (VARCHAR(100), pode ser nulo)
 - **telefone** (VARCHAR(15), pode ser nulo)
- Insira ao menos 5 registros na tabela **clientes**.
- Atualize o nome e o telefone do cliente com **id** 4 para 'Ana Costa' e '99999-8888', respectivamente.

4. Atualização com Subconsulta

- Crie uma tabela chamada **vendas** com as seguintes colunas:
 - **id** (INT, auto incremento, chave primária)
 - **produto_id** (INT, não nulo)
 - **valor** (DECIMAL(10,2), não nulo)
- Crie outra tabela chamada **ajustes_valores** com as seguintes colunas:
 - **produto_id** (INT, não nulo)
 - **ajuste** (DECIMAL(10,2), não nulo)
- Insira registros nas tabelas **vendas** e **ajustes_valores**.
- Atualize os valores na tabela **vendas** com base nos ajustes na tabela **ajustes_valores**. O novo valor de cada venda deve ser calculado como **valor + ajuste**.

5. Atualização em Massa

- Suponha que você tem uma tabela **funcionarios** com as seguintes colunas:
 - **id** (INT, auto incremento, chave primária)
 - **nome** (VARCHAR(100), não nulo)

- `salario` (DECIMAL(10,2), não nulo)
- `data_admissao` (DATE, não nulo)
- Insira ao menos 5 registros na tabela `funcionarios`.
- Aumente o salário de todos os funcionários contratados antes de 2020 em 10%.

6. Atualização com Condições Múltiplas

- Crie uma tabela chamada `pedidos` com as seguintes colunas:
 - `id` (INT, auto incremento, chave primária)
 - `cliente_id` (INT, não nulo)
 - `status` (VARCHAR(20), não nulo)
 - `valor_total` (DECIMAL(10,2), não nulo)
- Insira ao menos 5 registros na tabela `pedidos`.
- Atualize o status de todos os pedidos com `valor_total` acima de 500 e que pertencem ao cliente com `cliente_id` 2 para 'Aprovado'.

7. Verificação Antes da Atualização

- Crie uma tabela chamada `estoque` com as seguintes colunas:
 - `id` (INT, auto incremento, chave primária)
 - `produto` (VARCHAR(100), não nulo)
 - `quantidade` (INT, não nulo)
- Insira ao menos 5 registros na tabela `estoque`.
- Antes de atualizar a quantidade de um produto específico, faça uma consulta para verificar a quantidade atual do produto com `id` 3.

8. Atualização Condicional com Subconsulta

- Suponha que você tenha uma tabela `eventos` com as seguintes colunas:
 - `id` (INT, auto incremento, chave primária)
 - `data_evento` (DATE)
 - `local` (VARCHAR(100))
- Crie outra tabela chamada `novos_eventos` com a mesma estrutura.
- Insira registros nas tabelas `eventos` e `novos_eventos`.
- Atualize os registros na tabela `eventos` com base nas informações da tabela `novos_eventos`, onde o `data_evento` na tabela `eventos` deve ser atualizado para a data mais recente encontrada na tabela `novos_eventos` para o mesmo `id`.