



JOIN Academy

# Comandos Intermediários

Manipulando Dados e Estruturas

INSERT

UPDATE

DELETE

ALTER TABLE

# INSERT – Inserindo Dados

O comando **INSERT** é usado para adicionar novos registros em uma tabela. É fundamental para popular seu banco de dados com informações.

## SINTAXE BÁSICA:

```
INSERT INTO nome_tabela  
(coluna1, coluna2, coluna3)  
VALUES  
(valor1, valor2, valor3);
```

## DICAS IMPORTANTES:

- ✓ A ordem dos valores deve corresponder à ordem das colunas
- ✓ Textos devem estar entre aspas simples
- ✓ Você pode inserir múltiplos registros de uma vez

## EXEMPLO 1: Inserir um único registro

```
INSERT INTO alunos  
(nome, idade, cidade)  
VALUES  
( 'João Silva', 20, 'São Paulo');
```

## EXEMPLO 2: Inserir múltiplos registros

```
INSERT INTO alunos  
(nome, idade, cidade)  
VALUES  
( 'Maria Santos', 22, 'Rio'),  
( 'Pedro Costa', 19, 'Brasília'),  
( 'Ana Lima', 21, 'São Paulo');
```

## EXEMPLO 3: Inserir sem especificar colunas

```
INSERT INTO alunos  
VALUES  
(1, 'Lucas Souza', 23, 'Curitiba');
```

# UPDATE – Atualizando Dados

O comando **UPDATE** modifica dados existentes. Use-o para corrigir ou atualizar informações já cadastradas.

## SINTAXE BÁSICA:

```
UPDATE nome_tabela  
SET coluna1 = novo_valor1,  
    coluna2 = novo_valor2  
WHERE condição;
```

## ⚠ ATENÇÃO!

**SEMPRE use WHERE no UPDATE!** Se você esquecer o WHERE, TODOS os registros da tabela serão atualizados, causando perda de dados irreversível!

## ANTES DO UPDATE

ID	Nome	Idade	Cidade
1	João	20	São Paulo
2	Maria	22	Rio
3	Pedro	19	Brasília

```
UPDATE alunos  
SET idade = 21, cidade = 'Curitiba'  
WHERE id = 1;
```

## DEPOIS DO UPDATE

ID	Nome	Idade	Cidade
1	João	21	Curitiba
2	Maria	22	Rio
3	Pedro	19	Brasília

# DELETE – Removendo Dados

O comando **DELETE** remove registros de uma tabela. Use com muito cuidado, pois a exclusão é permanente!

## SINTAXE BÁSICA:

```
DELETE FROM nome_tabela  
WHERE condição;
```

## PERIGO!

**NUNCA execute DELETE sem WHERE!** Isso apagará TODOS os registros da tabela de forma irreversível. Sempre faça backup antes de deletar dados importantes!

### ANTES DO DELETE

ID	Nome	Idade	Cidade
1	João	20	São Paulo
2	Maria	22	Rio
3	Pedro	19	Brasília

```
DELETE FROM alunos  
WHERE id = 2;
```

### DEPOIS DO DELETE

ID	Nome	Idade	Cidade
1	João	20	São Paulo
3	Pedro	19	Brasília

# ALTER TABLE – Modificando Estruturas

O comando **ALTER TABLE** modifica a estrutura de uma tabela existente. Use para adicionar, remover ou modificar colunas.

## **ADD – Adicionar Coluna**

Adiciona uma nova coluna à tabela existente.

```
ALTER TABLE alunos  
ADD email VARCHAR(100);
```

## **DROP – Remover Coluna**

Remove uma coluna da tabela (e todos os seus dados).

```
ALTER TABLE alunos  
DROP COLUMN telefone;
```

## **MODIFY – Modificar Coluna**

Altera o tipo de dados ou propriedades de uma coluna.

```
ALTER TABLE alunos  
MODIFY COLUMN nome VARCHAR(150);
```

## **RENAME – Renomear Coluna**

Altera o nome de uma coluna existente.

```
ALTER TABLE alunos  
RENAME COLUMN cidade TO municipio;
```

## **RENAME TABLE – Renomear Tabela**

Altera o nome da tabela inteira.

```
ALTER TABLE alunos  
RENAME TO estudantes;
```

## **ADD CONSTRAINT – Adicionar Restrição**

Adiciona chaves primárias, estrangeiras ou outras restrições.

```
ALTER TABLE alunos  
ADD PRIMARY KEY (id);
```

# Exercícios Práticos

Pratique os comandos que você aprendeu!

## 1 INSERT

*Insira 3 novos produtos na tabela 'produtos' com nome, preço e categoria.*

## 2 UPDATE

*Aumente em 10% o salário de todos os funcionários do departamento 'Vendas'.*

## 3 DELETE

*Remova todos os pedidos com status 'Cancelado' da tabela 'pedidos'.*

## 4 ALTER TABLE

*Adicione uma coluna 'data\_nascimento' do tipo DATE na tabela 'clientes'.*



## DESAFIO COMPLETO!

Crie uma nova tabela, insira dados, atualize alguns registros, delete outros, e por fim adicione uma nova coluna. Pratique a sequência completa de comandos!

# Resoluções dos Exercícios

Aqui estão as soluções para os exercícios propostos!

## 1 INSERT - Solução

```
INSERT INTO produtos  
(nome, preco, categoria)  
VALUES  
( 'Notebook', 3500, 'Eletrônicos'),  
( 'Mouse', 150, 'Periféricos'),  
( 'Teclado', 450, 'Periféricos');
```

## 2 UPDATE - Solução

```
UPDATE funcionarios  
SET salario = salario * 1.10  
WHERE departamento = 'Vendas';
```

## 3 DELETE - Solução

```
DELETE FROM pedidos  
WHERE status = 'Cancelado';
```

## 4 ALTER TABLE - Solução

```
ALTER TABLE clientes  
ADD data_nascimento DATE;
```

## DESAFIO COMPLETO - Solução

```
CREATE TABLE produtos_vendas (id INT PRIMARY KEY, nome VARCHAR(100), preco DECIMAL(10,2));  
INSERT INTO produtos_vendas VALUES (1, 'Produto A', 100), (2, 'Produto B', 200);  
UPDATE produtos_vendas SET preco = 150 WHERE id = 1;  
DELETE FROM produtos_vendas WHERE preco < 120;  
ALTER TABLE produtos_vendas ADD estoque INT;
```

# Domine os Comandos!

Agora você sabe manipular dados e estruturas em SQL. Com INSERT, UPDATE, DELETE e ALTER TABLE, você tem o poder de gerenciar completamente seu banco de dados!



Sempre Use  
WHERE



Faça  
Backups



Pratique  
Sempre

