

# Banco de Dados

Professor Maurício Buess

[mbuess@up.edu.br](mailto:mbuess@up.edu.br)

[github.com/mauriciobuess](https://github.com/mauriciobuess)

## **Objetivo:**

- Revisar os conceitos básicos de bancos de dados relacionais.
- Praticar a criação de banco de dados e tabelas.
- Aprender e praticar os comandos SQL INSERT, SELECT, UPDATE e DELETE.

## Revisão Rápida

- Banco de Dados: Armazena e organiza dados em tabelas, disponibilizando outros objetos de auxílio à consultas e manutenção de tais dados.
- Tabela: Estrutura que organiza dados em linhas e colunas.
- Chave Primária: Identificador único para cada linha na tabela.
- Chave Estrangeira: Relaciona uma tabela com outra, criando vínculos entre elas.

## Criação do Banco de Dados e Tabelas

- Crie um banco de dados chamado empresa e uma tabela chamada funcionarios:
  - Tabela funcionarios:
  - id (INT, PRIMARY KEY, AUTO\_INCREMENT)
  - nome (VARCHAR(100))
  - cargo (VARCHAR(50))
  - salario (DECIMAL(10, 2))

## Criação do Banco de Dados e Tabelas

```
CREATE DATABASE empresa;  
USE empresa;
```

```
CREATE TABLE funcionarios (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(100),  
    cargo VARCHAR(50),  
    salario DECIMAL(10, 2)  
);
```

## Criação do Banco de Dados e Tabelas

```
CREATE DATABASE empresa;  
USE empresa;
```

```
CREATE TABLE funcionarios (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(100),  
    cargo VARCHAR(50),  
    salario DECIMAL(10, 2)  
);
```

## Inserção e Consulta de Dados

- Insira três registros na tabela funcionarios com os seguintes dados:
  - Nome: Ana Silva, Cargo: Gerente, Salário: 8000.00
  - Nome: Pedro Santos, Cargo: Analista, Salário: 5000.00
  - Nome: Maria Oliveira, Cargo: Assistente, Salário: 3000.00

## Inserção e Consulta de Dados

```
INSERT INTO funcionarios (nome, cargo, salario) VALUES ('Ana Silva', 'Gerente', 8000.00);
```

```
INSERT INTO funcionarios (nome, cargo, salario) VALUES ('Pedro Santos', 'Analista', 5000.00);
```

```
INSERT INTO funcionarios (nome, cargo, salario) VALUES ('Maria Oliveira', 'Assistente', 3000.00);
```



## Inserção e Consulta de Dados

- 1) Selecione todos os dados da tabela funcionarios:
- 2) Selecione todos os funcionários por ordem alfabética de nome:
- 3) Selecione todos os funcionários que não atuam como gerente:
- 4) Qual o comando que informa o salário do(s) gerente(s);

## Inserção e Consulta de Dados

- Faça a inclusão dos seguintes funcionários:

- 1)Nome: 'Ana Silva', cargo: 'Analista', salario: 3500.00;
- 2)Nome: 'João Souza', cargo: 'Desenvolvedor', salario: 4500.00;
- 3)Nome: 'Maria Oliveira', cargo: 'Gerente', salario: 6000.00;
- 4)Nome: 'Carlos Pereira', cargo: 'Designer', salario: 3200.00;

## Inserção e Consulta de Dados

```
INSERT INTO funcionarios (nome, cargo, salario)
VALUES ('Ana Silva', 'Analista', 3500.00),
       ('João Souza', 'Desenvolvedor', 4500.00),
       ('Maria Oliveira', 'Gerente', 6000.00),
       ('Carlos Pereira', 'Designer', 3200.00);
```

## Comando UPDATE

- UPDATE → usado para modificar os dados existentes em uma tabela.

Sintaxe:

```
UPDATE nome_da_tabela  
SET coluna1 = valor1, coluna2 = valor2, ...  
WHERE condição;
```

**Importante: O UPDATE sem uma cláusula WHERE atualiza todos os registros da tabela. Sempre use WHERE para especificar quais registros devem ser modificados.**

## Comando UPDATE

- A direção da empresa concedeu um aumento salarial para a funcionária 'Ana Silva' de 10%:
- Dica:
  - Use o comando SELECT para verificar se a condição do WHERE está correta e, também, para verificar os dados do registro.
  - Execute o comando UPDATE copiando a cláusula WHERE do comando SELECT;
  - Repita o comando SELECT para verificar o êxito do comando UPDATE.

## Comando UPDATE

```
SELECT * FROM funcionarios WHERE nome = 'Ana Silva';
```

```
UPDATE funcionarios  
SET salario = salario * 1.10  
WHERE nome = 'Ana Silva';
```

```
SELECT * FROM funcionarios WHERE nome = 'Ana Silva';
```

## Comando UPDATE

Como alterar o cargo de todos os funcionários de 'Designer' para 'Designer Sênior':

## Comando UPDATE

```
SELECT * FROM funcionarios WHERE cargo = 'Designer';
```

```
UPDATE funcionarios  
SET cargo = 'Designer Sênior'  
WHERE cargo = 'Designer';
```

```
SELECT * FROM funcionarios WHERE cargo = 'Designer';
```



## Comando UPDATE

- O funcionário 'João Souza' foi promovido à 'Lider Técnico' com a remuneração de R\$ 5000,00.

## Comando UPDATE

- O funcionário 'João Souza' foi promovido à 'Lider Técnico' com a remuneração de R\$ 5000,00.

```
UPDATE funcionarios  
SET salario = 5000.00, cargo = 'Líder Técnico'  
WHERE nome = 'João Souza';
```

## Comando UPDATE

- A funcionária 'Olinda Nentanto' foi promovido de 'Expositora' para 'Vendedora', passando a ganhar R\$ 2300,00.

```
UPDATE funcionarios  
SET salario = 2300.00, cargo = 'Vendedora'  
WHERE nome = 'Olinda Nentanto';
```

## Comando DELETE

- O comando DELETE é usado para remover dados de uma tabela.

Sintaxe:

```
DELETE FROM nome_da_tabela  
WHERE condição;
```

## Comando DELETE

- O comando DELETE é usado para remover dados de uma tabela.

Sintaxe:

```
DELETE FROM nome_da_tabela  
WHERE condição;
```

**\*\* Importante: O DELETE sem uma cláusula WHERE remove todos os registros da tabela. Sempre use WHERE para especificar quais registros devem ser removidos. \*\***

## Comando DELETE

- O funcionário 'Carlos Pereira' foi desligado da empresa e, a partir desse momento não deve mais constar nos registros da empresa.

## Comando DELETE

- O funcionário 'Carlos Pereira' foi desligado da empresa e, a partir desse momento não deve mais constar nos registros da empresa.

```
DELETE FROM funcionarios  
WHERE nome = 'Carlos Pereira';
```

## Comando DELETE

- A empresa também está desligando todos os funcionários que recebem menor que R\$ 3500,00.
- Dica:
  - Use o comando SELECT para verificar se a condição do WHERE está correta e, também, para verificar os dados do registro.
  - Execute o comando DELETE copiando a cláusula WHERE do comando SELECT;
  - Repita o comando SELECT para verificar o êxito do comando UPDATE.



## Comando DELETE

- A empresa também está desligando todos os funcionários que recebem menor que R\$ 3500,00.

```
SELECT * FROM funcionarios WHERE salario < 3500.00;
```

```
DELETE FROM funcionarios  
WHERE salario < 3500.00;
```

```
SELECT * FROM funcionarios WHERE salario < 3500.00;
```

## Atividade Prática

- 1)Atualize o salário de 'Maria Oliveira' para 6500.00 e altere seu cargo para 'Diretora'.
- 2)Remova todos os funcionários cujo cargo seja 'Analista'.
- 3)Adicione um novo funcionário chamado 'Juliana Lima' com cargo 'Assistente' e salário 2900.00.
- 4)Aumente o salário de todos os funcionários em 5%.

## Atividade Prática

1) Atualize o salário de 'Maria Oliveira' para 6500.00 e altere seu cargo para 'Diretora'.

```
UPDATE funcionarios
```

```
    set salario = 6500.00
```

```
    , cargo = 'Diretora'
```

```
WHERE nome = 'Maria Oliveira';
```

2) Remova todos os funcionários cujo cargo seja 'Analista'.

```
DELETE
```

```
    FROM funcionarios
```

```
WHERE cargo = 'Analista';
```

## Atividade Prática

- 3) Adicione um novo funcionário chamado 'Juliana Lima' com cargo 'Assistente' e salário 2900.00.

```
INSERT INTO funcionarios (nome, cargo, salario)
VALUES ('Juliana Lima', 'Assistente', 2900.00);
```

- 4) Aumente o salário de todos os funcionários em 5%.

```
UPDATE funcionarios
SET salario = salario * 1.05;
```

## Atividade Prática

Você foi contratado para criar um banco de dados para uma biblioteca local. O objetivo é gerenciar informações sobre livros, autores e empréstimos de livros. O banco de dados deve conter as seguintes tabelas:

- Autores - Para armazenar informações sobre os autores.
- Livros - Para armazenar informações sobre os livros.
- Empréstimos - Para armazenar informações sobre os empréstimos de livros.

## Atividade Prática – Estrutura das Tabelas

- Autores
  - autor\_id (INT, PRIMARY KEY, AUTO\_INCREMENT)
  - nome (VARCHAR(100))
  - nacionalidade (VARCHAR(50))
- Livros
  - livro\_id (INT, PRIMARY KEY, AUTO\_INCREMENT)
  - titulo (VARCHAR(150))
  - autor\_id (INT, FOREIGN KEY que referencia autor\_id na tabela Autores)
  - ano\_publicacao (YEAR)
- Empréstimos
  - emprestimo\_id (INT, PRIMARY KEY, AUTO\_INCREMENT)
  - livro\_id (INT, FOREIGN KEY que referencia livro\_id na tabela Livros)
  - data\_emprestimo (DATE)
  - data\_devolucao (DATE)

## Atividade Prática – Estrutura das Tabelas

### 1) Criação do Banco de Dados e Tabelas:

- 1) Crie um banco de dados chamado biblioteca.
- 2) Crie a tabela Autores com os campos e tipos de dados descritos acima.
- 3) Crie a tabela Livros com os campos e tipos de dados descritos acima, garantindo a relação com a tabela Autores através da chave estrangeira.
- 4) Crie a tabela Empréstimos com os campos e tipos de dados descritos acima, garantindo a relação com a tabela Livros através da chave estrangeira.

## Atividade Prática – Estrutura das Tabelas

### 2) Inserção de Dados:

- 1) Insira um autor chamado "J.K. Rowling" com a nacionalidade "Britânica" na tabela Autores.
- 2) Insira um livro chamado "Harry Potter e a Pedra Filosofal", publicado em 1997 e escrito por "J.K. Rowling" na tabela Livros.
- 3) Insira um empréstimo do livro "Harry Potter e a Pedra Filosofal" com data de empréstimo 2024-09-01 e data de devolução 2024-09-15 na tabela Empréstimos.



## Atividade Prática – Estrutura das Tabelas

### 3) Consulta de Dados:

- 1) Selecione todos os livros da tabela Livros.
- 2) Selecione todos os autores da tabela Autores.
- 3) Selecione todos os empréstimos da tabela Empréstimos.

## Atividade Prática – Estrutura das Tabelas

### 4) Atualização de Dados:

- 1) Atualize o nome do autor "J.K. Rowling" para "Joanne Rowling" na tabela Autores.
- 2) Atualize a data de devolução do empréstimo do livro "Harry Potter e a Pedra Filosofal" para 2024-09-20 na tabela Empréstimos.

## Atividade Prática – Estrutura das Tabelas

### 5) Exclusão de Dados:

- 1) Exclua o empréstimo registrado na tabela Empréstimos que foi feito para o livro "Harry Potter e a Pedra Filosofal".
- 2) Exclua o livro "Harry Potter e a Pedra Filosofal" da tabela Livros.
- 3) Exclua o autor "Joanne Rowling" da tabela Autores.

Correção na próxima aula

- Assuntos da próxima aula:
  - Operadores lógicos
  - Equações de junção