

# Fundamentos de Modelagem Relacional e SQL

*Professor: Bernardo Petry Prates*



**Kaike Torres da silva**

30.11.2024

Análise e Desenvolvimento de Sistemas (ADS)

## INTRODUÇÃO

Este relatório documenta a execução dos exercícios de SQL, abordando criação de tabelas,

inserção e modificação de dados,

bem como uma reflexão sobre os desafios enfrentados e soluções aplicadas durante o processo.

## RESULTADOS

### 1. Criação de Tabelas e Relacionamentos:

As tabelas 'Livros', 'Clientes' e 'Pedidos' foram criadas, incluindo as definições de colunas, tipos de dados e relacionamentos entre elas.

```
-- Criação de Tabelas e Relacionamentos

CREATE TABLE Livros (
    livro_id INT PRIMARY KEY AUTO_INCREMENT,
    titulo VARCHAR(200) NOT NULL,
    autor VARCHAR(150) NOT NULL,
    preco DECIMAL(10, 2) NOT NULL,
    estoque INT DEFAULT 0
);

CREATE TABLE Clientes (
    cliente_id INT PRIMARY KEY AUTO_INCREMENT,
    nome VARCHAR(100) NOT NULL,
    email VARCHAR(100) UNIQUE NOT NULL,
    telefone VARCHAR(20),
    endereco VARCHAR(200)
);

CREATE TABLE Pedidos (
    pedido_id INT PRIMARY KEY AUTO_INCREMENT,
    cliente_id INT NOT NULL,
    data_pedido DATE NOT NULL,
    valor_total DECIMAL(10, 2),
    FOREIGN KEY (cliente_id) REFERENCES Clientes(cliente_id)
);
```

## 2. Inserção e Modificação de Dados:

Diversas operações foram realizadas, como inserção de registros, alteração da estrutura das tabelas (ex.: adição de colunas) e modificação de dados utilizando comandos como UPDATE e DELETE.

```
INSERT INTO Livros (titulo, autor, preco, estoque) VALUES
('O Senhor dos Anéis', 'J.R.R. Tolkien', 59.90, 10),
('1984', 'George Orwell', 39.90, 15),
('Dom Quixote', 'Miguel de Cervantes', 49.90, 8),
('A Revolução dos Bichos', 'George Orwell', 29.90, 20),
('O Pequeno Príncipe', 'Antoine de Saint-Exupéry', 19.90, 25);

INSERT INTO Clientes (nome, email, telefone, endereco) VALUES
('João Silva', 'joao@example.com', '9999-8888', 'Rua A, 123'),
('Maria Souza', 'maria@example.com', '8888-7777', 'Rua B, 456'),
('Carlos Oliveira', 'carlos@example.com', '7777-6666', 'Rua C, 789'),
('Ana Lima', 'ana@example.com', '6666-5555', 'Rua D, 101'),
('Paula Ribeiro', 'paula@example.com', '5555-4444', 'Rua E, 202');

INSERT INTO Pedidos (cliente_id, data_pedido, valor_total) VALUES
(1, '2024-11-01', 150.00),
(2, '2024-11-02', 300.00),
(3, '2024-11-03', 100.00),
(4, '2024-11-04', 50.00),
(5, '2024-11-05', 250.00);
```

## 3. Reflexão:

Os principais desafios incluíram conflitos de chave estrangeira, dificuldades ao alterar tipos de dados e problemas de desempenho. As diferenças entre comandos como DELETE e DROP TABLE também foram exploradas.

-- Reflexão sobre o Processo de Implementação

-- Desafios enfrentados e soluções:

### **-- 1. Conflitos de chave estrangeira:**

-- Ao tentar excluir registros em 'Clientes', houve um erro devido a chaves estrangeiras em 'Pedidos'.

-- Para resolver, foi necessário excluir os pedidos associados primeiro ou usar 'ON DELETE CASCADE'.

### **-- 2. Alteração de tipos de dados:**

-- Alterar o tipo da coluna 'preco' gerou conflito com os dados existentes. A solução foi garantir que os

-- dados fossem compatíveis antes de alterar.

### **-- 3. Performance em grandes volumes:**

-- Consultas com muitas junções podem ser lentas. Índices em colunas de chaves estrangeiras e de pesquisa

- ajudam a otimizar.

### **-- Diferenças entre comandos:**

#### **-- 1. DELETE vs DROP TABLE:**

-- DELETE remove registros específicos de uma tabela, mas mantém sua estrutura.

-- DROP TABLE remove a tabela inteira, incluindo sua estrutura e dados.

#### **-- 2. ALTER TABLE vs UPDATE:**

-- ALTER TABLE modifica a estrutura da tabela (ex.: adiciona/remova colunas, altera tipos de dados).

-- UPDATE modifica os dados existentes dentro da tabela.