

ATIVIDADE ATIVA DE BANCO DE DADOS SQL E NO SQL

✓ **Objetivo**

- Compreender diferenças entre bancos de dados relacionais (SQL) e não relacionais (NoSQL).
- Praticar consultas, inserções e manipulação de dados em ambos os modelos.
- Analisar vantagens e desvantagens em cenários reais.

✓ **Contexto**

Imagine que você é um desenvolvedor contratado por uma empresa de e-commerce que precisa armazenar dados de clientes e pedidos.

O gestor pede para você testar dois modelos de banco:

- Relacional (MySQL/PostgreSQL)
- NoSQL (MongoDB)

Você deverá criar estruturas básicas, inserir dados e realizar consultas.

Parte 1 – Banco Relacional (SQL)

1. Criação das tabelas

```
CREATE DATABASE ecommerce;  
USE ecommerce;
```

```
CREATE TABLE clientes (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(100),  
    email VARCHAR(100),  
    cidade VARCHAR(50)  
);
```

```
CREATE TABLE pedidos (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    cliente_id INT,  
    produto VARCHAR(100),  
    valor DECIMAL(10,2),  
    FOREIGN KEY (cliente_id) REFERENCES clientes(id)  
);
```

2. Inserção de dados

```
INSERT INTO clientes (nome, email, cidade) VALUES
('Ana Souza', 'ana@email.com', 'Brasília'),
('Carlos Lima', 'carlos@email.com', 'São Paulo'),
('Mariana Torres', 'mariana@email.com', 'Rio');
```

```
INSERT INTO pedidos (cliente_id, produto, valor) VALUES
(1, 'Notebook', 3500.00),
(1, 'Mouse', 150.00),
(2, 'Celular', 2500.00),
(3, 'Fone de Ouvido', 200.00);
```

3. Consultas

-- Listar todos os clientes

```
SELECT * FROM clientes;
```

-- Listar pedidos e o nome do cliente

```
SELECT p.id, c.nome, p.produto, p.valor
FROM pedidos p
JOIN clientes c ON p.cliente_id = c.id;
```

-- Total gasto por cliente

```
SELECT c.nome, SUM(p.valor) AS total_gasto
FROM pedidos p
JOIN clientes c ON p.cliente_id = c.id
GROUP BY c.nome;
```

Parte 2 – Banco Não Relacional (NoSQL - MongoDB)

1. Inserção de documentos

```
use ecommerce;
```

```
db.clientes.insertMany([
  { _id: 1, nome: "Ana Souza", email: "ana@email.com", cidade: "Brasília" },
  { _id: 2, nome: "Carlos Lima", email: "carlos@email.com", cidade: "São Paulo" },
  { _id: 3, nome: "Mariana Torres", email: "mariana@email.com", cidade: "Rio" }
]);
```

```
db.pedidos.insertMany([
  { cliente_id: 1, produto: "Notebook", valor: 3500.00 },
  { cliente_id: 1, produto: "Mouse", valor: 150.00 },
  { cliente_id: 2, produto: "Celular", valor: 2500.00 },
```

```
{ cliente_id: 3, produto: "Fone de Ouvido", valor: 200.00 }  
]);
```

2. Consultas

```
// Listar todos os clientes  
db.clientes.find().pretty();
```

```
// Listar todos os pedidos  
db.pedidos.find().pretty();
```

```
// Pedidos de um cliente específico  
db.pedidos.find({ cliente_id: 1 });
```

```
// Agregação: total gasto por cliente  
db.pedidos.aggregate([  
  { $group: { _id: "$cliente_id", total: { $sum: "$valor" } } }  
]);
```

Execução:

1. Execute os scripts acima em MySQL/PostgreSQL e MongoDB.
2. Insira dois novos clientes e seus pedidos em cada modelo de banco.
3. Crie uma consulta que mostre:
 - Clientes da cidade de São Paulo (SQL e NoSQL).
 - O cliente que mais gastou (SQL e NoSQL).
4. Compare:
 - Qual banco foi mais fácil de usar?
 - Qual estrutura você acha mais flexível para um e-commerce?
 - Em que situações o modelo relacional é mais indicado? E o não relacional?

Entrega do relatório:

- Print das consultas SQL e NoSQL executadas.
- Explicação dos prints.
- Pequeno relatório (3 a 10 linhas) respondendo às perguntas de comparação.
- Atividade realizada por IA será zerada.
- Será aceito apenas arquivos em PDF ou DOCs.
- Não será analisado apenas imagem, precisa ter explicação.
- AA deve ser estruturada em formato de relatório, seguindo formatação da ABNT ou IEEE.