

# Diagrama Entidade-Relacionamento (DER) – Sistema de Clientes

O Diagrama Entidade-Relacionamento (DER) é um modelo conceitual que permite visualizar a estrutura lógica dos dados de uma aplicação. Ele mostra as **entidades** (que representam objetos ou conceitos, como tabelas em um banco de dados) e os **atributos** (as informações associadas a essas entidades).

## 📌 Entidade Principal: Cliente

Neste sistema, trabalhamos com uma única entidade chamada **Cliente**, responsável por armazenar os dados cadastrais de cada pessoa registrada.

### Atributos da Entidade Cliente:

Atributo	Tipo de Dado	Restrições	Descrição
id	INTEGER	Chave Primária, Auto-Incremento, Not Null	Identificador único de cada cliente.
nome	TEXT	Not Null	Nome próprio do cliente.
sobrenome	TEXT	Not Null	Sobrenome do cliente.
email	TEXT	Not Null	Endereço de e-mail.
cpf	TEXT	Not Null	Número do CPF (armazenado como texto).

💡 **Nota Técnica:** O campo `cpf` é armazenado como texto para preservar zeros à esquerda, o que não seria possível se usássemos o tipo numérico.

---

## 🔗 Relacionamentos

- O sistema atual não possui múltiplas entidades, portanto **não há relacionamentos** definidos entre tabelas.
- Em um cenário mais completo, seria possível adicionar entidades como "Compras", "Endereços" ou "Atendimentos", relacionadas ao cliente por meio de chaves estrangeiras (*foreign keys*).

---

## 📦 Representação Visual (em texto)

Aqui está uma forma simples de representar essa estrutura usando um quadro textual:

```
pgsql
CopiarEditar
+-----+
```

Cliente
id (PK, INTEGER)
nome (TEXT, NOT NULL)
sobrenome (TEXT)
email (TEXT)
cpf (TEXT)

- **PK** = Primary Key (Chave Primária).
- Todos os campos, exceto o `id` (gerado automaticamente), devem ser preenchidos.



## Considerações para Estudantes ou Iniciantes

- Cada cliente precisa ter um `id` único para evitar duplicações e facilitar operações como atualização ou exclusão.
- Mesmo que dois clientes compartilhem o mesmo nome ou sobrenome, o sistema os distingue pelo `id`.
- Este é um modelo **simples e linear**, ideal para iniciar o aprendizado sobre banco de dados e relacionamento de dados.
- Com o tempo, o banco pode ser expandido com mais entidades e relações, aplicando o conceito de **normalização**.