



Modelo Entidade-Relacionamento (MER) — *Gerenciador de Clientes*

O **Modelo Entidade-Relacionamento (MER)** é uma representação conceitual da estrutura de um banco de dados. Ele descreve os **componentes principais** do banco — como **entidades**, **atributos** e, quando aplicável, **relacionamentos** — de forma visual e lógica. Pode ser entendido como um "mapa" que mostra como os dados estão organizados e interconectados.



Entidades e Atributos

Nesta aplicação, temos apenas uma entidade principal: **Clientes**. Ela representa os registros de pessoas cadastradas no sistema.

Entidade: Clientes

- **Descrição:** Representa cada cliente registrado no banco de dados.
- **Atributos:**

Atributo	Tipo de Dado	Restrições	Descrição
id	INTEGER	PRIMARY KEY, NOT NULL, AUTOINCREMENT	Identificador único do cliente
nome	TEXT	NOT NULL	Nome do cliente
sobrenome	TEXT	NOT NULL	Sobrenome do cliente
email	TEXT	NOT NULL	Endereço de e-mail do cliente
cpf	TEXT	NOT NULL	CPF do cliente (armazenado como texto)

Nota: O tipo `TEXT` é utilizado para o CPF para preservar possíveis zeros à esquerda e permitir melhor validação de formato.

Relacionamentos

- **Situação atual:** Esta versão do sistema não possui relacionamentos, pois contém apenas uma tabela (Clientes).
- **Possibilidades futuras:** Em sistemas mais completos, você poderia criar entidades adicionais, como:
 - **Endereços** (relacionados a `Clientes` por meio de uma chave estrangeira `cliente_id`)
 - **Pedidos, Contatos, Histórico de Compras**, etc.

Representação Visual (Descrição Textual)

Visualize a estrutura da entidade "Clientes" como uma tabela com colunas representando os atributos:

```
pgsql
CopiarEditar
+-----+
|          Clientes          |
+-----+
| id      (PK, INTEGER) |
| nome    (TEXT)        |
| sobrenome (TEXT)       |
| email   (TEXT)        |
| cpf     (TEXT)        |
+-----+
```

- **PK = Primary Key (Chave Primária):** garante que cada cliente tenha um identificador único.
- Todos os atributos são **obrigatórios** (NOT NULL).
- O campo `id` é **gerado automaticamente** pelo banco de dados com incremento sequencial.

Observações para Iniciantes

- A **chave primária (id)** é essencial para garantir unicidade entre os registros e para permitir futuras relações com outras entidades.
- **TEXT é preferido ao INTEGER para o CPF**, pois:
 - Evita a perda de zeros à esquerda.
 - Permite tratar o campo como uma string, facilitando formatação e validação.
- Esse modelo é **simples e direto**, ideal para iniciantes entenderem como se estruturam dados relacionais.
- Em projetos maiores, o MER se torna um recurso fundamental para planejar corretamente os relacionamentos entre tabelas e garantir a integridade dos dados.

