

Modelo Entidade-Relacionamento (MER) - Gerenciador de Clientes

Pense no Modelo Entidade-Relacionamento, ou MER, como um **mapa detalhado do seu banco de dados**. Ele nos ajuda a visualizar as peças principais – as **entidades** (que são como tabelas, onde guardamos nossos dados) – e as **características** de cada peça, que chamamos de **atributos** (as colunas dentro dessas tabelas). Para quem está começando, é uma maneira fantástica de entender a estrutura por trás de um sistema.

Entidades e Seus Detalhes

Neste projeto simples, mas eficaz, temos foco em apenas uma "peça" principal, uma única entidade para manter tudo organizado: **Clientes**.

Entidade: Clientes

- **O que ela representa?** Basicamente, essa entidade é como sua **ficha cadastral digital** de cada pessoa. Ela guarda todas as informações essenciais dos clientes no banco de dados.
 - **Quais informações ela armazena? (Atributos)**
 - **id:** Imagine isso como o **número de matrícula único** de cada cliente. É um código exclusivo que o sistema gera automaticamente para cada novo cliente, garantindo que ninguém se confunda.
 - **Tipo:** É um número inteiro (INTEGER).
 - **Regras:** É a **chave primária (PK)** – o identificador principal –, nunca pode estar vazio e o próprio sistema cuida de gerar o próximo número.
 - **nome:** O **primeiro nome** do seu cliente.
 - **Tipo:** Texto (TEXT).
 - **Regras:** Precisa ser preenchido (não nulo).
 - **sobrenome:** A **parte final do nome** do seu cliente.
 - **Tipo:** Texto (TEXT).
 - **Regras:** Também precisa ser preenchido (não nulo).
 - **email:** O **endereço de e-mail** do cliente, para você poder se comunicar com ele.
 - **Tipo:** Texto (TEXT).
 - **Regras:** Não pode ficar vazio.
 - **cpf:** O **número do Cadastro de Pessoas Físicas** do cliente.
 - **Tipo:** Texto (TEXT).
 - **Regras:** É uma informação obrigatória (não nulo).
-

Conexões (Relacionamentos)

- Neste "Gerenciador de Clientes", como só temos uma tabela (`clientes`), **não há relacionamentos** entre diferentes entidades. É um sistema bem direto ao ponto.
- Em sistemas maiores, porém, você veria outras "caixas" (entidades) se conectando à "Clientes". Por exemplo, uma entidade "Pedidos" se ligaria a "Clientes" para saber qual cliente fez qual pedido, usando o `id` do cliente como uma "ponte" (chave estrangeira).

Olhando o Mapa (Representação Gráfica Textual)

Para ter uma ideia visual de como a entidade "Clientes" se parece nesse mapa, pense nela como uma caixa, e dentro dela, listamos seus atributos:

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

- O **PK** que você vê ao lado do `id` é a abreviação para "Primary Key" (Chave Primária), que é a garantia de que cada cliente tem um código de identificação totalmente único.
- Todos esses campos (atributos) são obrigatórios, exceto o `id`, que o próprio sistema SQLite cuida de preencher automaticamente.

Uma Dica para Quem Está Começando

- O `id` é super importante! Mesmo que você tenha dois clientes chamados "Maria Silva", o `id` deles será diferente, evitando qualquer confusão.
- Usamos o tipo **TEXTO (TEXT)** para o `cpf` porque, no Brasil, o CPF pode começar com o número zero. Se usássemos um tipo numérico, esse zero inicial se perderia, e não é isso que queremos!

Este modelo é um ponto de partida simples e eficaz, mas serve como base para projetos muito mais complexos, onde você poderia adicionar novas tabelas e ligações entre elas!