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

O Modelo Entidade-Relacionamento (MER) é uma forma de representar a estrutura de um banco de dados, mostrando as \*\*entidades\*\* (como tabelas) e seus \*\*atributos\*\* (como colunas). Para iniciantes, pense no MER como um mapa que descreve como os dados estão organizados.

### Entidades e Atributos

Este projeto tem apenas uma entidade: \*\*Clientes\*\*. Abaixo está a descrição da entidade e seus atributos:

\*\*Entidade: Clientes\*\*

- \*\*Descrição\*\*: Representa as informações de clientes armazenadas no banco de dados.

- \*\*Atributos\*\*:

- \*\*id\*\*: Um número único que identifica cada cliente (chave primária, gerada automaticamente).

- Tipo: Inteiro (INTEGER).

- Propriedade: Chave primária, não nulo, auto-incrementado.

- \*\*nome\*\*: O nome do cliente.

- Tipo: Texto (TEXT).

- Propriedade: Não nulo.

- \*\*sobrenome\*\*: O sobrenome do cliente.

- Tipo: Texto (TEXT).

- Propriedade: Não nulo.

- \*\*email\*\*: O endereço de email do cliente.

- Tipo: Texto (TEXT).

- Propriedade: Não nulo.

- \*\*cpf\*\*: O CPF do cliente.

- Tipo: Texto (TEXT).

- Propriedade: Não nulo.

### Relacionamentos

- Não há relacionamentos entre entidades neste projeto, pois existe apenas uma tabela (`clientes`).

- Em projetos mais complexos, você poderia ter outras entidades (como "Pedidos" ou "Endereços") relacionadas à entidade "Clientes" por meio de chaves estrangeiras.

### Representação Gráfica (Descrição Textual)

Imagine uma caixa representando a entidade "Clientes", com os atributos listados dentro dela:

```

+-------------------+

| Clientes |

+-------------------+

| id (PK, INTEGER) |

| nome (TEXT) |

| sobrenome (TEXT) |

| email (TEXT) |

| cpf (TEXT) |

+-------------------+

```

- \*\*PK\*\* significa "Primary Key" (chave primária), que garante que cada cliente tenha um identificador único.

- Todos os atributos são obrigatórios (não nulo), exceto que o `id` é gerado automaticamente pelo SQLite.

### Observações para Iniciantes

- O `id` é importante porque ajuda a identificar cada cliente de forma única, mesmo que dois clientes tenham o mesmo nome.

- Usamos TEXT para `cpf` (em vez de número) porque o CPF pode começar com zero, e números inteiros não armazenam zeros à esquerda.

- Este é um modelo simples, mas em projetos reais, você pode adicionar mais tabelas e relacionamentos, como uma tabela de "Pedidos" vinculada ao `id` do cliente.