

1. Definição de Relacionamentos

Relacionamento entre tabelas é a forma como os dados em uma tabela estão ligados aos dados em outra. Esse vínculo é implementado usando **chaves primárias** (identificadores únicos em uma tabela) e **chaves estrangeiras** (colunas que referenciam chaves primárias de outras tabelas), garantindo a conexão entre registros relacionados.

2. Chave Primária

A chave primária é um campo ou conjunto de campos que identifica unicamente cada registro em uma tabela. Características principais:

- Deve ser única para cada registro;
 - Não pode ser nula (NOT NULL);
 - Usada para garantir a integridade dos dados e para relacionar tabelas.
Ela é essencial para garantir que cada linha seja individualmente identificável.
-

3. Chave Estrangeira

Chave estrangeira é uma coluna ou conjunto de colunas que cria um vínculo entre uma tabela e a chave primária de outra, estabelecendo um relacionamento. Sua importância está em garantir que os dados relacionados existam, permitindo integridade entre tabelas.

4. Integridade Referencial

Integridade referencial assegura que as referências entre tabelas sejam válidas, ou seja, que uma chave estrangeira aponte para um registro existente na tabela referenciada.

Exemplo prático: em uma tabela de **Pedidos**, o campo **cliente_id** (chave estrangeira) deve corresponder a um **id** existente na tabela **Clientes**. Se um cliente for excluído, o banco pode impedir a exclusão ou agir conforme regras definidas (excluir pedidos relacionados, etc.).

5. Tipos de Relacionamentos

- **1:1 (Um para Um):** Cada registro em uma tabela está relacionado a, no máximo, um registro em outra tabela.

Exemplo: Uma pessoa tem um único passaporte.

- **1:N (Um para Muitos):** Um registro em uma tabela pode estar relacionado a vários registros em outra.

Exemplo: Um cliente pode ter vários pedidos.

- **N:M (Muitos para Muitos):** Vários registros em uma tabela podem estar relacionados a vários registros em outra.

Exemplo: Estudantes podem se matricular em vários cursos, e cursos podem ter vários estudantes.

6. Exclusão e Atualização em Cascata

- **ON DELETE CASCADE:** Quando um registro pai é excluído, os registros filhos relacionados também são excluídos automaticamente.

- **ON UPDATE CASCADE:** Quando a chave primária de um registro pai é atualizada, os valores correspondentes nas chaves estrangeiras dos registros filhos também são atualizados automaticamente.

Use cascata quando desejar manter a integridade sem dados órfãos, especialmente em relacionamentos fortes.

7. Tabelas Intermediárias

Em relacionamentos N:M, uma tabela intermediária (ou tabela de junção) é usada para representar as associações entre as duas tabelas principais.

Sua estrutura inclui pelo menos duas colunas que são chaves estrangeiras referenciando as tabelas relacionadas.

Exemplo: Uma tabela **Matriculas** que liga **Estudantes** e **Cursos** no relacionamento muitos-para-muitos.

8. Vantagens dos Relacionamentos

Organizar dados em tabelas relacionadas:

- Evita redundância e inconsistência de dados;
- Facilita a manutenção e atualização;

- Melhora a integridade dos dados;
 - Permite consultas mais eficientes e complexas.
-

9. Problemas sem Relacionamentos

Sem relacionamentos adequados, podem surgir:

- Dados duplicados e inconsistentes;
 - Registros órfãos (referências inválidas);
 - Dificuldade em garantir a integridade e qualidade dos dados;
 - Complexidade maior na manutenção e consultas.
-

10. Aplicabilidade Prática

- **Relacionamento 1:N:** Em um sistema de e-commerce, um cliente pode fazer vários pedidos. Cada pedido pertence a um único cliente.
- **Relacionamento N:M:** Em uma gestão acadêmica, estudantes podem estar matriculados em vários cursos, e cada curso tem vários estudantes. Isso é representado por uma tabela intermediária de matrículas.