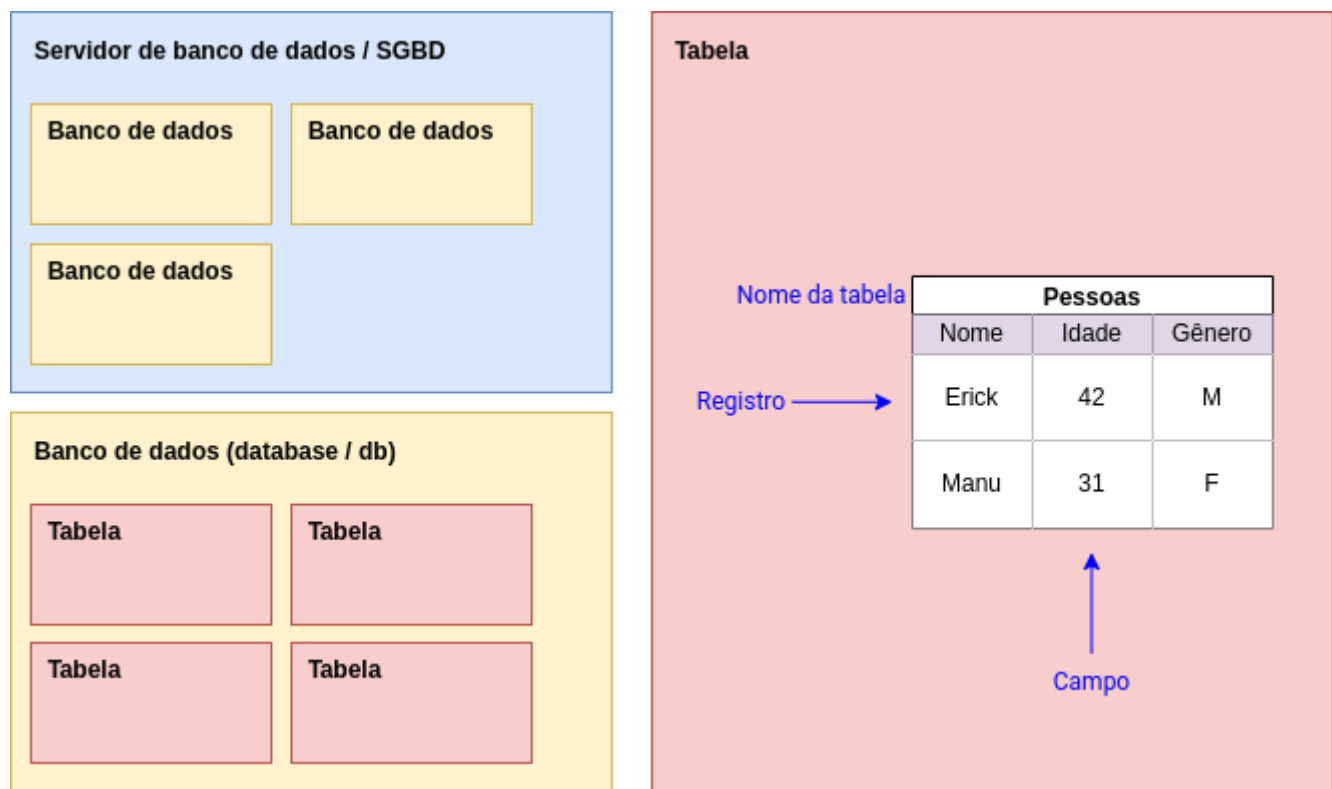


Revisão da aula anterior, um diagrama



Transformando uma ideia em um modelo através das análises

Quando criamos nossas listas de dados, e as estruturas que vão armazenar estes dados, usamos algumas regras para orientar os esforços.

Vamos falar mais sobre essas regras nas próximas aulas, mas podemos colocar algumas de forma simplificada agora:

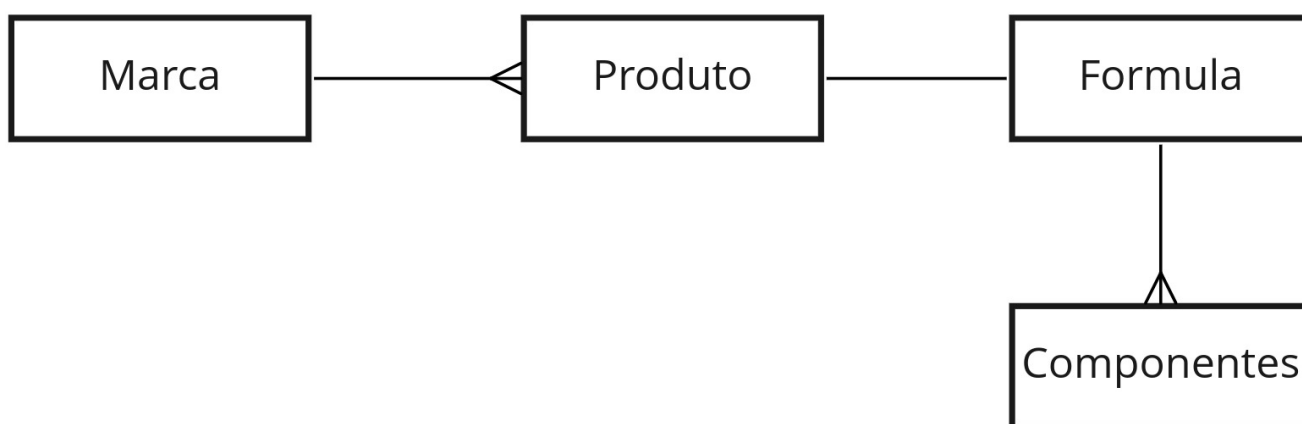
- Dados não devem se repetir
- Dados não devem ser ambíguos
- Dados devem ser armazenados de acordo com o uso.

Por meio dessas análises, vamos fazer a ponte entre a *análise* e a *modelagem* de dados através de uma ferramenta, o *Modelo Entidade-Relacionamento*, ou MER.

O modelo conceitual

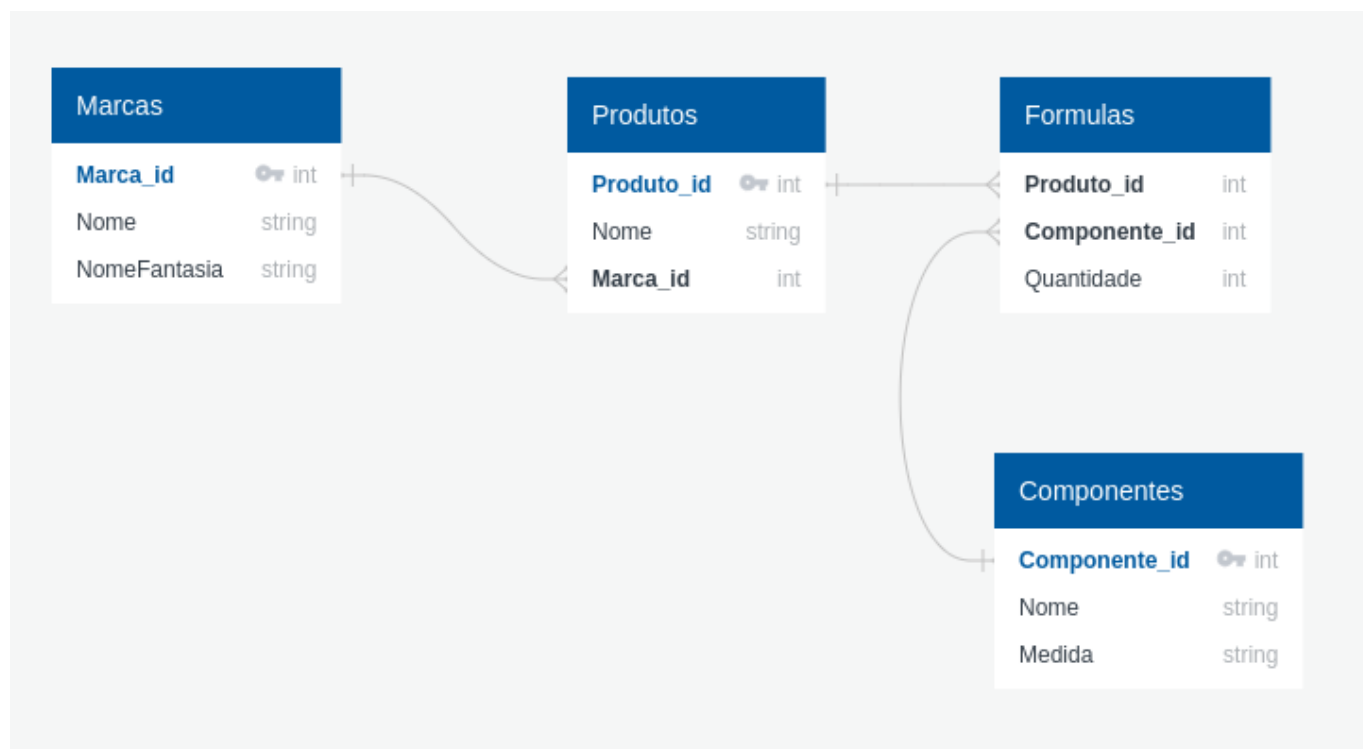
O modelo conceitual define, de forma mais geral, os dados que serão armazenados e seus relacionamentos. O objetivo é entender as principais "entidades", os grupos de dados que

precisamos armazenar, e como eles se relacionam.



O MER, ou o "diagrama do banco de dados"

O Modelo Entidade Relacionamento define, visualmente, as estruturas de armazenamento de dados e suas relações. Ele também é conhecido como modelo lógico, uma vez que ele é linguagem intermediária entre a análise de dados e o modelo físico, que são os comandos que alteram o banco de dados para que este espelhe o modelo físico.



Existem várias ferramentas para construir este tipo de desenho, as mais recomendadas:

modo texto

- <https://dbdiagram.io/home>
- <https://app.quickdatabasediagrams.com/#/>

modo desenho

- <https://www.lucidchart.com/pages/>
- <https://app.diagrams.net/>

Qual a relação entre SQL e o MER?

Nesse primeiro momento vamos conhecer como montar um Modelo Entidade Relacionamento usando desenhos e elementos comuns.

Mas com o tempo vamos perceber que a construção do MER segue as mesmas regras usadas na SQL:

- Organização de dados em tabelas
- Identificação dos campos e seus tipos
- Identificação das chaves

Por isso, o desenho correto do MER nos ajuda a usar melhor o banco de dados, uma vez que, se bem feito e mantido atualizado, ele serve como documentação.

Quais são os elementos do MER?

Entidade

Também chamada de tabela. Define um conjunto de dados da mesma natureza. Por exemplo, **contatos**, **pedidos**, **alunos**.

Nome

O nome de uma tabela deve ser inequívoca e única. Deve identificar os dados. É boa prática que os nomes de tabelas sejam no plural.

Campo (ou atributo)

Um campo define um valor do registro dentro da entidade.

Tipo de Campo

O tipo de campo define a natureza de valor que este campo armazena, como número, texto, data, booleano (true/false).

Relacionamento

Uma "linha" que define o relacionamento entre duas tabelas. Os relacionamentos são definidos de acordo com a cardinalidade, ou seja, a quantidade de dados que uma tabela se refere a outra.

- 1-1,
- 1-muitos (1-n)
- muitos-muitos (n-n)

E em alguns relacionamentos podemos definir que estes são **dependentes** ou não. Por exemplo, se vamos modelar um banco de dados de vendas a clientes, os dados de uma venda não podem existir sem que um usuário exista antes. Isso estabelece mais alguns tipos de relacionamento:

- 0,1-1
- 0,1-n

Chaves

Alguns campos extras podem servir para identificação unívoca de dados dentro de uma entidade. Por exemplo, em um cadastro de clientes brasileiros, o campo CPF pode ser usado para identificar uma pessoa, porque este dados não se repete para outras pessoas no Brasil. Este campo é chamado de **chave primária (ou primary key)**. As chaves primárias não são obrigatórias, mas recomendadas sempre que precisamos armazenar dados que devem ter uma identificação única na entidade. E podem ser compostas de mais de um campo.

Outras vezes, criamos campos que fazem referência a uma entidade externa àquela que estamos criando. Neste caso, criamos este campo com o mesmo tipo de dados que a chave primária da tabela externa, e em nossa tabela este campo é chamado de **chave estrangeira**.