

## 2 - Introdução

- O banco de dados é a base de um sistema, de forma que qualquer alteração em sua estrutura pode ser um processo bastante complexo. Portanto, deve-se tomar muito cuidado com a modelagem do banco de dados.
- Para modelar um banco de dados, deve-se analisar atentamente os requisitos do cliente, e garantir que existirão tabelas capazes de armazenar todas as informações necessárias para atender os requisitos coletados.
- Em se tratando de banco de dados, existem três níveis de modelagem:
  1. Modelagem Conceitual
  2. Modelagem Lógica
  3. Modelagem Física

## 2 - Introdução

- A modelagem do banco de dados se dá por meio de diagramas, que vão evoluindo do modelo lógico (mais simples) até o modelo físico.
- Existem inúmeras ferramentas de modelagem de banco de dados, sendo as mais conhecidas o Erwin, Workbench, DBDesign, BRModelo, etc.
- A parte mais complexa da modelagem é criar as tabelas com seus respectivos campos e relacionamentos de forma que possam atender todos os requisitos do cliente.
- Após os diagramas estarem prontos, o processo de implementação é muito simples, basta ter conhecimento da linguagem SQL. Algumas ferramentas geram automaticamente o código necessário.

## 2.1 - Modelagem Conceitual

- A modelagem conceitual é a forma mais simples de modelagem, sendo extremamente abstrata. Seu objetivo é mostrar um esquema geral de como ficará o banco de dados, servindo de base para o modelo lógico.
- Alguns estudiosos afirmam que o modelo conceitual não precisa ser feito, com a experiência pode-se elaborar o modelo lógico ou físico diretamente.
- Entretanto, para situações mais complexas é importante que os três níveis de modelagem sejam realizados.

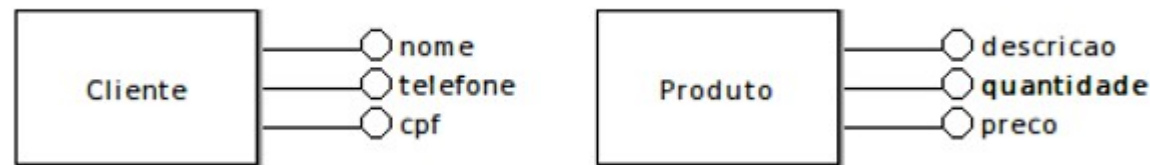
## 2.1.1 - Entidade

- Uma entidade é o conceito fundamental de um banco de dados relacional, pois ela representa um conjunto de objetos da realidade modelada. Futuramente, as entidades se tornarão tabelas.
- As entidades são representadas por um retângulo com o nome dentro. Como exemplo, em uma modelagem de uma biblioteca, existirão entidades aluno, livro, autor, empréstimo, etc. A figura abaixo mostra como são representadas as entidades:



## 2.1.2 - Atributos

- Os atributos são as especificações da entidade, ou seja, são os campos de uma tabela do banco de dados. Por exemplo, uma entidade cliente pode ter como atributos o nome, CPF, endereço e telefone.
- Graficamente, os atributos são representados por traços com um círculo em sua extremidade, seguido pelo nome do atributo. Caso uma entidade tenha muitos atributos, pode-se utilizar um **dicionário de dados** a parte para inserção dos mesmos. A figura abaixo ilustra um exemplo de representação de atributo:



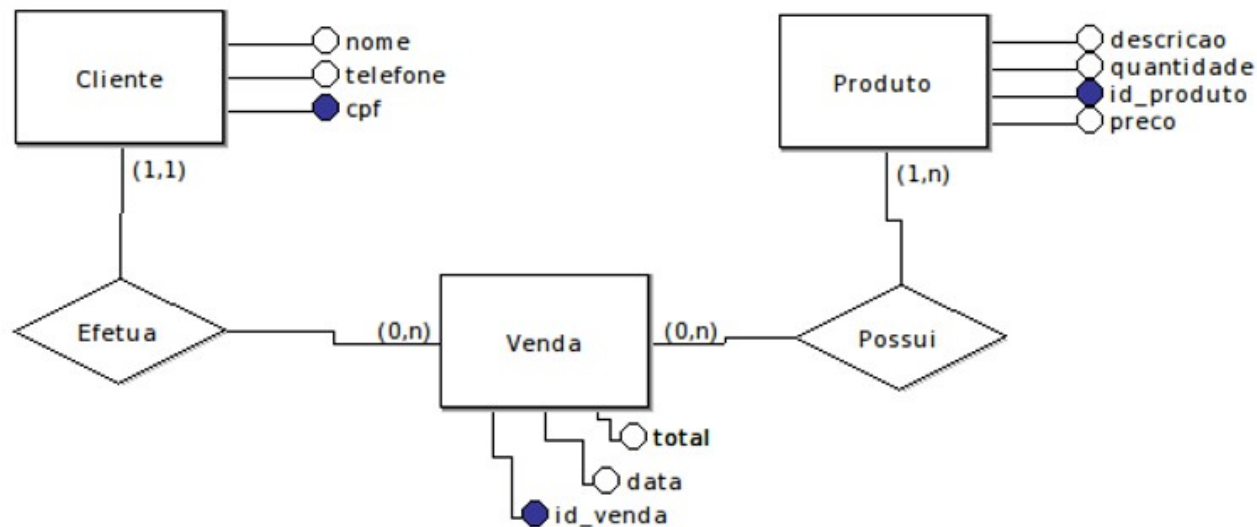
## 2.1.3 - Atributos Identificadores

- Toda entidade deve possuir um identificador, que pode ser um ou mais atributos, cujos valores servem para distinguir uma ocorrência da entidade das demais ocorrências da mesma entidade.
- Por exemplo, em uma entidade cliente, o identificador pode ser o atributo CPF, visto que cada cliente possui um número de CPF único. Ou pode ser criado um campo para ser identificador, como id\_produto. Esse atributo identificador é mais conhecido como **chave primária**.
- O atributo nome não poderia ser identificador, pois pode existir mais de um cliente com o mesmo nome. O atributo identificador possui o círculo preenchido:



## 2.1.4 - Relacionamento

- As entidades na maioria das vezes precisam compartilhar dados entre si, sendo que para isso é necessário criar um relacionamento entre elas. Por exemplo, um entidade pedido precisa de dados vindos da entidade cliente.
- O relacionamento permite cruzar dados entre as tabelas, permitindo buscas de dados, consultas e relatórios mais elaborados e com mais informações. Graficamente, o relacionamento é representado por um alinha ligando as duas entidades, com um losango descrevendo o relacionamento, como mostra a figura:



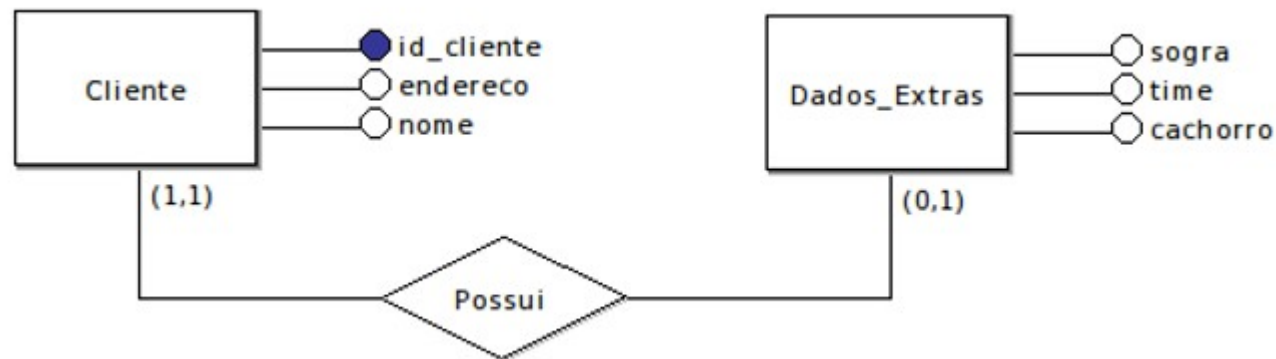
## 2.1.5 - Cardinalidade

- Uma propriedade importante na modelagem de um banco de dados é a cardinalidade da entidade, ou seja, quantas ocorrências de uma entidade podem estar associadas a outra entidade.
- Existem três possibilidades de cardinalidade, são elas: Um pra Um (1:1), Um pra Muitos (1:N) e Muitos pra Muitos (N:N).
- Também existe a possibilidade de informar a cardinalidade em mínima e máxima, sendo apenas a máxima obrigatória.



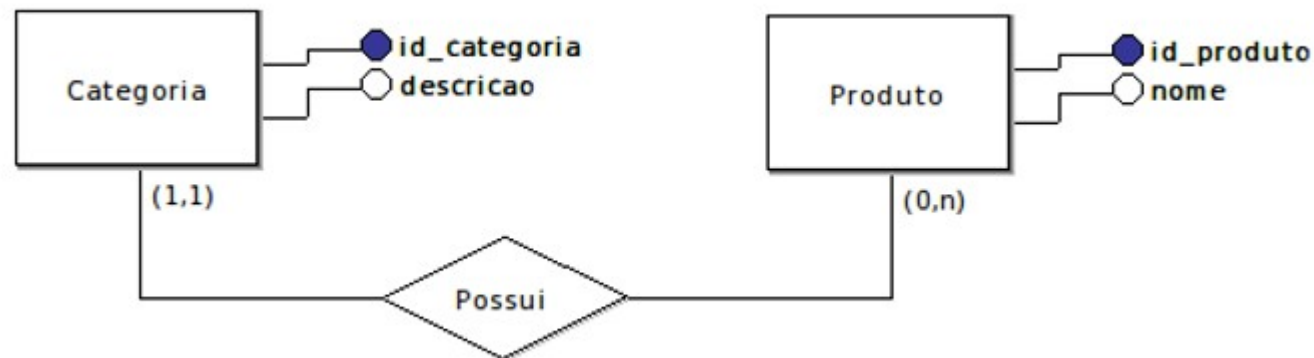
## 2.1.5 - Cardinalidade

- **Um pra Um (1:1):** É o tipo menos usado de cardinalidade, pois na maioria das vezes que duas entidades possuem (1:1), é possível unir as duas entidades em uma só. Neste caso, para uma ocorrência de uma entidade, haverá apenas uma ocorrência na outra entidade relacionada.
- Como exemplo, pode-se analisar a figura abaixo, onde existe uma entidade cliente com o objetivo de cadastrar os dados básicos, mas eventualmente pode ser necessário dados extras, que ficam armazenados na entidade Dados\_Extras:



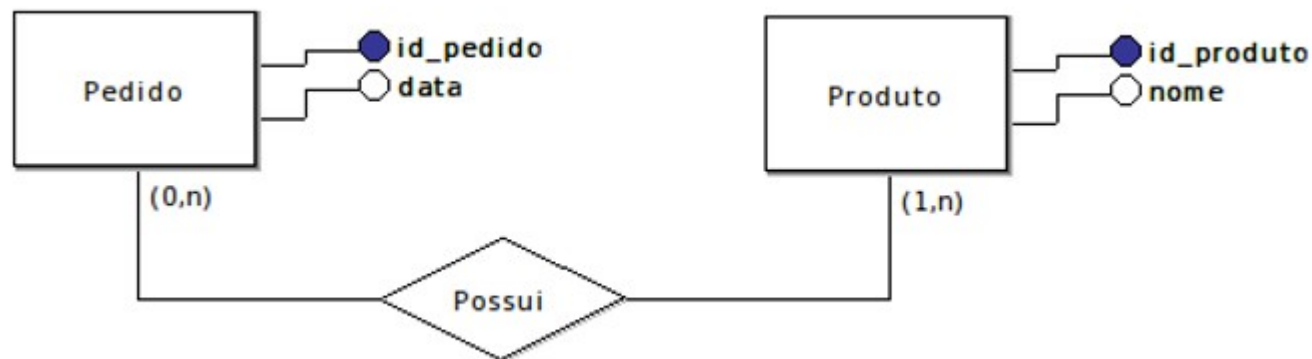
## 2.1.5 - Cardinalidade

- **Um pra Muitos (1:N):** É o tipo mais comum de cardinalidade, sendo que o ideal é deixar todos os relacionamentos com esta cardinalidade. Neste caso, uma ocorrência de uma entidade pode estar associado a várias ocorrências de outra entidade.
- No exemplo abaixo, uma categoria de produto pode estar associada a diversos produtos (a categoria celular pode estar associada aos produtos galaxy, iphone, moto g, etc.).



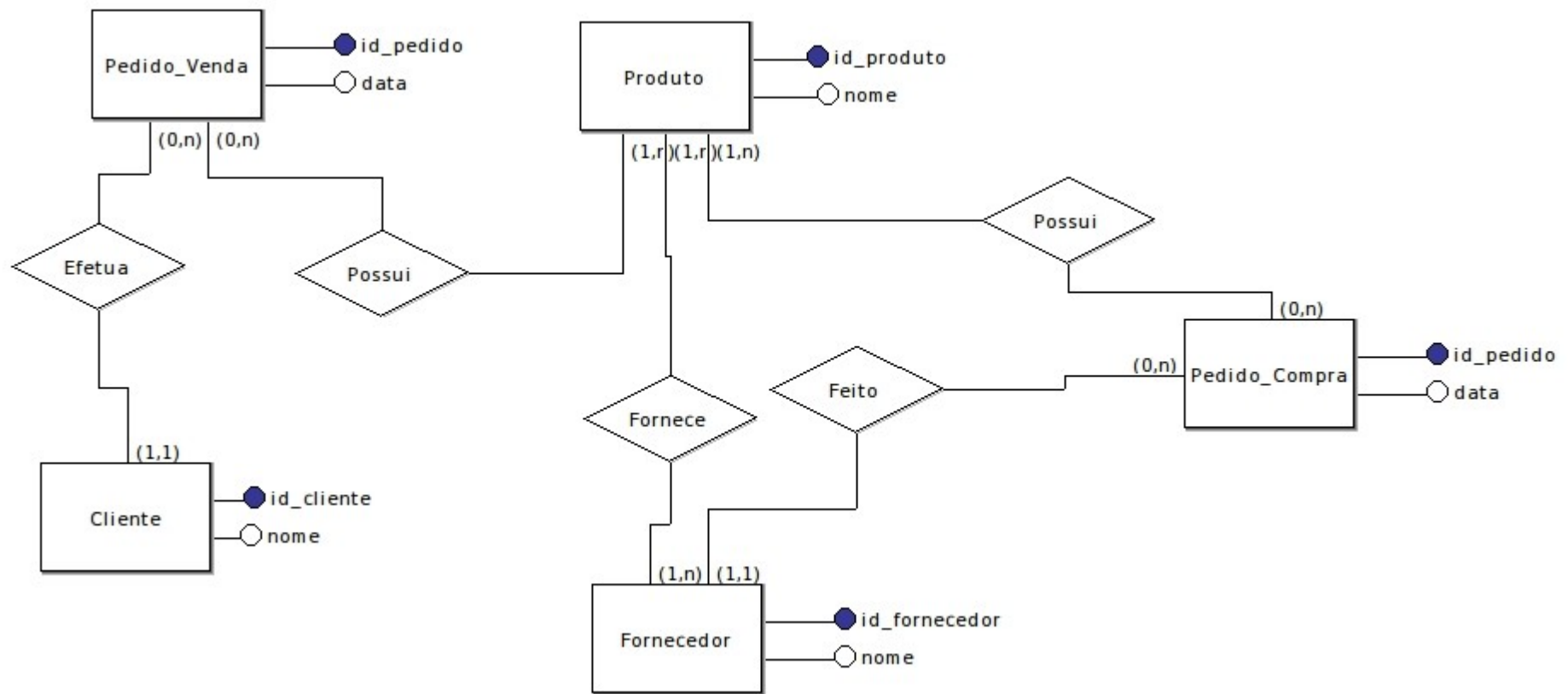
## 2.1.5 - Cardinalidade

- **Muitos pra Muitos (N:N):** É o caso mais complexo de cardinalidade, pois não tem como ser implementado no banco de dados, e deverá ser transformado em (1:N). Mas no modelo conceitual, pode ser representado como (N:N) mesmo.
- O exemplo abaixo mostra um exemplo onde um pedido pode ter vários produtos, mas um produto pode estar em vários pedidos:



## 2.1.6 - Exemplo

- Elabore um modelo conceitual para um comércio de vendas em geral, onde se deve controlar: Clientes, fornecedores, vendas, compras, estoque:



# Exercícios

- Elabore um modelo conceitual de uma locadora, onde se deve controlar os filmes, atores dos filmes, cliente, categoria do filme e locação.