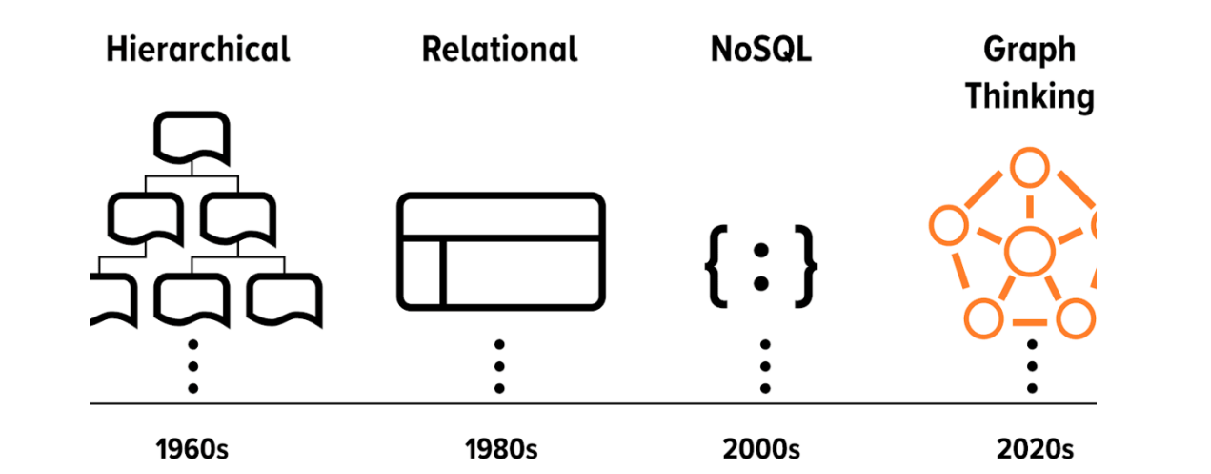
Introdução a Arquitetura de Bancos de Dados



**Dados Estruturados: Banco De Dados**

**Dados Semi Estruturados: XML**

**Dados Não Estruturados: Documentos**

SQL x NoSQL

- Declarativa, padrão

- Relacional

- Sintaxe Amigável, Eficiência de Recursos

- Dimensionamento Vertical Caro

- Possível deixar lento

SGBD

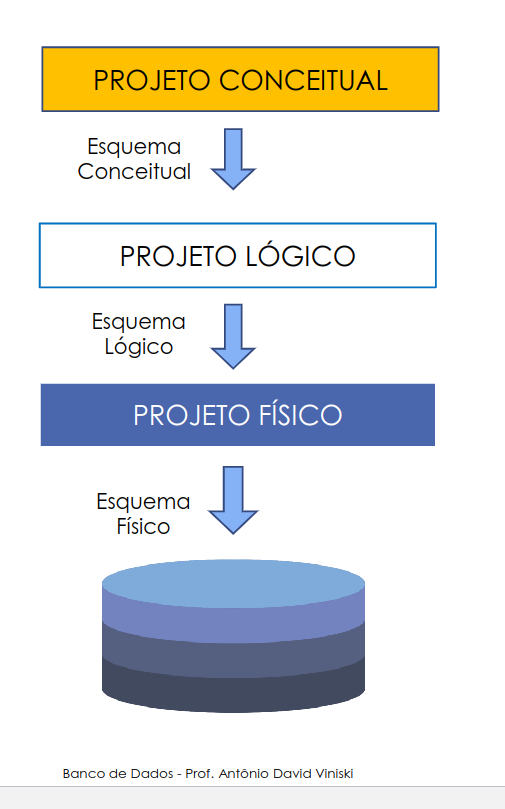
**Sistema de gerenciamento de banco de dados**

* O usuário pode adicionar, remover, alterar arquivos do banco de dados.
* Inserir, alterar ou recuperar dados
* Controle de acesso
* Armazenamento Persistente de Dados
* Acesso de Multi-Usuário
* Linguagem de Definição de dados (DDL)

Análise de Requisitos

**Relações, Atributos, Restrições**

* Necessidades de armazenamento de dados
* Formato textual, gráfico e instância
* **Tabela e atributos**



Requisitos Funcionais

**Transações e Manipulação de Dados**

* **Modificações e Atualizações do Banco de Dados**

Modelagem Conceitual

**Entidades, Atributos, Relações, Generalização e Especialização, Entidade Fraca e Associativa**

* Especificação geral do sistema
* Entidades, Atributos e Relações
* Informações do Banco de Dados
* Objetos Concretos e Abstratos

Atributos

**CHAVE:**

**Atributo com um único valor (Id\_Funcionário) (Primary\_Key)**

**SIMPLES OU COMPOSTO:**

**Nome, Endereço**

**MULTIVALORADOS:**

**Atributo com vários valores (Sexo, Email)**



**ARMAZENADOS OU DERIVADOS:**

**Atributo derivado de outro (data de nascimento -> idade)**

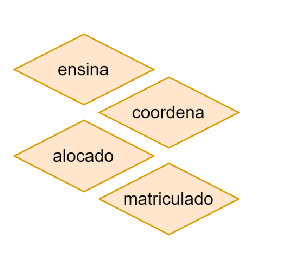


**VALORES NULL:**

**Pode ser negativo ou não**

Relacionamento

* Uma **Associação** entre duas ou mais **Entidades**
* Pertence / Possui



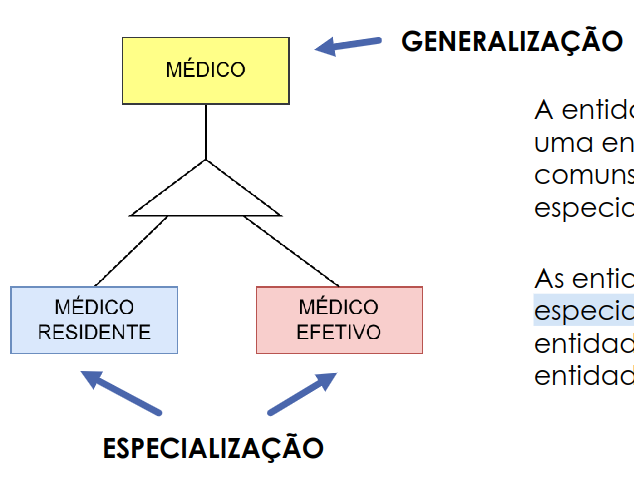
Docente Antônio David Viniski ensina Disciplina ABD’

-> **Ensina** é uma relação entre a **Entidade** Docente Antônia e Disciplina ABD

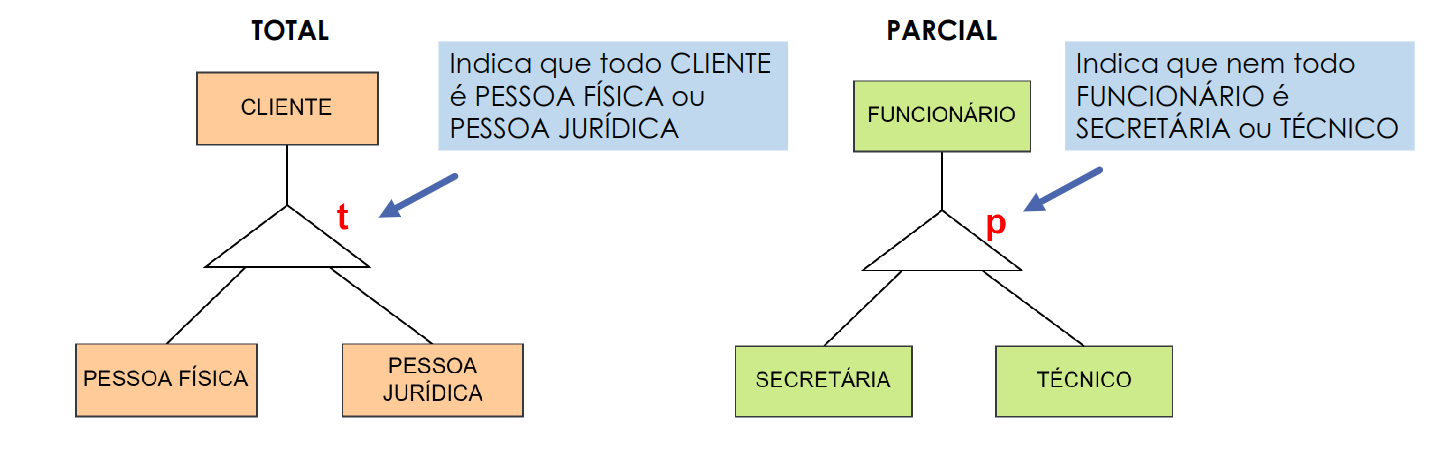
* Define um conjunto de associações entre N tipos de Entidades
* **Unário:** Apenas uma entidade, mesmo tipo de entidade
* **Binário:** Duas entidades (mais comum)
* **Ternário:**  Três entidades
* **N-ário:** Onde tem Quatro ou mais entidades

Generalização e Especialização

* Conceito para representar objetos do mundo real que possuem **os mesmos** Atributos
* Podem ser representados por hierarquias



* **Especialização Total:** Toda entidade precisa ser membro das subclasses
* **Especialização Parcial:** Nem toda entidade precisa ser membro das subclasses

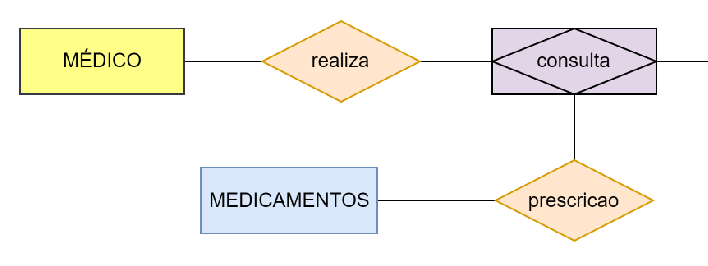


Entidade Fraca

* Tipo de **Entidade** que não tem **Identidade Própria (Primary\_Key)/(Atributo Chave)**
* As instâncias são identificadas por uma chave parcial/chave estrangeira com outra entidade

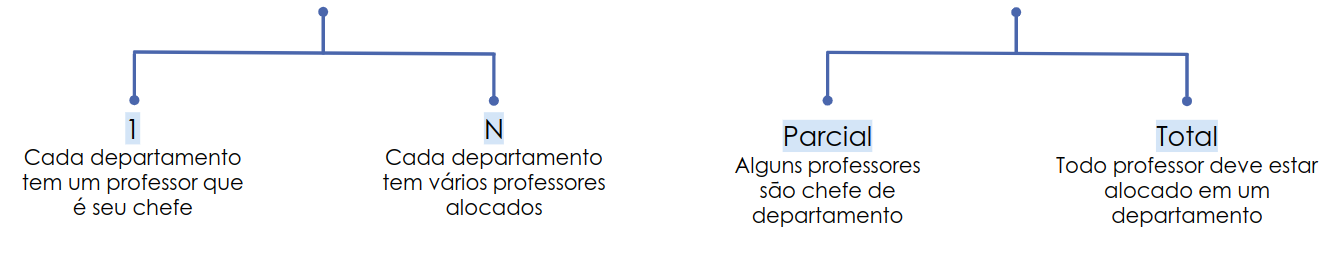
Entidade Associativa

* Adicionar relacionamentos a outra entidade já relacionada

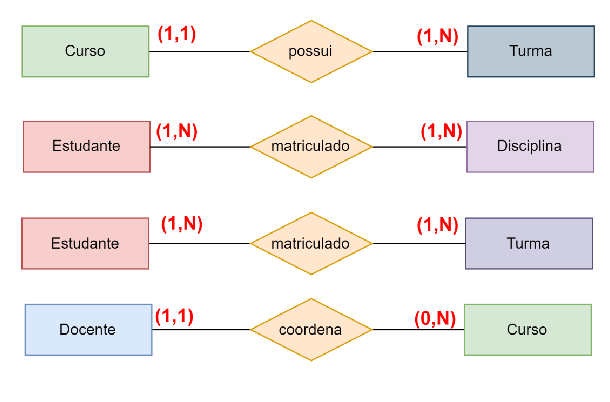


Cardinalidades

* Limitam as possíveis combinações de entidades que podem participar dos relacionamentos



* **Cardinalidade Mínima:** Número mínimo de ocorrências
  + Cardinalidade Mínima = 0 (relacionamento opcional)
  + Cardinalidade Mínima = 1 (relacionamento obrigatório)
* **Cardinalidade Máxima:**  Quantidade máxima de ocorrências
  + Cardinalidade Máxima = 1 (docente locado em um único curso)
  + Cardinalidade Máxima = N (cursos possui vários docentes locados nele)



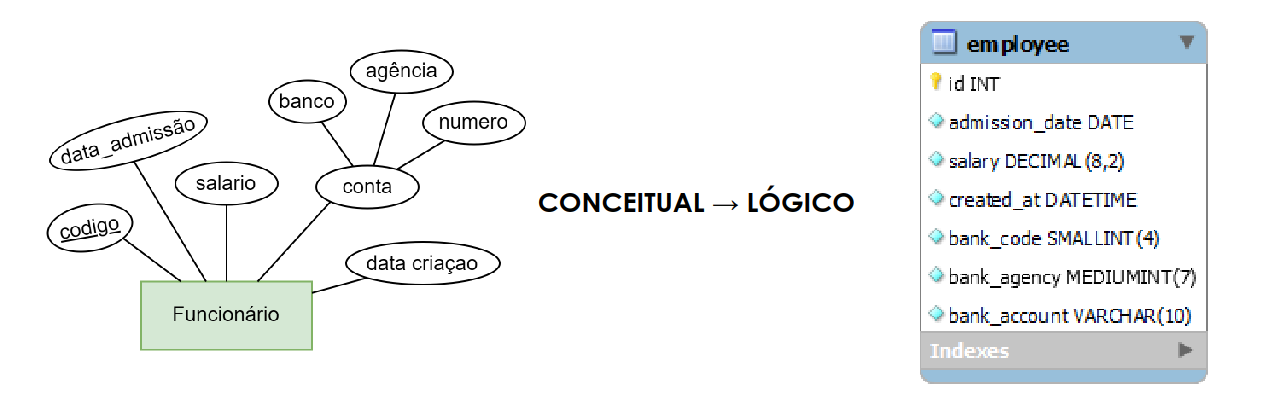
Mapeamento Lógico

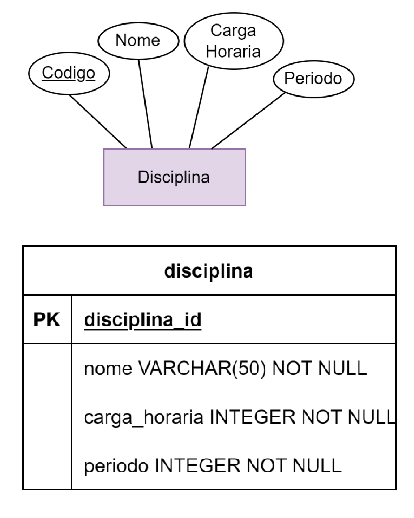
**Dependente do SGBD**

* Transformação do mapeamento conceitual para o modelo Lógico no MySql
* Lista de atributos, suas restrições
* Contém **Tabelas, Campos e Restrições de Integridade**
* Representação de um banco de dados relacional
* **Restrições de Integridade**
  + Domínio (valores permitidos nos campos)
  + Vazio (campos obrigatórios ou opcionais)
  + Chave (primária)
  + Referencial (chaves estrangeiras)
* **Chave Primária - Identificador Único (id\_person)**
  + **Primary Key**
* **Chave Estrangeira - Relaciona Duas Tabelas**
  + **Foreign Key**

Tipos De Dados

* **NUMÉRICOS**
  + **UNSIGNED** (aceita só valores positivos)
  + **ZEROFILL** (preenche ZEROS à esquerda do número)
  + **AUTO\_INCREMENT** (incrementa automaticamente números)
  + **SERIAL**
  + **BIT**
  + **TINYINT, SMALLINT, INT, MEDIUMINT, BIGINT**
  + **DECIMAL**
  + **FLOAT**
  + **DOUBLE**
* **DATA**
  + **DATE** (valores de data sem hora AAAA/MM/DD)
  + **DATETIME** (valores de data com hora AAAA/MM/DD hh:mm:ss)
  + **TIME** (valores do dia ou intervalos de tempo hh:mm:ss)
  + **TIMESTAMP** (valores que contêm parte de hora e data)
  + **YEAR** (valores de anos AAAA)
* **STRING**
  + **CHAR**
  + **VARCHAR**
  + **BINARY**
  + **VARBINARY**
  + **BLOB**
  + **TEXT**
  + **ENUM**
  + **SET**

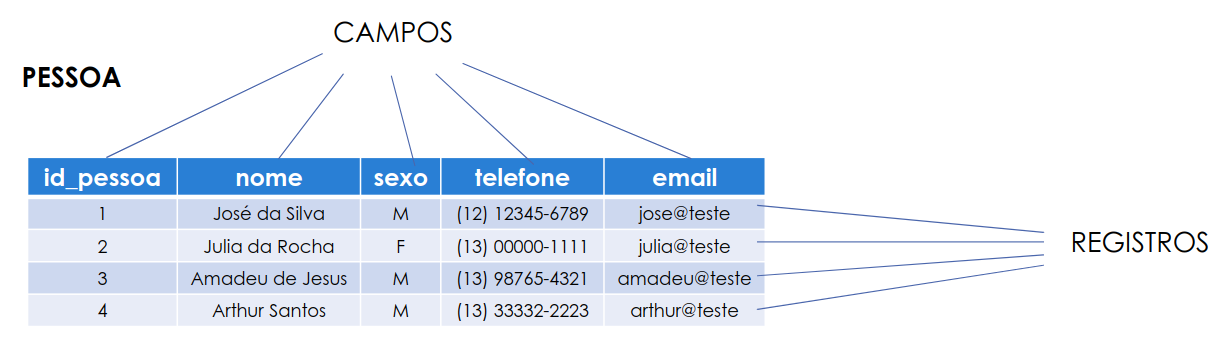




Modelo Físico

**MySQL, DDL, DQL , DML, DCL, DTL**

* Conceitos
* Criação / Exclusão
* Tabelas / Atributos / Restrições
  + Modificações
* Manipulação de Banco de Dados
  + INSERT, DELETE, UPDATE, SELECT



DATABASE

**CREATE DATABASE IF NOT EXISTS** restaurante;

* Cria data-base se não existir ela já

**CREATE SCHEMA IF NOT EXISTS** restaurante;

**DROP DATABASE IF EXISTS** restaurante;

* Deleta a data-base se já existir (Costume é colocar antes do CREATE)

**DROP SCHEMA IF EXISTS** restaurante;

**USE** restaurante;

TABELAS/TABLES

**CREATE TABLE restaurante.pessoa(**

id **INTERGER NOT NULL AUTO\_INCREMENT,**

nome **VARCHAR (50),**

cpf **CHAR (11) UNIQUE NOT NULL,**

sexo **CHAR (1)**

**PRIMARY KEY (id)**

**FOREIGN KEY (client\_id) REFERENCES cliente(id) );**