

Design e Desenvolvimento de Banco de Dados

Objetivos

Conceitos Fundamentais

- ▶ Compreender os conceitos básicos de banco de dados, como dados, informação e conhecimento.

SGBDs e suas Funções

- ▶ Explorar o papel do SGBD (Sistema Gerenciador de Banco de Dados) na gestão de informações e suas funções essenciais.

Modelo Relacional

- ▶ Apresentar o modelo relacional (SQL) e suas características.

Conceitos

Dados

São elementos brutos sem interpretação, como números, letras ou símbolos.

10552	João da Silva	15/03/2000	A
10567	Maria Souza	22/08/1999	B
10589	Pedro Santos	10/11/2001	A
10599	Ana Oliveira	05/05/2002	B
10612	Lucas Ferreira	30/07/2000	A

Conceitos

Informação

A informação surge da organização e interpretação dos dados, atribuindo significado.

Registro Acadêmico	Nome	Data de Nascimento	Turma
10552	João da Silva	15/03/2000	A
10567	Maria Souza	22/08/1999	B
10589	Pedro Santos	10/11/2001	A
10599	Ana Oliveira	05/05/2002	B
10612	Lucas Ferreira	30/07/2000	A

Conceitos

Conhecimento

Informação analisada e interpretada para tomada de decisão.

- ▶ Turma A tem 3 alunos
- ▶ Média idade turma A: 23,67 anos

- ▶ Turma B tem 2 alunos
- ▶ Média idade turma B: 24 anos

Conceitos

Banco de Dados

Um banco de dados é um conjunto organizado de dados, armazenados para facilitar o acesso e a gestão de informações.

Conceitos

Banco de Dados

Um banco de dados é um conjunto organizado de dados, armazenados para facilitar o acesso e a gestão de informações.



Banco de Dados (Arquivo)

Refere-se ao armazenamento de dados em arquivos simples, como CSV, TXT, JSON, XML ou até mesmo planilhas Excel.

- ▶ A manipulação dos dados é feita diretamente pelo programa que acessa esses arquivos.
- ▶ Se você salvar informações de clientes em um arquivo clientes.csv, cada linha pode representar um cliente. Se outro programa tentar modificar o mesmo arquivo ao mesmo tempo, pode haver problemas de corrupção ou perda de dados.

SGBD - Sistema Gerenciador de Banco de Dados

Um SGBD é um software especializado que permite criar, gerenciar e manipular bancos de dados de maneira estruturada e eficiente.

Funciona como uma camada intermediária entre os usuários e os dados, garantindo segurança, integridade e acessibilidade.

SGBD - Sistema Gerenciador de Banco de Dados

Armazenamento e Gerenciamento de Dados

- ▶ Organiza os dados em estruturas eficientes, permitindo fácil recuperação e manipulação.

Controle de Concorrência

- ▶ Gerencia o acesso simultâneo aos dados, evitando inconsistências.

Segurança e Controle de Acesso

- ▶ Define permissões para proteger os dados contra acessos não autorizados.

SGBD - Sistema Gerenciador de Banco de Dados

Mecanismo de Backup e Recuperação

- ▶ Garante a recuperação dos dados em caso de falhas.

Linguagem de Consulta

- ▶ Suporte a SQL (bancos relacionais) ou outras linguagens específicas para manipulação dos dados (bancos não relacionais).

SGBD - Sistema Gerenciador de Banco de Dados

Oracle Database

Lançamento: 1979

Empresa: Oracle Corporation

Linguagem: SQL, PL/SQL

Principais Características:

- ▶ Alta escalabilidade e desempenho.
- ▶ Suporte a grandes volumes de dados (Big Data).
- ▶ Ferramentas avançadas de segurança e backup.

Curiosidade: É o SGBD mais usado em empresas da Fortune 500.



SGBD - Sistema Gerenciador de Banco de Dados

MySQL



Lançamento: 1995

Empresa: Oracle Corporation (adquirido em 2010)

Linguagem: SQL

Principais Características:

- ▶ Open-source e gratuito (com versões pagas).
- ▶ Fácil de usar e configurar.
- ▶ Amplamente utilizado em aplicações web.

Curiosidade: É o SGBD mais popular para aplicações baseadas em PHP.

SGBD - Sistema Gerenciador de Banco de Dados

Microsoft SQL Server

Lançamento: 1989

Empresa: Microsoft

Linguagem: SQL, T-SQL

Principais Características:

- ▶ Integração com outras ferramentas Microsoft (ex.: Azure, Power BI).
- ▶ Ferramentas avançadas de análise de dados.
- ▶ Suporte a machine learning e IA.

Curiosidade: Amplamente utilizado em empresas que já usam o ecossistema Microsoft.



SGBD - Sistema Gerenciador de Banco de Dados

PostgreSQL

Lançamento: 1996

Empresa: Comunidade open-source

Linguagem: SQL, PL/pgSQL

Principais Características:

- ▶ Open-source e gratuito.
- ▶ Suporte a JSON, GIS e extensões personalizadas.
- ▶ Conhecido por sua robustez e conformidade com padrões SQL.

Curiosidade: É o SGBD open-source mais avançado do mercado.



SGBD - Sistema Gerenciador de Banco de Dados

SQLite



Lançamento: 2000

Empresa: Comunidade open-source

Linguagem: SQL

Principais Características:

- ▶ Banco de dados embarcado (não requer servidor).
- ▶ Leve e de fácil integração.
- ▶ Ideal para aplicações móveis e desktop.

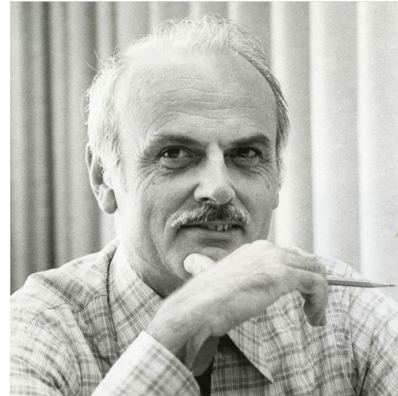
Curiosidade: É o SGBD mais usado no mundo, presente em bilhões de dispositivos.

SGBD - Sistema Gerenciador de Banco de Dados

SGBD	Tipo	Licença	Destaque	Melhor Para
Oracle	Proprietário	Paga	Escalabilidade e segurança	Grandes corporações
MySQL	Open-source	Gratuita/Paga	Facilidade de uso	Aplicações web
SQL Server	Proprietário	Paga	Integração com Microsoft	Empresas com ecossistema Microsoft
PostgreSQL	Open-source	Gratuita	Conformidade com padrões SQL	Sistemas complexos
SQLite	Open-source	Gratuita	Leve e embarcado	Aplicativos móveis e desktop

Banco de Dados Relacional

Um **banco de dados relacional** é um sistema de armazenamento de dados que organiza informações em **estruturas tabulares** (tabelas), seguindo os princípios do modelo relacional proposto por Edgar F. Codd em 1970, enquanto trabalhava como pesquisador na IBM.



Banco de Dados Relacional

As informações são estruturadas em tabelas, que representam entidades do mundo real, como clientes, produtos ou pedidos. Cada tabela é composta por linhas (registros individuais) e colunas (atributos que definem as propriedades dos dados).

Banco de Dados Relacional

Tabelas

- ▶ Representam entidades do mundo real (ex.: Clientes, Produtos).
- ▶ Organizam dados de forma tabular, com linhas e colunas.

Banco de Dados Relacional

Linhas (Tuplas)

- ▶ Correspondem a registros individuais (ex.: um cliente específico).
- ▶ Cada linha contém um conjunto de valores que descrevem uma instância da entidade.

Banco de Dados Relacional

Colunas (Atributos)

- ▶ Definem as propriedades dos dados (ex.: ID, Nome, Email, Foto).
- ▶ Cada coluna tem um tipo de dado específico (ex.: inteiro, texto, data, imagem).

Banco de Dados Relacional

Além da estrutura básica de tabelas, linhas e colunas, o modelo relacional se destaca por sua capacidade de **estabelecer relacionamentos** entre entidades e garantir a **integridade dos dados**.

Para isso, o modelo utiliza chaves, como a **chave primária** (identificador único) e a **chave estrangeira** (que estabelece relacionamentos lógicos entre tabelas).

Banco de Dados Relacional

Chaves

- ▶ **Primária:** Identificador único de cada registro (ex.: CPF em Clientes, Registro Acadêmico em Alunos).

Banco de Dados Relacional

Chaves

- ▶ **Estrangeira:** Liga tabelas, garantindo relacionamentos lógicos (ex.: Clienteld em Pedidos, Alunold em Aulas).


Banco de Dados Relacional

Tipo de Relacionamento 1 para 1 (1:1)

- ▶ Uma linha em uma tabela está associada a apenas uma linha em outra tabela.
Exemplo: Funcionário ↔ Carteira de Trabalho.

Banco de Dados Relacional

FUNCIONARIO	
ID 	int
Nome	varchar
CPF	varchar

CARTEIRA_TRABALHO	
ID 	int
Numero	varchar
Funcionario_ID	int



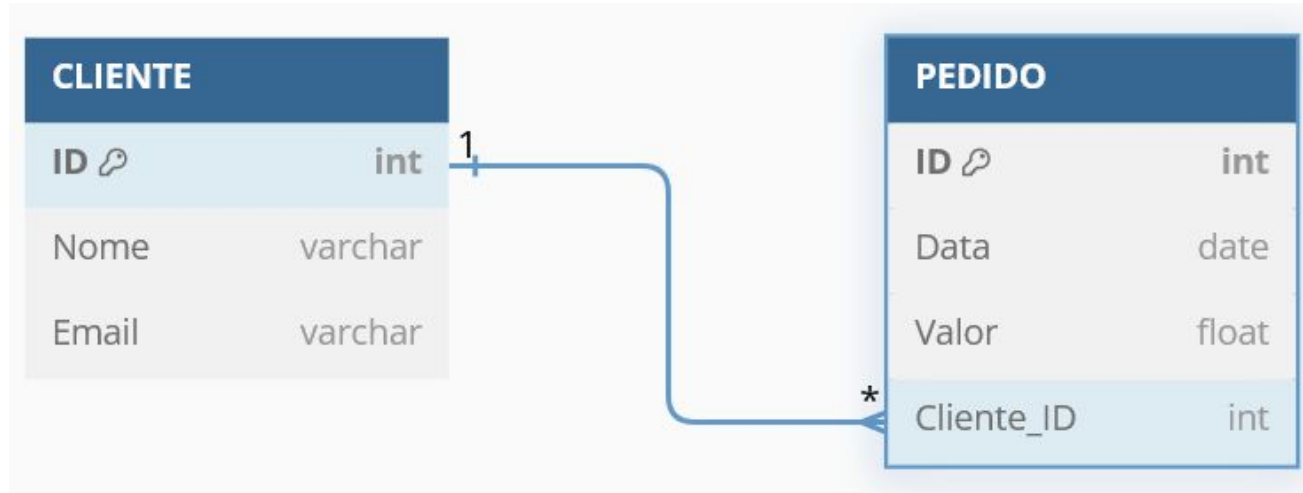
Banco de Dados Relacional

Tipo de Relacionamento 1 para Muitos (1:N)

- ▶ Uma linha em uma tabela pode estar associada a várias linhas em outra tabela.

Exemplo: Cliente ↔ Pedidos (um cliente faz vários pedidos).

Banco de Dados Relacional



Banco de Dados Relacional

Tipo de Relacionamento Muitos para Muitos (N:N)

- ▶ Várias linhas em uma tabela podem estar associadas a várias linhas em outra tabela.

Exemplo: Alunos \leftrightarrow Cursos (um aluno faz vários cursos, e um curso tem vários alunos).

Banco de Dados Relacional

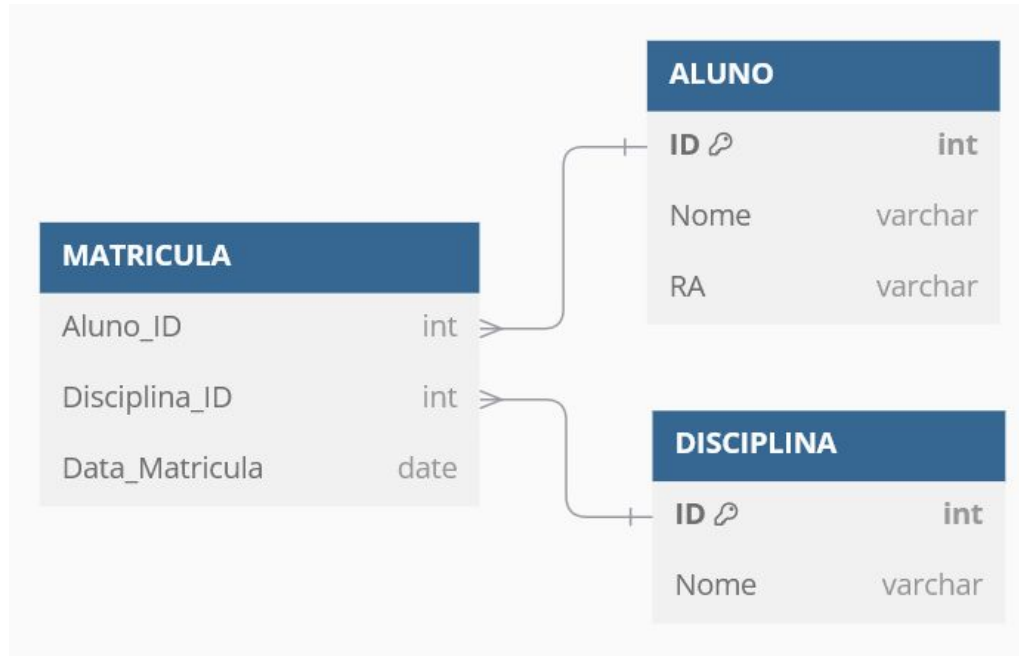
Tipo de Relacionamento Muitos para Muitos (N:N)

- ▶ Várias linhas em uma tabela podem estar associadas a várias linhas em outra tabela.

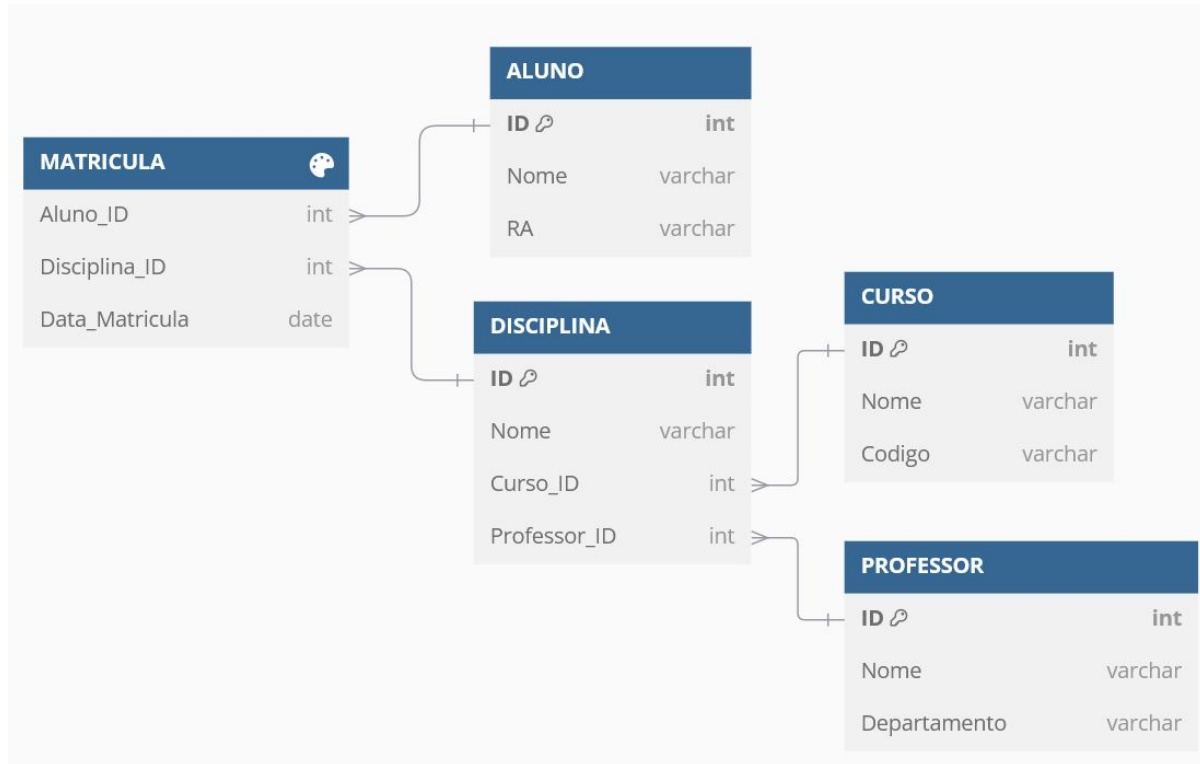
Exemplo: Alunos ↔ Cursos (um aluno faz vários cursos, e um curso tem vários alunos).

Implementação: Requer uma tabela associativa (ex.: Matrículas).

Banco de Dados Relacional



Banco de Dados Relacional



Banco de Dados Relacional

Integridade de Entidade

- ▶ Garante que cada tabela tenha uma chave primária única.

Exemplo: Dois clientes não podem ter o mesmo ID.

Banco de Dados Relacional

Integridade Referencial

- ▶ Garante que as chaves estrangeiras sempre referenciam chaves primárias válidas.

Exemplo: Um Pedido não pode ser associado a um Cliente_ID que não existe.

Banco de Dados Relacional

Integridade de Domínio

- ▶ Garante que os valores em uma coluna respeitem regras pré-definidas (tipo de dado, intervalo de valores, etc).

Exemplo: A coluna Idade só pode conter números positivos.

Banco de Dados Relacional

Normalização de Dados

A normalização é um processo essencial no design de bancos de dados relacionais. Ela visa organizar os dados de forma eficiente, eliminando redundâncias e inconsistências, e garantindo a integridade dos dados.

Banco de Dados Relacional

Normalização de Dados

A normalização é dividida em etapas (formas normais) que garantem a organização progressiva dos dados. Cada nível resolve problemas específicos de redundância e dependência.

Banco de Dados Relacional

1a Forma Normal (1FN):

- ▶ Cada coluna deve conter valores atômicos (indivisíveis).

Exemplo: Evitar listas ou múltiplos valores em uma única célula.

Banco de Dados Relacional

1a Forma Normal (1FN):

Aluno_ID	Nome	Cursos_Matriculados
1	Ana	Matemática, Física
2	Carlos	Química

Banco de Dados Relacional

1a Forma Normal (1FN):

Aluno_ID	Nome	Curso
1	Ana	Matemática
1	Ana	Física
2	Carlos	Química

Banco de Dados Relacional

2a Forma Normal (2FN)

- ▶ Todos os atributos não-chave devem depender totalmente da chave primária.

Eliminar dependências parciais (atributos que dependem apenas de parte de uma chave composta).

Banco de Dados Relacional

2a Forma Normal (2FN)

Matricula_ID	Aluno_ID	Curso_ID	Nome_Aluno	Nota
101	1	MAT101	Ana	9.5
102	2	QUIM202	Carlos	8.0

Banco de Dados Relacional

2a Forma Normal (2FN)

Matricula_ID	Aluno_ID	Curso_ID	Nome_Aluno	Nota
101	1	MAT101	Ana	9.5
102	2	QUIM202	Carlos	8.0

Problema:

- ▶ Nome_Aluno depende apenas de Aluno_ID, não da chave composta (Aluno_ID, Curso_ID).

Banco de Dados Relacional

2a Forma Normal (2FN)

Aluno

Aluno_ID	Nome
1	Ana
2	Carlos

Matrícula

Matricula_ID	Aluno_ID	Curso_ID	Nota
101	1	MAT101	9.5
102	2	QUIM202	8.0

Banco de Dados Relacional

3a Forma Normal (3FN)

- ▶ Eliminar dependências transitivas (atributos que dependem de outros atributos não-chave)

Banco de Dados Relacional

3a Forma Normal (3FN)

Cliente_ID	Nome	Cidade	Estado
1	João	São Paulo	SP
2	Maria	Rio de Janeiro	RJ

Banco de Dados Relacional

3a Forma Normal (3FN)

Cliente_ID	Nome	Cidade	Estado
1	João	São Paulo	SP
2	Maria	Rio de Janeiro	RJ

Problema:

- ▶ Estado depende de Cidade, não diretamente de Cliente_ID.

Banco de Dados Relacional

3a Forma Normal (3FN)

Cliente

Cliente_ID	Nome	Cidade_ID
1	João	1
2	Maria	2

Cidade

Cidade_ID	Cidade	Estado
1	São Paulo	SP
2	Rio de Janeiro	RJ

Banco de Dados Relacional

Forma Normal de Boyce-Codd (BCNF)

- ▶ Versão mais rigorosa da 3FN.
- ▶ Toda dependência funcional deve ter uma chave candidata como determinante.

Banco de Dados Relacional

Forma Normal de Boyce-Codd (BCNF)

Curso_ID	Professor_ID	Departamento
MAT101	100	Exatas
FIS201	100	Exatas

Banco de Dados Relacional

Forma Normal de Boyce-Codd (BCNF)

Curso_ID	Professor_ID	Departamento
MAT101	100	Exatas
FIS201	100	Exatas

Problema:

- ▶ Departamento depende de Professor_ID, que não é uma chave candidata.

Banco de Dados Relacional

Forma Normal de Boyce-Codd (BCNF)

Curso

Curso_ID	Professor_ID
MAT101	100
FIS201	100

Professor

Professor_ID	Departamento
100	Exatas

Banco de Dados Relacional

4a Forma Normal (4FN)

- ▶ Eliminar dependências multivaloradas (quando um atributo depende de outro, mas não da chave inteira).

Banco de Dados Relacional

4a Forma Normal (4FN)

Aluno_ID	Telefones	Cursos
1	9999-8888, 7777-6666	Matemática, Física

Banco de Dados Relacional

4a Forma Normal (4FN)

Aluno_Telefone

Aluno_ID	Telefone
1	9999-8888
1	7777-6666

Aluno_Curso

Aluno_ID	Curso
1	Matemática
1	Física

Banco de Dados Relacional

Normalização de Dados

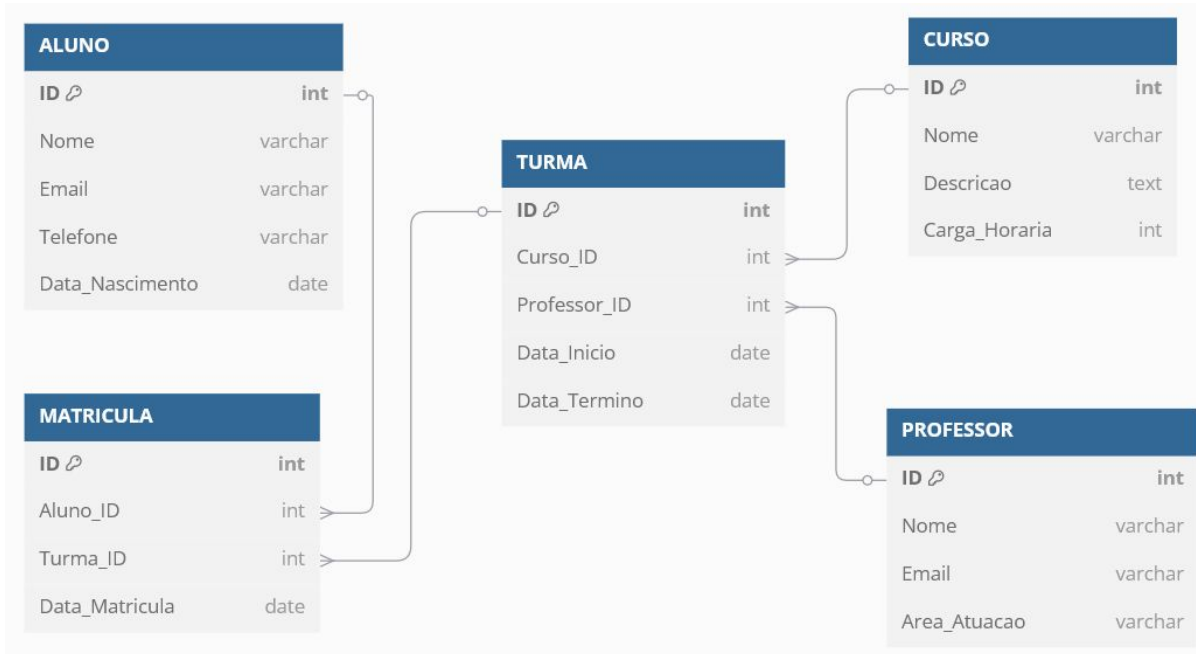
Forma Normal	Objetivo Principal	Exemplo de Problema Resolvido
1FN	Valores atômicos	Listas em células
2FN	Dependência total da chave primária	Atributos dependentes de parte da chave
3FN	Eliminar dependências transitivas	<code>Estado</code> dependendo de <code>Cidade</code>
BCNF	Chave candidata como determinante	Dependência em atributo não-chave
4FN	Eliminar dependências multivaloradas	Múltiplos valores em uma coluna

Banco de Dados Relacional

Exercício

Criar um banco de dados (planilha) com pelo menos 5 tabelas, sendo a tabela principal com no mínimo 5 colunas. As tabelas devem ter relacionamentos entre si, e pelo menos 4 delas devem estar conectadas. Após a modelagem, popular as tabelas com ao menos 10 linhas de dados, garantindo que os relacionamentos sejam corretamente representados.

Banco de Dados Relacional



Boa noite!