

Modelagem de Banco de dados (SQL e NoSQL)

Prof. Daniel Brandão

Bancos de dados

Banco de dados - Definição

O que é um BANCO DE DADOS?

“Um banco de dados é uma coleção de dados persistentes armazenados utilizados pelas aplicações de uma determinada organização ou empresa.”¹

¹C.J. Date (2000)

O que é um banco de dados?

Sistema que reúne e mantém **organizada** uma série de **informações relacionadas** a um determinado assunto em uma determinada ordem.

Uma **agenda de contatos** é um exemplo simples, nela percebemos que todos os dados referentes a uma pessoa estão **agrupadas**, a isso chamamos **registros**.

Utilizamos SGBDs para controlar os registros.

Modelo Relacional - Definição

- O **Modelo Relacional** é um conceito abstrato que define maneiras de **armazenar, manipular e recuperar dados** estruturados unicamente na forma de tabelas, construindo um banco de dados.
- O termo é aplicado aos próprios dados, quando organizados dessa forma, ou a um **Sistema Gerenciador de Banco de Dados (SGBD)** - do inglês Relational DataBase Management System (RDBMS) - um programa de computador que implementa a abstração.



**SGBDs = Sistema
Gerenciador de
Banco de dados**

Exemplos:

7

Conceitos Gerais: **Atributo**

Um **item de dado** do Banco de Dados (BD)

Possui um **nome** e um **domínio**

Exemplos

- nome: varchar(20)
- matrícula: integer
- dataNasc: date



Nome	Matrícula	DataNasc
------	-----------	----------

Exemplo: Tupla

Tupla 1



Nome	Matrícula	DataNasc
Renata	01035	12/11/1980
Vânia	02467	03/07/1976
Maria	01427	20/02/1985

Atributo: Nome
Valor: Renata



Tipos de Dados em Banco de Dados

Tipos de Dados em Banco de Dados:

- A escolha do tipo de dado correto é crucial para garantir a integridade e a eficiência do banco de dados. Os tipos de dados definem que informação poderá ser obtida de cada input.
- É importante entender que, dependendo do SGBD escolhido, alguns tipos podem se comportar de maneira diferente. É sempre bom consultar a documentação do sistema de gerenciamento de banco de dados para obter mais detalhes sobre os tipos de dados disponíveis.

Tipos de Dados em Banco de Dados:

1. Numéricos:

- **Inteiros:** INT, SMALLINT, BIGINT (armazenam números inteiros sem casas decimais).
- **Reais:** FLOAT, DOUBLE, DECIMAL (armazenam números com casas decimais).

2. Alfanuméricos:

- **Texto:** VARCHAR(n), CHAR(n) (armazenam sequências de caracteres com tamanho fixo ou variável).

Tipos de Dados em Banco de Dados:

3. Datas e Horas:

DATE: armazena datas no formato YYYY-MM-DD.

TIME: armazena horas no formato HH:MM:SS.

DATETIME: armazena datas e horas no formato YYYY-MM-DD
HH:MM:SS.

4. Booleanos:

BOOLEAN, TINYINT(1) (armazenam valores TRUE ou FALSE).

Tipos de Dados em Banco de Dados:

5. Outros:

BLOB: armazena grandes objetos binários (imagens, arquivos).

ENUM: armazena valores predefinidos em uma lista.

SET: armazena um conjunto de valores de um tipo de dado específico.

SERIAL: número sequencial automático a partir de 1.



Atributos Chave

Atributos Chave

- **Atributos chave** são características que identificam unicamente cada registro em uma tabela de banco de dados.
- São como a “carteira de identidade” de cada linha na tabela, garantindo que **não existam duplicatas** e que cada registro seja facilmente acessível.

Atributos Chave

- Atributos chave são aqueles que podem **identificar unicamente** cada registro em um BD.
- A escolha das chaves é fundamental para o bom funcionamento do banco de dados.
- É importante ter um bom conhecimento dos tipos de dados e das chaves para projetar um banco de dados eficiente e confiável.

Tipos de Atributos Chave

1. Chave Primária (PK):

- Identifica unicamente cada registro na tabela.
- Não pode conter valores **nulos**.
- É geralmente um campo numérico auto-incrementado.

2. Chave Estrangeira (FK):

- Estabelece uma relação entre duas tabelas.
- Faz referência à chave primária de outra tabela.
- Garante a integridade referencial dos dados.

Tipos de Atributos Chave

3. Chave Candidata:

- Qualquer atributo ou conjunto de atributos que identifica unicamente cada registro na tabela.
- A chave primária é sempre uma chave candidata.
- Pode haver mais de uma chave candidata em uma tabela.

4. Superchave:

- Qualquer conjunto de atributos que determina todos os outros atributos da tabela.
- Uma superchave pode conter chaves candidatas e outros atributos.

Tipos de Atributos Chave

5. Atributo Composto:

Chave formada por mais de um atributo.

Exemplo: (nome, sobrenome) pode ser a chave primária de uma tabela de pessoas.

Tabela: **CLIENTES**

COLUNA	TIPO DE DADO	DESCRIÇÃO
id_cliente	INT	Identificador do cliente
nome	VARCHAR(255)	Nome do cliente
email	VARCHAR(255)	Email do cliente
cpf	VARCHAR(11)	CPF do cliente
data_nascimento	DATE	Data de nascimento do cliente



**Exemplos
Práticos**

Tabela: CLIENTES

COLUNA	TIPO DE DADO	CHAVE	DESCRIÇÃO
id_cliente	INT	?	Identificador único do cliente
nome	VARCHAR(255)	?	Nome do cliente
email	VARCHAR(255)	?	Email do cliente
cpf	VARCHAR(11)	?	CPF do cliente
data_nascimento	DATE	?	Data de nascimento do cliente

**Que atributo
pode ser a
PK?**

Tabela: CLIENTES

COLUNA	TIPO DE DADO	CHAVE	DESCRIÇÃO
id_cliente	INT	?	Identificador único do cliente
nome	VARCHAR(255)	?	Nome do cliente
email	VARCHAR(255)	?	Email do cliente
cpf	VARCHAR(11)	?	CPF do cliente
data_nascimento	DATE	?	Data de nascimento do cliente

**Que atributo
pode ser a
PK?**

Tabela: CLIENTES

COLUNA	TIPO DE DADO	CHAVE	DESCRIÇÃO
id_cliente	INT	PK	Identificador único do cliente
nome	VARCHAR(255)	-	Nome do cliente
email	VARCHAR(255)	-	Email do cliente
cpf	VARCHAR(11)	-	CPF do cliente
data_nascimento	DATE	-	Data de nascimento do cliente

**Que atributo
pode ser a
PK?**

Tabela: **PRODUTOS**

Coluna	Tipo de Dado	Chave	Descrição
id_produto	INT	?	Identificador único do produto
nome	VARCHAR(255)	?	Nome do produto
descricao	TEXT	?	Descrição do produto
preco	DECIMAL	?	Preço do produto
categoria	VARCHAR(255)	?	Categoria do produto

**Que atributo
pode ser a
PK?**

Tabela: **PRODUTOS**

Coluna	Tipo de Dado	Chave	Descrição
id_produto	INT	PK	Identificador único do produto
nome	VARCHAR(255)		Nome do produto
descricao	TEXT		Descrição do produto
preco	DECIMAL		Preço do produto
categoria	VARCHAR(255)		Categoria do produto

**Que atributo
pode ser a
PK?**

Tabela: **PEDIDOS**

COLUNA	TIPO DE DADO	CHAVE	DESCRIÇÃO
id_pedido	INT	?	Identificador único do pedido
id_cliente	INT	?	Referencia o cliente que fez o pedido (chave estrangeira)
id_produto	INT	?	Identificador único do produto
data_pedido	DATE	?	Data do pedido
valor_total	DECIMAL	?	Valor total do pedido

Que atributo
pode ser a
PK?

Tabela: **PEDIDOS**

COLUNA	TIPO DE DADO	CHAVE	DESCRIÇÃO
id_pedido	INT	PK	Identificador único do pedido
id_cliente	INT	FK	Referencia o cliente que fez o pedido (chave estrangeira)
id_produto	INT	FK	Identificador único do produto
data_pedido	DATE	-	Data do pedido
valor_total	DECIMAL	-	Valor total do pedido

Que atributo
pode ser a
PK?



Modelagem de dados

Modelagem de dados

- A modelagem de dados é o **processo** de **organizar os dados** em uma estrutura que permita uma análise mais eficiente e precisa. Isso geralmente envolve a identificação das **entidades** e **relacionamentos** existentes entre elas.

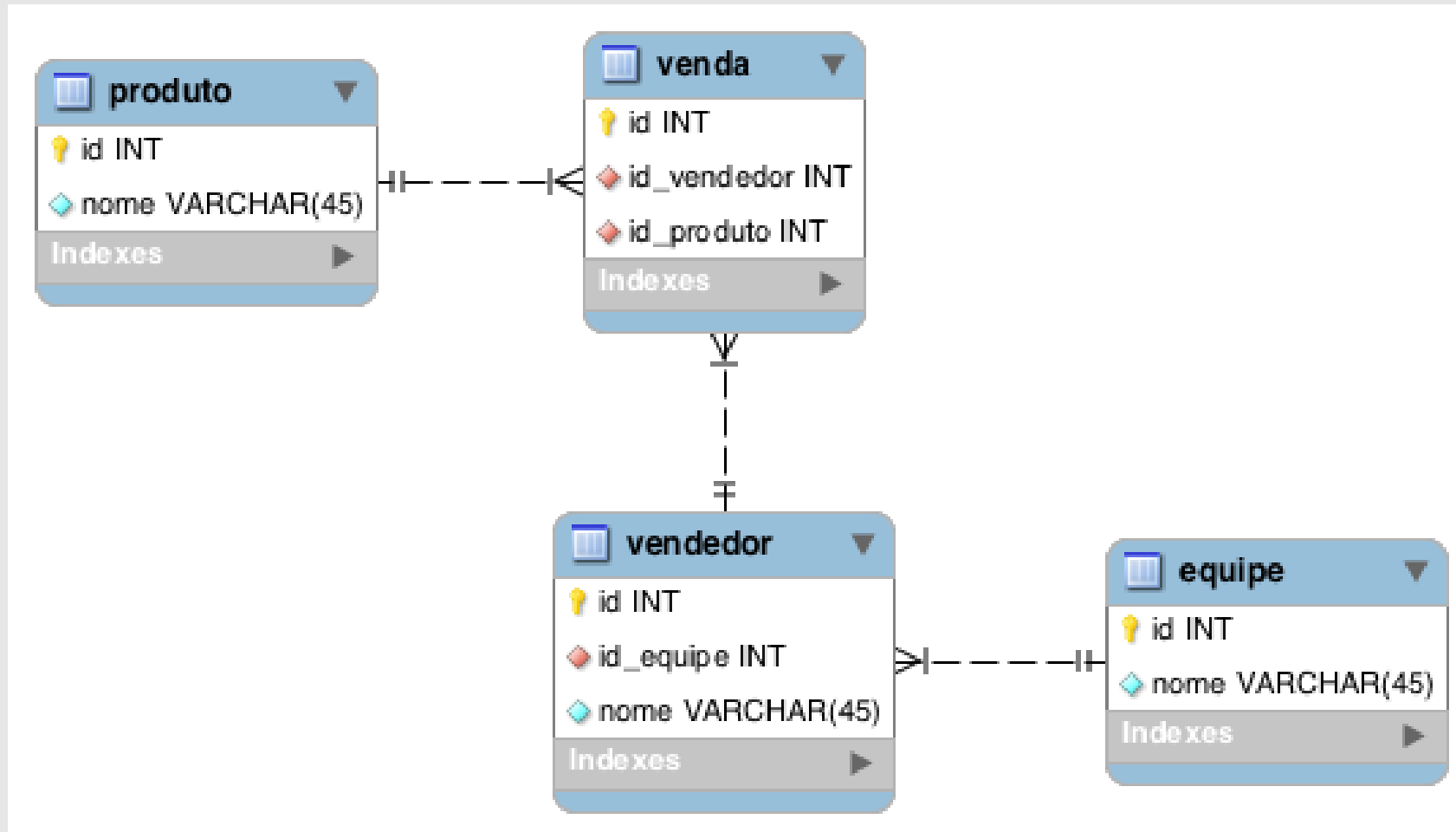
Modelagem de dados

- O objetivo é criar uma representação dos dados que seja fácil de entender e que possa ser usada para responder a perguntas de negócios.

Modelagem de dados

- Um banco de dados é uma coleção organizada de dados. Ele armazena informações em **tabelas**, onde cada tabela é composta de **colunas** (que representam os atributos dos dados) e **linhas** (que representam as entradas dos dados).
- Os bancos de dados são projetados para armazenar, gerenciar e recuperar informações com eficiência e segurança.

- Exemplo de Modelagem de dados





Cardinalidade

Cardinalidade

- Os relacionamentos entre tabelas são fundamentais na modelagem de dados. Eles acontecem através da quantidade de possibilidades que os dados podem se relacionar, chamadas de cardinalidade.
- Existem três tipos principais de relacionamentos:

1 para 1 | 1 para Muitos | Muitos para Muitos

Cardinalidade

1. **Um para um (1:1):** quando cada registro em uma tabela tem apenas um registro correspondente em outra tabela. Por exemplo, um funcionário pode ter apenas um endereço residencial. Por exemplo, cada apólice só pode ter um dono.

Cardinalidade

2. **Um para muitos (1:N):** quando cada registro em uma tabela pode ter muitos registros correspondentes em outra tabela. Por exemplo, um cliente pode ter muitos pedidos.

Cardinalidade

3. **Muitos para muitos (N:N):** quando muitos registros em uma tabela podem corresponder a muitos registros em outra tabela. Esse tipo de relacionamento requer uma tabela intermediária (também chamada de tabela de associação) para conectar as duas tabelas. Por exemplo, um aluno pode ter muitas disciplinas e uma disciplina pode ter muitos alunos.



SGBDs e Conexão de dados

SGBDs

- Sistemas Gerenciadores de Bancos de Dados (SGBDs) são softwares responsáveis por gerenciar e administrar bancos de dados. Eles fornecem um conjunto de ferramentas e funcionalidades para criar, manter, manipular, proteger e otimizar o acesso aos dados armazenados.
- Os SGBDs facilitam a interação entre os usuários e os bancos de dados, permitindo que eles executem operações como inserção, atualização, exclusão e consulta de dados.

SGBDs x SQL

- SGBDs relacionais usam SQL como forma de realizar as transações. Ela é manipulada através de uma sintaxe padrão para praticamente qualquer SGBD. Sua linguagem segue a seguinte estrutura:
- `SELECT * FROM produtos -- retorna todas as colunas da tabela produtos`
- `SELECT (nome) FROM clientes -- retorna todas as colunas da tabela clientes`
- `INSERT INTO produtos VALUES (xxx,yyy,zzz,kkk, ddd) -- Insere na tabela produtos novos dados`

SGBDs x Power BI

- UPDATE **produtos** WHERE id_produto = 20 --
atualiza um registro na tabela **produtos**
- DELETE produtos WHERE id = 99 -- exclui um
registro na tabela **produtos**

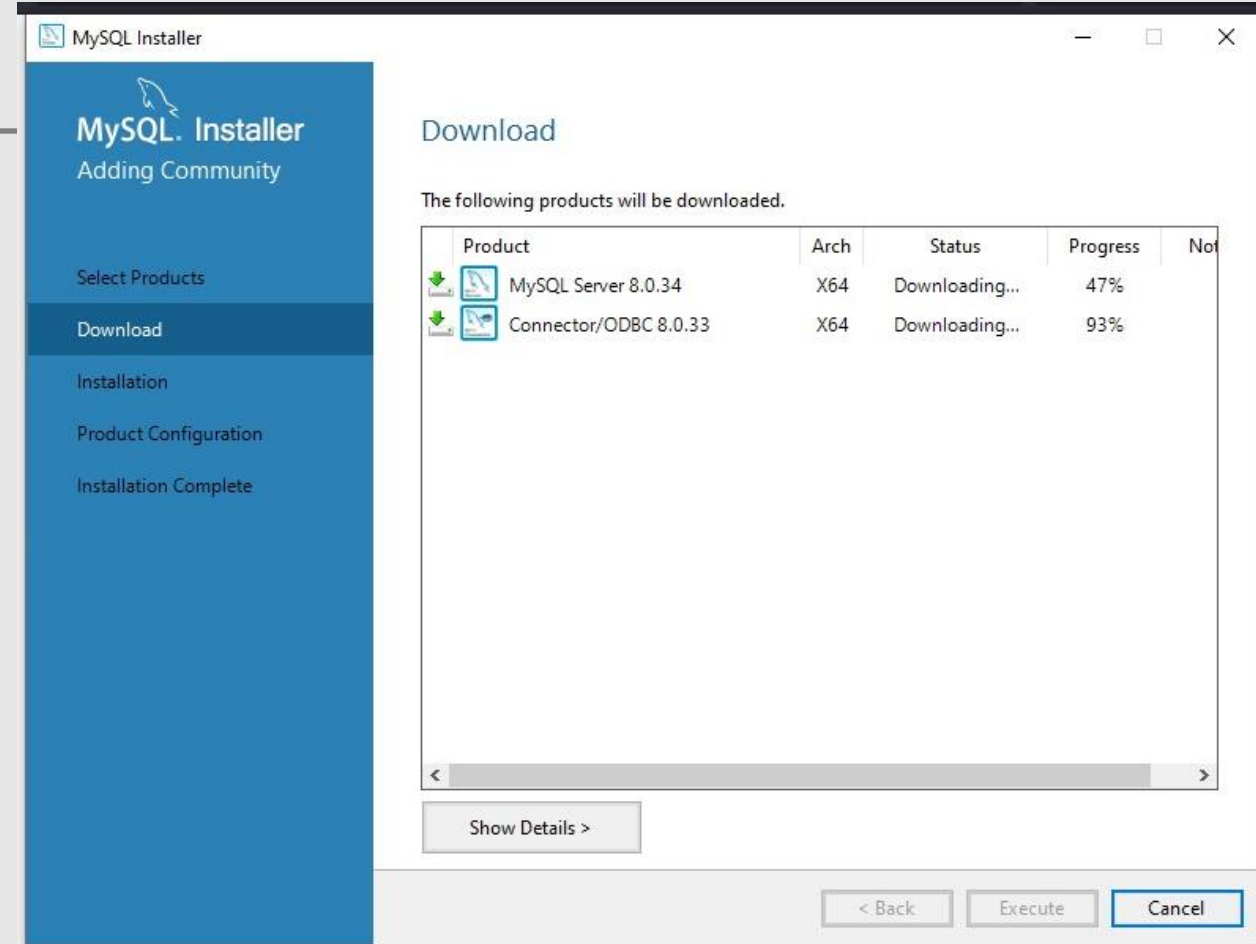


Conectando ao SGBD

SGBD MySQL

O que vamos precisar?

- MySQL (ou serviço online)
- MySQL Workbench



Prática com banco de dados MySQL

MÃO NA MASSA!

- Abra o Workbench
- Vamos conectar ao servidor
- Criaremos nosso BD e tabelas
- Realizaremos um CRUD

MÃO NA MASSA!

Exemplo 1: Criando uma tabela e inserindo dados

- Comando **CREATE TABLE**:

Criar a estrutura da tabela com nome, tipo de dados e tamanho das colunas.

Exemplo: `CREATE TABLE clientes (id INT, nome VARCHAR(255), email VARCHAR(255), cpf VARCHAR(11), data_Nascimento DATE());`

- Comando **INSERT INTO**:

Inserir registros na tabela com valores específicos para cada coluna.

Exemplo: `INSERT INTO clientes (id, nome, email, cpf) VALUES (1, 'João Silva', 'joaosilva@email.com', '72349257899');`

MÃO NA MASSA!

Exemplo 2: Selecionando e filtrando dados

- **Comando SELECT:**

Obter dados específicos da tabela.

Exemplo: `SELECT * FROM clientes;` (seleciona todos os campos)

Exemplo: `SELECT nome, email FROM clientes;` (seleciona campos específicos)

- **Cláusula WHERE:**

Filtrar dados com base em critérios específicos.

Exemplo: `SELECT * FROM clientes WHERE id = 1;` (seleciona cliente com ID 1)

Exemplo: `SELECT * FROM clientes WHERE nome LIKE '%Silva%';` (seleciona clientes com nome contendo "Silva")

MÃO NA MASSA!

Exemplo 3: Atualizando e excluindo dados

- **Comando UPDATE:**

Modificar dados existentes na tabela.

Exemplo: `UPDATE clientes SET nome = 'Maria Oliveira' WHERE id = 2;` (altera nome do usuário com ID 2)

- **Comando DELETE:**

Remover registros da tabela.

Exemplo: `DELETE FROM clientes WHERE email = 'joaosilva@email.com';` (exclui usuário com email específico)



INTERVALO

VOLTAMOS APÓS O ALMOÇO

DESAFIO I

- Seguindo a mesma lógica, crie agora as tabelas **PRODUTOS** e **PEDIDOS** de acordo com os exemplos passados;
- Realize o CRUD completo em cada uma delas, salve os códigos SQL utilizados para envio.

DESAFIO II

- Seguindo a mesma lógica, crie agora o projeto de banco de dados que sirva para o seu projeto real (seja algum de módulos passados, pensando no projeto final ou um projeto pessoal);
- Realize o CRUD completo em cada uma delas, salve os códigos SQL utilizados para envio.

Considerações Finais

Referências

- DATE, C.J. Introdução a Sistemas de Bancos de Dados (8ª edição). Editora: Pearson: São Paulo, 2023.
- Roben J. Coronel, Carlos Coronel. Banco de Dados: Conceitos, Modelagem e Design (6ª edição). Editora: Cengage Learning: São Paulo, 2022.
- FORTA, Ben. SQL in Ten Minutes, Sams Teach Yourself (5ª edição). Editora: Sams Publishing: Indianapolis, 2022
- BRAGHITTONI, Ronaldo. **Business Intelligence - Implementar do jeito certo e a custo zero**. São Paulo: Casa do Código, 2017.

Arquivos: **bit.ly/arquivosposbi**