

Banco de Dados I

Ronierison Maciel

Agosto 2024

Quem sou eu?



Nome: Ronierison Maciel / Roni

Formação: Mestre em Ciência da Computação

Ocupação: Pesquisador, Professor e Desenvolvedor de Software

Hobbies: Jogar cartas, ficar com a família no final de semana conversando sobre diversos temas

Interesses: Carros, aprimoramento na área educacional, desenvolvimento de software, data science e machine learning

Email: ronierison.maciел@pe.senac.br

GitHub: <https://github.com/ronierisonmaciel>

- 1 Introdução
- 2 Modelo de dados relacional e SQL básico
- 3 Modelagem de dados

Tópicos:

- Visão geral sobre BD e SGBD, instalação do MySQL
- Modelos de Dados, esquemas e arquiteturas
- Linguagens e interfaces de SGBDs, criação de tabelas

Objetivo:

- Introduzir os conceitos fundamentais de Banco de Dados e realizar a configuração inicial do MySQL

O que são Bancos de Dados (BD)?

Um Banco de Dados é uma coleção organizada de dados, tipicamente armazenados e acessíveis eletronicamente.

- **Exemplo:** Catálogo de produtos de uma loja, lista de alunos de uma escola.

| CLIENTE | |
|---------|--------|
| CODIGO | NOME |
| 1234 | CARLOS |
| 5678 | JOÃO |
| 9101 | PEDRO |
| 1213 | MARIA |

| VENDEDOR | |
|----------|---------|
| CODIGO | NOME |
| 11 | CARMEM |
| 12 | DJANIRA |
| 13 | ZECA |
| 14 | MARIO |

| PRODUTO | |
|---------|-----------|
| CODIGO | DESCRICAO |
| 123 | LAPIS |
| 456 | CANETA |
| 789 | PAPELA4 |
| 101 | TESOURA |
| 123 | BORRACHA |
| 141 | LIVRO |

| CONTÉM | | |
|--------|---------|------------|
| PEDIDO | PRODUTO | QUANTIDADE |
| 100/05 | 123 | 10 |
| 100/05 | 789 | 20 |
| 101/05 | 456 | 30 |
| 102/05 | 456 | 40 |
| 103/05 | 101 | 50 |
| 103/05 | 121 | 60 |
| 103/05 | 141 | 70 |
| 104/05 | 456 | 80 |

| PEDIDO | | | |
|--------|----------|----------|---------|
| NUMERO | DATA | VENDEDOR | CLIENTE |
| 100/05 | 01/01/05 | 12 | 5678 |
| 101/05 | 01/02/05 | 11 | 9101 |
| 102/05 | 01/03/05 | 13 | 1213 |
| 103/05 | 01/04/05 | 14 | 1234 |
| 104/05 | 01/05/05 | 12 | 1213 |

Figure: Tabelas

O que é um Sistema de Gerenciamento de Banco de Dados (SGBD)?

Conteúdo:

- **Definição:** Um SGBD é um software que permite a criação, gestão, manipulação e controle de acesso a bancos de dados.
- **Funções:** Controle de concorrência, recuperação de falhas, segurança e integridade de dados.

Exemplos: MySQL, PostgreSQL, Oracle, SQL Server.

Por que Bancos de Dados são importantes?

- Centralização e organização dos dados.
- Suporte à tomada de decisão.
- Eficiência e escalabilidade em operações de negócio.

Casos de Uso: Comércio eletrônico, sistemas de gerenciamento de clientes (CRM), sistemas de controle financeiro.

Visão geral dos SGBDs mais usados

- **MySQL:** Open source, amplamente usado em aplicações web.
- **PostgreSQL:** Avançado, com suporte a tipos de dados complexos.
- **Oracle:** Focado em grandes empresas, oferece alta performance e segurança.
- **SQL Server:** Solução da Microsoft, integrada com outras ferramentas da empresa.

Gráfico Comparativo: Popularidade dos SGBDs (baseado em pesquisas recentes).

Instalação do MySQL

Como instalar o MySQL

Passos:

- Baixar o MySQL Community Server do site oficial.
- Executar o instalador e seguir as instruções.
- Configurar a senha do root e as opções de segurança.
- Verificar a instalação via terminal.

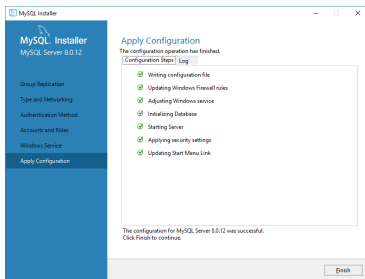


Figure: Instalação do MySQL

Configuração inicial do MySQL Workbench

O que é MySQL Workbench?

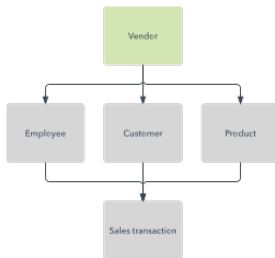
Ferramenta GUI para modelagem de dados, desenvolvimento SQL e administração de servidores.

- Conectar ao servidor MySQL.
- Criar um banco de dados.
- Explorar as funcionalidades básicas.

Modelos de dados

Introdução aos Modelos de Dados

- **Modelos hierárquicos:** Dados organizados em uma estrutura de árvore.
- **Modelos em rede:** Dados organizados em gráficos, permitindo múltiplas relações.
- **Modelos relacionais:** Dados organizados em tabelas, a base do MySQL.



| Student ID | First name | Last name |
|------------|------------|-----------|
| 52-743965 | Charles | Peters |
| 48-209689 | Anthony | Sondrup |
| 14-204968 | Rebecca | Phillips |

| ProviderID | Provider name |
|------------|----------------|
| 156-983 | UnitedHealth |
| 146-823 | Blue Shield |
| 447-784 | Carefirst Inc. |

| Student ID | ProviderID | Type of plan | Start date |
|------------|------------|--------------|------------|
| 52-743965 | 156-983 | HSA | 04/01/2016 |
| 48-209689 | 146-823 | HMO | 12/01/2015 |
| 14-204968 | 447-784 | HSA | 03/14/2016 |

Estrutura dos Bancos de Dados

- **Esquema:** A estrutura lógica de um banco de dados, definindo como os dados são organizados e inter-relacionados.

Arquiteturas:

- **Monolítica:** Todos os dados e serviços estão centralizados em um único sistema.
- **Cliente-Servidor:** Dados armazenados em servidores, acessados por clientes.
- **Distribuída:** Dados espalhados por múltiplos sistemas interconectados.

Linguagens de SGBDs

Linguagens e interfaces de SGBDs

Linguagens:

- **DDL** (Data Definition Language): Criação e modificação de estruturas de dados.
- **DML** (Data Manipulation Language): Inserção, atualização e exclusão de dados.

Interfaces:

- **CLI** (Command-Line Interface): Interação via comandos de texto.
- **GUI** (Graphical User Interface): Interação via interface gráfica, como o MySQL Workbench.

```
1 ALTER TABLE clientes ADD telefone VARCHAR(15);  
2 /* Comandos DDL */
```

```
1 SELECT nome, email FROM clientes WHERE data_cadastro > '  
    2023-01-01';  
2 /* Comandos DML */
```

Criação de tabelas em MySQL

Definindo tabelas e tipos de dados

Tipos de Dados:

- Numéricos (INT, FLOAT).
- Texto (VARCHAR, TEXT).
- Data/Hora (DATE, TIMESTAMP).

Exemplo de criação de tabela:

```
1 CREATE TABLE alunos (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     nome VARCHAR(100),  
4     data_nascimento DATE  
5 );
```

Descrição:

- 1 Instalar o MySQL e MySQL Workbench em seus computadores.
- 2 Criar um banco de dados simples e tabelas com diferentes tipos de dados.
- 3 Inserir alguns registros e praticar comandos básicos de consulta.

Entrega: Enviar um relatório com capturas de tela e código SQL até a próxima aula.

O que vamos fazer na próxima semana?

- Revisão dos conceitos básicos de SQL.
- Introdução ao modelo de dados relacional.
- Primeiras operações com SQL (SELECT, INSERT, UPDATE, DELETE).

Tópicos:

- Conceitos do modelo de dados relacional e tabelas relacionais
- Operações básicas em SQL (SELECT, INSERT, UPDATE, DELETE)
- Transações e controle de transações em MySQL

Objetivo:

- Entender o modelo de dados relacional e realizar operações básicas utilizando SQL no MySQL.

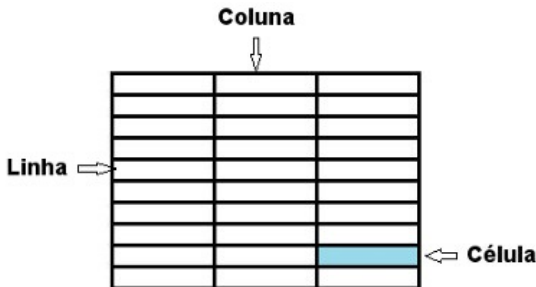
O que é o modelo de dados relacional?

Conceito:

- Modelo de dados que organiza informações em tabelas (também conhecidas como relações).

Elementos-chave:

- Tabelas:** Conjuntos de dados organizados em linhas e colunas.
- Linhas (Tuplas):** Cada linha representa um registro único.
- Colunas (Atributos):** Cada coluna representa um campo de dado dentro de um registro.



Componentes de uma tabela relacional

Estrutura de uma tabela relacional

- **Chave primária:** Coluna ou conjunto de colunas que identifica de forma única cada registro em uma tabela.
- **Chave estrangeira:** Coluna que cria uma ligação entre duas tabelas diferentes, referenciando a chave primária de outra tabela.
- **Índices:** Estruturas que melhoram a velocidade de operações de consulta nas tabelas.

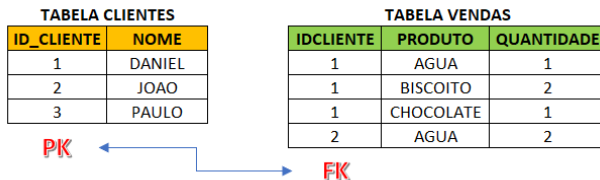
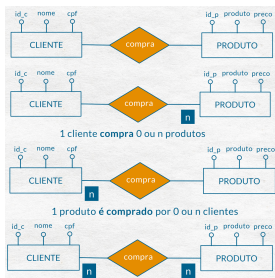


Figure: Foreign Key (FK) and Primary Key (PK)

Relações entre tabelas

Como relacionar tabelas no modelo relacional?

- **Relações de 1 para 1 (1:1):** Cada registro em uma tabela está relacionado a no máximo um registro em outra tabela.
- **Relações de 1 para muitos (1):** Um registro em uma tabela pode estar relacionado a múltiplos registros em outra tabela.
- **Relações de Muitos para muitos (M):** Múltiplos registros em uma tabela podem estar relacionados a múltiplos registros em outra tabela.



Definição de tabelas e relações no MySQL

Criando tabelas e definindo relações

Exemplo de SQL:

```
1 CREATE TABLE departamentos (  
2     id INT AUTO_INCREMENT PRIMARY KEY,  
3     nome VARCHAR(100)  
4 );  
5  
6 CREATE TABLE empregados (  
7     id INT AUTO_INCREMENT PRIMARY KEY,  
8     nome VARCHAR(100),  
9     departamento_id INT,  
10    FOREIGN KEY (departamento_id) REFERENCES departamentos (  
11    id)  
12 );
```

Prática: Criar tabelas e estabelecer relações no MySQL Workbench.



Introdução às operações básicas em SQL

- SELECT: Consulta de dados.
- INSERT: Inserção de novos registros.
- UPDATE: Atualização de registros existentes.
- DELETE: Exclusão de registros.

Consultas básicas com SELECT

```
1 SELECT coluna1, coluna2  
2 FROM tabela  
3 WHERE condição;
```

Cláusulas:

- WHERE: Filtros para as consultas.
- ORDER BY: Ordenação dos resultados.
- GROUP BY: Agrupamento de registros.

Operação INSERT

Inserindo dados em tabelas

```
1 INSERT INTO <nome_da_tabela> (<nomes_dos_atributos>)  
2 VALUES (<valores>);
```

Inserção de múltiplos registros de uma vez.

Atualizando registros com UPDATE

```
1 UPDATE tabela  
2 SET coluna1 = valor1, coluna2 = valor2  
3 WHERE condição;
```

Atualização condicional com WHERE.

Operação DELETE

Excluindo registros com DELETE

```
1 DELETE FROM tabela  
2 WHERE condição;
```

Cuidado ao usar DELETE sem cláusula WHERE.

O que são transações em Banco de Dados?

- Uma transaction (ou transação) é uma unidade de trabalho em um sistema de banco de dados que agrupa uma ou mais operações (como inserir, atualizar, excluir dados) que devem ser executadas de forma atômica e consistente. Ou seja, todas as operações dentro de uma transação devem ser concluídas com sucesso, ou nenhuma delas deve ser aplicada (em caso de falha, deve ocorrer um rollback).

Propriedades ACID:

- **Atomicidade:** Todas as operações são realizadas ou nenhuma.
- **Consistência:** O banco de dados deve permanecer em um estado consistente após a transação.
- **Isolamento:** Transações independentes umas das outras.
- **Durabilidade:** Após uma transação ser confirmada, as mudanças persistem.

Controle de transações no MySQL

Trabalhando com transações no MySQL

Comandos importantes:

- **START TRANSACTION:** Inicia uma nova transação.
- **COMMIT:** Confirma as mudanças realizadas pela transação.
- **ROLLBACK:** Reverte as mudanças realizadas pela transação.

Exemplo:

```
1 START TRANSACTION;  
2 UPDATE empregados SET salario = salario * 1.1 WHERE  
   departamento_id = 1;  
3 COMMIT;
```

Essa query usa uma transação para aumentar em 10% os salários dos empregados do departamento `departamento_id = 1`. A transação começa com `START TRANSACTION`, a atualização é feita pelo `UPDATE`, e o comando `COMMIT` torna as mudanças permanentes.

Descrição:

- 1 Criar tabelas relacionadas no MySQL.
- 2 Inserir, atualizar e excluir registros utilizando SQL básico.
- 3 Realizar consultas complexas com filtros e agrupamentos.
- 4 Implementar transações envolvendo múltiplas operações.

Entrega: Enviar código SQL e capturas de tela com os resultados até a próxima aula.

O que iremos aprender na próxima semana?

- Modelagem de Dados com ER e EER.
- Introdução ao projeto de banco de dados relacional.
- Aplicação de UML na modelagem de banco de dados.

Tópicos:

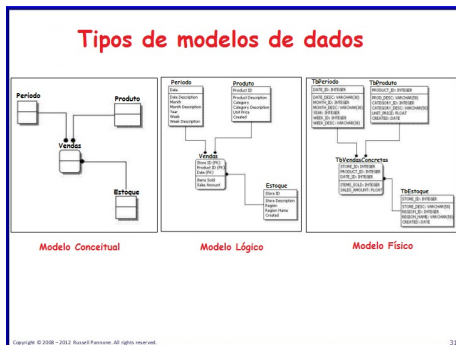
- Modelo Entidade-Relacionamento (ER)
- Modelo Entidade-Relacionamento Estendido (EER)
- Projeto de Banco de Dados Relacional e UML
- **Objetivos:**

Aprender a modelar dados utilizando ER e UML, aplicando técnicas avançadas de modelagem.

que é modelagem de dados?

Conceito de modelagem de dados

- **Definição:** A modelagem de dados é o processo de criar um modelo visual das informações que serão armazenadas em um banco de dados.
- **Objetivo:** Estruturar os dados de forma que possam ser armazenados, acessados e gerenciados de maneira eficiente.



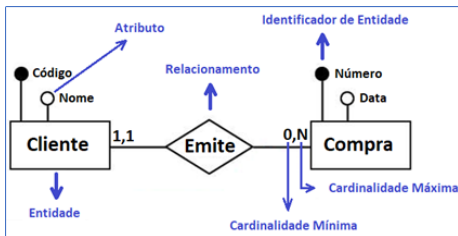
Entidade-Relacionamento (ER)

Conceito:

- O Modelo Entidade-Relacionamento é uma representação gráfica que descreve a estrutura lógica de um banco de dados.

Componentes:

- **Entidades:** Objetos ou conceitos sobre os quais os dados são armazenados (ex.: Cliente, Compra).
- **Atributos:** Características das entidades (ex.: Nome, Código).
- **Relacionamentos:** Associações entre entidades (ex.: Um cliente emite uma compra).



Detalhando os componentes do diagrama ER

Entidades:

- Representadas por retângulos. Exemplo: Clientes, Compra.

Atributos:

- Representados por elipses. Exemplo: Nome, Código.

Relacionamentos:

- Representados por losangos. Exemplo: Emite, Contém, Faz.

Chave primária:

- Um atributo ou conjunto de atributos que identifica de forma única uma entidade.

Modelagem de um sistema de vendas

1 Entidades identificadas:

- Cliente: Atributos: ClienteID, Nome, Endereço.
- Produto: Atributos: ProdutoID, NomeProduto, Preço.
- Pedido: Atributos: PedidoID, DataPedido, ClienteID.

1 Relacionamentos:

- Cliente faz Pedido.
- Pedido contém Produto.

Modelo Entidade-Relacionamento Estendido (EER)

Introdução ao modelo EER

- **Conceito:** O Modelo Entidade-Relacionamento Estendido é uma extensão do ER que incorpora conceitos mais complexos, como especialização, generalização e agregação.

Componentes adicionais:

- Generalização/Specialização: Permite que uma entidade seja uma subclasse de outra entidade.
- Agregação: Permite que um relacionamento seja tratado como uma entidade.