

Módulo 1: Introdução ao MySQL e Bancos de Dados Relacionais

Módulo 1: Introdução ao MySQL e Bancos de Dados Relacionais

1. O que é um Banco de Dados Relacional

Um Banco de Dados Relacional (BDR) é um conjunto de dados organizados em tabelas que se relacionam entre si por meio de chaves primárias e chaves estrangeiras. Ele foi introduzido por E. F. Codd em 1970 e se baseia no modelo relacional, onde os dados são armazenados em linhas (tuplas) e colunas (atributos) dentro de tabelas (relações). Essa abordagem permite acessar e manipular os dados de forma eficiente utilizando a linguagem SQL (Structured Query Language) (DATE, 2006).

2. Diferença entre Banco de Dados Relacional e Não Relacional

Os Bancos de Dados Relacionais (BDR) utilizam o modelo de tabelas com relações entre si. Em contrapartida, os Bancos de Dados Não Relacionais (NoSQL) armazenam dados de forma mais flexível, utilizando modelos de documentos, chave-valor, grafos ou colunas (Sadalage e Fowler, 2012). Abaixo estão as principais diferenças:

Aspecto	Relacional	Não Relacional
Modelo de Dados	Tabelas (linhas e colunas)	Documentos, Grafos, Chave-Valor, Colunas
Linguagem	SQL	Varia conforme a ferramenta (MongoDB usa JSON)
Escalabilidade	Vertical (mais recursos)	Horizontal (várias máquinas)
Flexibilidade	Estrutura fixa	Estrutura dinâmica

3. Componentes de um Banco de Dados

- **Tabelas:** Estrutura que armazena os dados. Cada tabela possui linhas e colunas. Exemplo: Tabela **Cientes**.
- **Colunas:** Representam os atributos ou campos da tabela. Exemplo: **Nome**, **Email** e **Telefone**.
- **Linhas:** Cada linha ou registro (tupla) representa um conjunto de dados. Exemplo: Um cliente chamado "João" com seu e-mail e telefone.
- **Índices:** Usados para acelerar a busca de informações. Eles funcionam como um "atalho" para localizar dados.
- **Chaves:** Dividem-se em Chave Primária (PK), que identifica unicamente cada registro, e Chave Estrangeira (FK), que cria relações entre tabelas.

Exemplo Prático:

```
CREATE TABLE Clientes (  
    id INT AUTO_INCREMENT PRIMARY KEY,  
    nome VARCHAR(100),  
    email VARCHAR(100) UNIQUE,  
    telefone VARCHAR(15)  
);
```

No exemplo, `id` é a chave primária e `email` é único para cada cliente.

4. O que é SQL (Structured Query Language)

SQL é a linguagem padrão para interação com bancos de dados relacionais. Com SQL, é possível criar, consultar, atualizar e excluir dados (Date, 2006). Principais comandos SQL:

- **DDL (Data Definition Language):** `CREATE`, `ALTER`, `DROP`
- **DML (Data Manipulation Language):** `INSERT`, `UPDATE`, `DELETE`
- **DQL (Data Query Language):** `SELECT`
- **DCL (Data Control Language):** `GRANT`, `REVOKE`

5. Principais Sistemas de Gerenciamento de Banco de Dados (SGBDs)

Os SGBDs são softwares que permitem gerenciar os bancos de dados. Exemplos:

- **MySQL:** Open source e amplamente usado em aplicações web.
- **PostgreSQL:** Focado em conformidade com o padrão SQL e robustez.
- **Oracle:** Usado em aplicações corporativas de grande porte.
- **SQL Server:** Utilizado por muitas aplicações Microsoft.

6. Instalação e Configuração do MySQL

6.1 Instalação no Windows

1. Baixe o MySQL Installer no site oficial.
2. Execute o instalador e selecione "Developer Default".
3. Configure o servidor MySQL (porta padrão 3306).
4. Defina a senha do usuário `root`.
5. Finalize a instalação e teste a conexão com o MySQL Workbench.

6.2 Instalação no Linux

1. Execute os comandos:

```
sudo apt-get update  
sudo apt-get install mysql-server  
sudo systemctl start mysql
```

2. Execute o assistente de configuração:

```
sudo mysql_secure_installation
```

3. Defina a senha do usuário **root** e ajuste as configurações de segurança.

4. Teste a conexão:

```
mysql -u root -p
```

6.3 Instalação no MacOS

1. Instale o Homebrew (se não estiver instalado):

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/Homebrew/install/HEAD/install.sh)"
```

2. Instale o MySQL:

```
brew install mysql
```

3. Inicie o MySQL:

```
brew services start mysql
```

4. Defina a senha do usuário **root** e teste a conexão:

```
mysql -u root -p
```

2. Introdução ao MySQL

1. O que é o MySQL e suas Vantagens

O MySQL é um sistema de gerenciamento de banco de dados relacional (SGBD) de código aberto. Desenvolvido inicialmente pela MySQL AB e posteriormente adquirido pela Oracle Corporation, ele permite a criação, manipulação e administração de bancos de dados relacionais. Por ser gratuito, leve e eficiente, é amplamente utilizado no desenvolvimento de aplicações web.

Vantagens do MySQL

- **Open Source:** Possui versão gratuita e código aberto.
- **Alta Performance:** Manipula grandes volumes de dados de forma rápida e eficiente.
- **Multiplataforma:** Compatível com Windows, Linux e MacOS.
- **Suporte a Múltiplas Linguagens:** Se integra com linguagens de programação como PHP, Java, Python e outras.
- **Comunidade Ativa:** Grande comunidade de desenvolvedores que contribuem para sua evolução.

2. Arquitetura do MySQL

A arquitetura do MySQL é composta por vários componentes essenciais para o funcionamento do sistema:

- **Conector:** Responsável por permitir a comunicação entre as aplicações e o servidor MySQL.
- **Gerenciador de Consultas:** Analisa, otimiza e executa as consultas SQL enviadas.
- **Motor de Armazenamento:** Onde os dados são fisicamente armazenados. O MySQL suporta diferentes motores, como InnoDB e MyISAM.
- **Buffer de Memória:** Facilita a execução rápida de consultas através do cache de dados.

3. Ferramentas de Interface Gráfica

- **MySQL Workbench:** Ferramenta oficial para administração e modelagem de banco de dados. Permite criar tabelas, executar consultas SQL e visualizar esquemas de forma gráfica.
- **DBeaver:** Ferramenta multiplataforma que suporta vários SGBDs, incluindo MySQL. É amplamente utilizada para realizar consultas e manipulação de dados.
- **phpMyAdmin:** Interface web utilizada para administrar o MySQL diretamente no navegador. É uma das mais utilizadas em ambientes web.

Exemplo Prático (MySQL Workbench)

1. Abra o MySQL Workbench.
2. Clique em "New Connection" e configure os parâmetros de conexão.
3. Insira o nome da conexão, host (localhost), porta (3306), usuário (root) e senha.
4. Clique em "Test Connection" para verificar se a conexão está correta.

4. Conexão com o Banco de Dados MySQL

4.1 Via Terminal

Para conectar ao MySQL via terminal, siga os seguintes passos:

1. Abra o terminal (Linux ou Mac) ou o prompt de comando (Windows).
2. Digite o comando:

```
mysql -u root -p
```

3. Insira a senha do usuário **root** quando solicitado.

Comandos Básicos

- Mostrar bancos de dados:

```
SHOW DATABASES;
```

- Selecionar um banco de dados:

```
USE nome_do_banco;
```

- Mostrar tabelas do banco de dados:

```
SHOW TABLES;
```

4.2 Via Ferramentas Gráficas

MySQL Workbench:

1. Clique em "New Connection".
2. Configure os parâmetros de conexão (nome da conexão, host, porta, usuário e senha).
3. Clique em "Test Connection" para verificar a conexão.

DBeaver:

1. Abra o DBeaver e crie uma nova conexão clicando no ícone de "Nova Conexão".
2. Escolha o MySQL na lista de SGBDs.
3. Configure os parâmetros de conexão (host, porta, usuário e senha).
4. Clique em "Test Connection".

phpMyAdmin:

1. Abra o navegador e acesse o URL onde o phpMyAdmin está instalado.
2. Insira o usuário (root) e a senha configurada durante a instalação.

Atividades de Fixação

1. Explique o que é o MySQL e cite ao menos três de suas vantagens.
2. Qual a diferença entre um banco de dados relacional e um banco de dados não relacional? Dê exemplos práticos de uso.
3. Cite e explique os principais componentes da arquitetura do MySQL.
4. Qual a função do motor de armazenamento do MySQL? Explique a diferença entre InnoDB e MyISAM.
5. Identifique e explique a função de três ferramentas de interface gráfica utilizadas para o MySQL.
6. Descreva o processo de conexão ao MySQL via terminal. Quais são os comandos necessários?
7. Como é possível listar todos os bancos de dados existentes no MySQL? Escreva o comando SQL correspondente.
8. Como criar uma conexão no MySQL Workbench? Liste o passo a passo.
9. O que é e para que serve o comando `USE nome_do_banco;` no MySQL?
10. No MySQL, qual é o comando para listar todas as tabelas de um banco de dados? Escreva o comando e um exemplo prático.
11. Dê um exemplo prático de conexão ao MySQL utilizando o DBeaver. Quais são os parâmetros necessários para essa conexão?
12. Durante a conexão via terminal, o comando `mysql -u root -p` é utilizado. Explique o que cada parte desse comando significa.
13. Cite as principais diferenças entre o MySQL Workbench, o DBeaver e o phpMyAdmin. Qual deles é mais indicado para o uso web?
14. Suponha que você precise garantir que apenas usuários autenticados possam acessar um banco de dados MySQL. Que práticas de segurança você aplicaria?
15. Explique o conceito de buffer de memória no MySQL e sua importância para a performance do banco de dados.

Instruções para a Entrega das Atividades

1. **Elaboração e Envio do Arquivo**
 - Responda todas as questões de forma clara e objetiva.
 - Gere um arquivo no formato **.PDF** contendo as respostas de cada questão.
 - Envie o arquivo para os e-mails dos professores responsáveis.
2. **Validação da Atividade**
 - Após o envio do arquivo, procure o(s) professor(es) para realizar a validação da atividade.
 - **Não inicie a próxima atividade sem antes validar a anterior com o professor.**
3. **Forma de Validação**
 - **Explicação Verbal:** Explique cada resposta verbalmente ao(s) professor(es).
 - **Perguntas e Respostas:** Esteja preparado para responder aos questionamentos do(s) professor(es) sobre o conteúdo das respostas.
 - **Orientação:** Receba orientações sobre a apresentação do(s) tema(s).