

1. O que é o MySQL e suas vantagens

MySQL é um sistema gerenciador de banco de dados relacional (SGBDR) open source. É usado para armazenar, organizar e consultar dados com alta eficiência.

Vantagens:

- Gratuito e de código aberto;
 - Alta performance e escalabilidade;
 - Suporte a transações e integridade dos dados;
 - Grande comunidade e documentação disponível.
-

2. Diferença entre banco de dados relacional e não relacional

- **Relacional:** dados organizados em tabelas (linhas e colunas), usando linguagem SQL.
 - Exemplos: MySQL, PostgreSQL.
 - Uso: dados estruturados com relacionamentos complexos (ex: sistemas financeiros).
 - **Não relacional (NoSQL):** dados em formatos flexíveis, sem esquema fixo.
 - Exemplos: MongoDB (documentos), Redis (chave-valor).
 - Uso: grandes volumes de dados não estruturados, redes sociais, Big Data.
-

3. Principais componentes da arquitetura do MySQL

- **Servidor MySQL:** gerencia conexões, consultas e armazenamento.
- **Cliente:** ferramentas ou aplicativos que enviam comandos ao servidor.
- **Motor de armazenamento:** gerencia como os dados são salvos e recuperados.
- **Parser SQL:** interpreta os comandos SQL.

- **Otimizador de consultas:** escolhe a melhor forma de executar as consultas.
 - **Gerenciador de buffer:** controla a memória usada nas operações.
-

4. Função do motor de armazenamento e diferença entre InnoDB e MyISAM

- **Motor de armazenamento:** componente que armazena dados e gerencia leituras/escritas.
 - **InnoDB:** suporta transações, integridade referencial, bloqueio por linha, é o padrão atual.
 - **MyISAM:** não suporta transações, usa bloqueio por tabela, rápido em leituras simples.
-

5. Três ferramentas de interface gráfica para MySQL

- **MySQL Workbench:** ferramenta oficial para modelagem, desenvolvimento e administração.
 - **phpMyAdmin:** interface web para gerenciar MySQL via navegador.
 - **DBeaver:** cliente universal com interface para múltiplos bancos.
-

6. Conexão ao MySQL via terminal e comandos necessários

- Abra o terminal.
 - Digite: `mysql -u usuario -p` (substitua *usuario* pelo nome do usuário).
 - Insira a senha quando solicitado.
 - Execute comandos SQL normalmente.
-

7. Comando para listar todos os bancos de dados no MySQL

- SHOW DATABASES;
-

8. Passo a passo para criar uma conexão no MySQL Workbench

- Abra o MySQL Workbench.
 - Clique em **MySQL Connections** e depois no botão **+** para nova conexão.
 - Informe nome da conexão, host, porta, usuário e senha.
 - Teste a conexão e salve.
-

9. Função do comando **USE nome_do_banco;**

Seleciona o banco de dados para o contexto das próximas consultas.

10. Comando para listar todas as tabelas de um banco de dados e exemplo

- SHOW TABLES;

Exemplo:

- USE clientes;
 - SHOW TABLES;
-

11. Exemplo prático de conexão ao MySQL usando DBeaver e parâmetros necessários

- Abra o DBeaver.
- Crie nova conexão e selecione MySQL.

- Informe host, porta, banco, usuário e senha.
 - Teste e salve.
-

12. Explicação do comando `mysql -u root -p`

- `mysql`: executa o cliente MySQL no terminal.
 - `-u root`: conecta com o usuário root.
 - `-p`: solicita senha para autenticação.
-

13. Diferenças entre MySQL Workbench, DBeaver e phpMyAdmin; qual é mais indicado para uso web

- **MySQL Workbench**: desktop oficial para administração e desenvolvimento.
 - **DBeaver**: cliente universal desktop para vários bancos.
 - **phpMyAdmin**: interface web para administração via navegador.
 - Para uso web, **phpMyAdmin** é o mais indicado.
-

14. Práticas de segurança para garantir acesso apenas a usuários autenticados

- Criar usuários com senhas fortes.
- Conceder privilégios mínimos necessários.
- Usar criptografia SSL/TLS nas conexões.
- Configurar firewall e restrição por IP.
- Manter sistema atualizado.
- Monitorar e auditar acessos.

15. Conceito de buffer de memória no MySQL e importância para performance

Buffer de memória é uma área da RAM para armazenar dados temporariamente.

Reduz acessos ao disco e acelera leituras e escritas, melhorando a performance, principalmente em consultas frequentes.