REVISÃO INTERNET E PROGRAMAÇÃO WEB
II UNIDADE

Professor: Marcos Brandão

Versão: 2026

SUMÁRIO

---

1. INTERNET E PROGRAMAÇÃO WEB

A Programação Web é utilizada para desenvolver sistemas acessados por navegadores na Internet.

Exemplos:

- Sites institucionais
- Sistemas acadêmicos
- Lojas virtuais
- Sistemas de gerenciamento
- Aplicações corporativas

---

2. PÁGINAS WEB DINÂMICAS

Uma página dinâmica gera conteúdo de acordo com as informações armazenadas em banco de dados ou enviadas pelos usuários.

Exemplo

Um sistema de cadastro de alunos pode exibir informações diferentes para cada usuário.

Vantagens

- Atualização automática dos dados
- Interatividade
- Integração com banco de dados
- Personalização

---

3. LINGUAGENS DE PROGRAMAÇÃO PARA WEB

Principais tecnologias:

Front-end

Executadas no navegador.

- HTML
- CSS
- JavaScript

Back-end

Executadas no servidor.

- PHP
- Python
- Java
- Node.js
- C#

---

4. BANCO DE DADOS MARIADB

O MariaDB é um Sistema Gerenciador de Banco de Dados (SGBD) relacional.

Foi criado como alternativa ao MySQL.

Para que serve?

- Armazenar dados
- Organizar informações
- Permitir consultas rápidas
- Controlar usuários e permissões

Exemplos de uso

- Sistemas acadêmicos
- E-commerce
- Sistemas financeiros
- Aplicações web

---

5. COMANDOS BÁSICOS DO MARIADB

Mostrar bancos de dados

SHOW DATABASES;

Exibe todos os bancos existentes no servidor.

Selecionar um banco

USE escola;

Define o banco de dados que será utilizado.

Descrever uma tabela

DESCRIBE alunos;

ou

DESC alunos;

Mostra:

- Campos
- Tipos de dados
- Chaves
- Restrições

---

6. CONSULTAS COM SELECT

O comando SELECT é utilizado para consultar dados.

Exemplo

SELECT * FROM alunos;

Exibe todos os registros.

Selecionar colunas específicas

SELECT nome, curso
FROM alunos;

Alias com AS

Permite renomear colunas.

SELECT nome AS Aluno
FROM alunos;

Resultado:

Aluno
João
Maria

---

7. FILTROS COM WHERE

Utilizado para selecionar registros específicos.

Exemplo

```
SELECT *
FROM alunos
WHERE curso = 'Informática';
```

Outro exemplo

```
SELECT *
FROM produtos
WHERE preco > 100;
```

---

8. ORDENAÇÃO E LIMITAÇÃO

ORDER BY

Ordena os resultados.

Crescente

```
SELECT *
FROM alunos
ORDER BY nome ASC;
```

Decrescente

```
SELECT *
FROM alunos
ORDER BY nome DESC;
```

LIMIT

Limita a quantidade de registros retornados.

Exemplo

```
SELECT *
FROM alunos
LIMIT 5;
```

Retorna apenas os 5 primeiros registros.

---

9. FUNÇÕES DE AGREGAÇÃO

São utilizadas para realizar cálculos sobre conjuntos de dados.

COUNT()

Conta registros.

```
SELECT COUNT(*)
FROM alunos;
```

SUM()

Realiza somatórios.

```
SELECT SUM(salario)
FROM funcionarios;
```

AVG()

Calcula média.

```
SELECT AVG(nota)
FROM alunos;
```

---

10. INNER JOIN

Utilizado para relacionar tabelas.

Exemplo

Tabela alunos

id	nome

1	João

Tabela matriculas

aluno_id	curso
1	Redes

Consulta:

```
SELECT alunos.nome, matriculas.curso
FROM alunos
INNER JOIN matriculas
ON alunos.id = matriculas.aluno_id;
```

Resultado:

nome	curso
João	Redes

---

11. ALTERAÇÃO E MANIPULAÇÃO DE DADOS

ALTER TABLE

Modifica a estrutura da tabela.

Adicionar coluna

```
ALTER TABLE alunos
ADD telefone VARCHAR(20);
```

UPDATE

Atualiza registros.

```
UPDATE alunos
SET curso = 'Redes'
```

WHERE id = 1;

DELETE

Remove registros.

DELETE FROM alunos
WHERE id = 1;

⚠ Atenção:

Sem WHERE todos os registros serão removidos.

Exemplo perigoso:

DELETE FROM alunos;

---

12. BOAS PRÁTICAS EM BANCO DE DADOS

- Sempre utilizar WHERE em UPDATE e DELETE.
- Fazer backup regularmente.
- Utilizar nomes padronizados para tabelas.
- Evitar armazenar informações duplicadas.
- Utilizar relacionamentos entre tabelas.
- Testar consultas antes de executá-las em produção.
