**ROTEIRO DE ATIVIDADE DE LABORATÓRIO**No. LAB 02

Disciplina: 12902P - DESENVOLVIMENTO DE SISTEMAS WEB  
Professor: José Matias Lemes Filho

**Título:** Explorando *tags* HTML básicas e avançadas

**Objetivo:**   
  
O objetivo deste laboratório é familiarizar os alunos com uma variedade de *tags* HTML, desde as mais básicas até as mais avançadas, explorando suas funcionalidades e aplicabilidade no desenvolvimento de páginas web. Os alunos criarão uma página web simples utilizando as *tags* discutidas em aula.

**Materiais Necessários:**

• Computadores com acesso à internet

• Editor de texto (recomendado: Visual Studio Code)

• Navegador web (Chrome, Firefox, etc.)

**Descrição da Atividade:**

Neste laboratório, os alunos irão desenvolver uma página web que inclui:

1. Cabeçalhos de diferentes níveis.

2. Parágrafos e listas (ordenadas e não ordenadas).

3. Formulários com diferentes tipos de campos de entrada.

4. Elementos gráficos usando <canvas> e <svg>.

5. Incorporação de mídia, como áudio e vídeo.

**Instruções:**

**1. Estrutura Básica da Página**

• Crie um arquivo HTML chamado lab02.html.

• Insira a estrutura básica de um documento HTML5 com as *tags* <html>, <head>, <title>, e <body>.

**2. Cabeçalhos e Parágrafos**

• Dentro da *tag* <body>, crie cabeçalhos utilizando as *tags* <h1> a <h6>, cada um com um texto apropriado.

• Adicione alguns parágrafos (<p>) para complementar os cabeçalhos.

**3. Listas**

• Crie uma lista não ordenada (<ul>) e uma lista ordenada (<ol>) com pelo menos três itens cada, utilizando a *tag* <li> para definir os itens da lista.

**4. Formulários**

• Adicione um formulário (<form>) com os seguintes elementos:

• Campo de texto com um rótulo (<label>) para o nome do usuário.

• Menu suspenso (<select>) para escolher a fruta favorita, com opções como Maçã, Banana e Laranja.

• Área de texto (<textarea>) para que o usuário possa inserir comentários adicionais.

• Botão de envio (<input type="submit">).

**5. Elementos Gráficos**

• Utilize a *tag* <canvas> para desenhar um retângulo colorido.

• Insira um gráfico vetorial simples utilizando a *tag* <svg>, como um círculo ou retângulo.

**6. Incorporação de Mídia**

• Adicione um áudio (<audio>) com controles de reprodução.

• Incorpore um vídeo (<video>) que os usuários possam assistir diretamente na página.

**Entrega:**

• Salve o arquivo lab02.html e faça o upload para a plataforma de submissão conforme as instruções do professor.

• O código deve estar formatado de maneira clara e organizado.

**Avaliação:**

• A atividade será avaliada com base nos seguintes critérios:

• Correta utilização das *tags* HTML discutidas.

• Organização e clareza do código.

• Funcionalidade da página criada.

**A Evolução do Desenvolvimento Web**

Ao longo dos anos, a linguagem HTML ganhou ampla aceitação no meio acadêmico e rapidamente se difundiu para o setor empresarial. Em 1995, foi fundado o W3C (World Wide Web Consortium), uma entidade responsável por supervisionar e estabelecer os padrões da Web, assegurando a evolução e consistência da linguagem HTML.

**Versão 2.0 (1995**): Lançada junto com a criação do W3C, esta versão tinha como principal objetivo formalizar todas as características da linguagem HTML que já estavam em uso. Foi o primeiro passo para a padronização do HTML na Web.

**Versão 3.2 (1997):** Esta versão corrigiu problemas de compatibilidade que existiam na versão anterior e introduziu novas funcionalidades, como a tag `<table>`, que permitiu a criação de tabelas, ampliando significativamente as capacidades de formatação e organização de dados nas páginas web.

**Versão 4.01 (1999):** Com o HTML 4.01, o suporte para multimídia foi ampliado, tornando a web mais rica e interativa. Além disso, essa versão introduziu o CSS (Cascading Style Sheets), uma poderosa ferramenta para separar o conteúdo da apresentação, permitindo maior controle sobre o design das páginas.

**Versão 5 (2014):** O HTML5 é a versão mais recente e revolucionária da linguagem, trazendo uma série de inovações que transformaram a maneira como interagimos com a web. Algumas das principais melhorias incluem:

* Suporte para banco de dados na Web: permitindo que aplicações web armazenem dados localmente, sem depender exclusivamente do cache do navegador.
* Remoção do núcleo SGML: que deu ao HTML maior autonomia, permitindo uma evolução mais ágil e a incorporação de novas funcionalidades.
* Formulários aprimorados: com novos tipos de entrada, como data/hora, e-mail e campos de pesquisa, melhorando a experiência do usuário e a funcionalidade dos formulários.

A evolução do HTML reflete o crescimento e a complexidade crescente da web, tornando-se uma ferramenta essencial para o desenvolvimento de aplicações e a criação de experiências digitais mais dinâmicas e envolventes.

**Explicação das Tags HTML**

**1. `<header>`: Define o cabeçalho de uma seção ou página, geralmente utilizado para incluir títulos, logotipos, e elementos de navegação.**

**2. `<nav>`: Identifica uma seção de navegação, que contém links para outras partes da página ou do site.**

**3. `<main>`: Representa o conteúdo principal de uma página, diferenciando-o de outras partes como cabeçalhos, rodapés ou barras laterais.**

**4. `<article>`: Indica um conteúdo independente e autossuficiente, como um artigo, post de blog, ou notícia.**

**5. `<section>`: Define uma seção temática de um documento, utilizada para agrupar conteúdo relacionado.**

**6. `<aside>`: Representa conteúdo complementar ou relacionado, como barras laterais ou caixas de informações adicionais.**

**7. `<footer>`: Define o rodapé de uma seção ou página, onde normalmente são inseridas informações de contato, direitos autorais ou links secundários.**

**8. `<h1>` a `<h6>`: Representam cabeçalhos de diferentes níveis, onde `<h1>` é o mais importante e `<h6>` o menos importante, utilizados para hierarquizar títulos e subtítulos.**

**9. `<p>`: Define um parágrafo de texto, o bloco básico de conteúdo textual.**

**10. `<a>`: Cria um link para outra página, recurso ou seção do site. O atributo `href` especifica o destino do link.**

**11. `<img>`: Insere uma imagem na página. O atributo `src` define o caminho da imagem, e `alt` fornece um texto alternativo caso a imagem não seja carregada.**

**12. `<video>`: Permite a incorporação de vídeos na página, com suporte para controles de reprodução como play, pause e volume.**

**13. `<audio>`: Insere arquivos de áudio na página, com opções para controles de reprodução.**

**14. `<ul>`: Cria uma lista não ordenada, onde os itens são exibidos com marcadores (bullet points).**

**15. `<ol>`: Cria uma lista ordenada, onde os itens são numerados ou identificados por letras.**

**16. `<li>`: Define um item de lista, utilizado dentro das tags `<ul>` ou `<ol>`.**

**17. `<table>`: Cria uma tabela, utilizada para organizar dados em linhas e colunas.**

**18. `<form>`: Define um formulário para entrada de dados, permitindo que os usuários enviem informações para o servidor.**

**19. `<input>`: Cria um campo de entrada em um formulário, como uma caixa de texto, botão, caixa de seleção, entre outros.**

**20. `<textarea>`: Cria uma área de texto multilinha, permitindo a entrada de grandes quantidades de texto.**

**21. `<label>`: Fornece um rótulo para um elemento de formulário, melhorando a acessibilidade e a usabilidade.**

**22. `<select>`: Cria um menu suspenso que permite ao usuário selecionar uma opção de uma lista.**

**23. `<canvas>`: Permite a criação de gráficos e animações diretamente no navegador utilizando JavaScript.**

**24. `<svg>`: Insere gráficos vetoriais escaláveis, permitindo criar formas como círculos, retângulos e linhas que podem ser dimensionados sem perder qualidade.**

**25. `<div>`: Define uma divisão genérica ou um contêiner, utilizado para agrupar elementos e aplicar estilos em conjunto.**

**26. `<span>`: Define uma divisão em linha, utilizada para aplicar estilos a uma parte específica do texto sem quebrar a linha.**

**Explicação dos Atributos HTML**

**1. `id`: Atribui um identificador exclusivo a um elemento HTML, permitindo referenciá-lo em estilos CSS ou scripts JavaScript.**

**2. `class`: Define uma classe para um elemento, permitindo agrupar elementos similares para aplicar estilos e comportamentos em conjunto.**

**3. `style`: Permite definir estilos CSS diretamente em um elemento específico, aplicando regras como cor, fonte, tamanho, entre outros.**

**4. `src`: Especifica o caminho para a origem de um recurso, como uma imagem, vídeo, ou script.**

**5. `alt`: Fornece um texto alternativo para imagens, que será exibido se a imagem não puder ser carregada, além de melhorar a acessibilidade.**

**6. `width` e `height`: Define a largura e altura de um elemento, como uma imagem ou vídeo, em pixels.**

**7. `href`: Especifica o destino de um link, podendo ser outra página, um arquivo, ou um endereço de email.**

**8. `target`: Define onde o link será aberto, por exemplo, em uma nova janela (`\_blank`) ou na mesma janela (`\_self`).**

**9. `rel`: Define o relacionamento entre a página atual e o recurso vinculado, sendo comum em links para folhas de estilo (`rel="stylesheet"`) ou ícones (`rel="icon"`).**

**10. `disabled`: Desabilita a interação com um elemento, como um botão ou campo de formulário, tornando-o inativo.**

**11. `required`: Indica que um campo de formulário é obrigatório e deve ser preenchido antes de o formulário ser enviado.**

**12. `placeholder`: Exibe um texto de exemplo dentro de um campo de entrada, sugerindo o tipo de informação que deve ser fornecida.**

**13. `autoplay`: Especifica se um elemento de áudio ou vídeo deve começar a ser reproduzido automaticamente ao carregar a página.**

**14. `loop`: Indica se um elemento de áudio ou vídeo deve ser reproduzido repetidamente de forma contínua.**

**15. `download`: Força o download de um recurso quando um link é clicado, em vez de exibi-lo diretamente no navegador.**

**16. `contenteditable`: Permite que o conteúdo de um elemento seja editável pelo usuário diretamente no navegador.**

**17. `aria-\*`: Atributos ARIA (Accessible Rich Internet Applications) fornecem informações adicionais para tecnologias assistivas, melhorando a acessibilidade para usuários com deficiências.**