

Resumo Estudos Prova 2 2025-2

1) Estrutura Básica e Elementos de HTML

- **Estrutura Essencial**

- Todo documento HTML deve incluir:
 - <!DOCTYPE html>: Define o documento como HTML 5.
 - <html>: O elemento raiz do documento.
 - <head>: Contém metadados e informações para o navegador.
Elementos importantes aqui são: <meta charset="utf-8"> (codificação de caracteres, essencial para acentos e cedilha) e <title> (define o título exibido na aba do navegador).
 - <body>: Contém o conteúdo visível da página.

- **Elementos Principais do Corpo (<body>)**

- Títulos (<h1> a <h6>): Representam os níveis hierárquicos de seção, sendo <h1> o nível mais alto e importante.
- Parágrafos (<p>): Elementos de bloco usados para agrupar conteúdo textual relacionado.
- Divisão (<div>) e Span (): Containers genéricos. <div> é um elemento de bloco (ocupa toda a largura), enquanto é um elemento inline (acompanha o fluxo do texto).
- Formatação de Texto: Inclui tags para negrito (ou) e ênfase ().
- Linha Horizontal (<hr>) e Texto animado (legado) (<marquee>).
- Imagens, Listas e Links
- Imagens (): Usa o atributo src para especificar o caminho da imagem e alt para fornecer texto alternativo (importante para acessibilidade e mecanismos de busca).
- Listas:
 - Não Ordenadas (): Usadas para itens sem ordem específica (utilizam para cada item). Podem usar type ("disk", "circle", "square").
 - Ordenadas (): Usadas para listas sequenciais (como um passo-a-passo). Aceitam atributos como type (para letras ou algarismos romanos) e start (para definir o valor inicial).
- Links (<a>): Criam hiperligações usando o atributo href (destino). target="_blank" abre o link em uma nova aba. Links podem ser internos (para outra parte da mesma página, usando #id) ou externos.

- **Tabelas**

- Tabelas são construídas com <table> e estruturadas por:
<thead>: Cabeçalho
<tbody>: Corpo
<tfoot>: Rodapé
<tr>: Linhas
<th> e <td>: Células de cabeçalho e de dados, respectivamente. colspan e rowspan são usados para agrupar células.

2) Formulários HTML

- **Estrutura e Elementos de Formulário**
 - <form>: É o container para os elementos do formulário.
 - Inputs (<input>): Elementos de entrada genéricos definidos pelo atributo type. Tipos comuns incluem:
 - text, email, password
 - radio: Para seleção única em um grupo (definido pelo atributo name)
 - checkbox: Para seleção múltipla
 - number, date, time
 - submit (botão de envio) e button (botão genérico).
 - <label>: Rótulos usados para descrever os campos. Devem ser vinculados ao campo correspondente pelo atributo for (igual ao id do input).
 - <textarea>: Campo de texto com múltiplas linhas (atributos rows e cols definem o tamanho).
 - <select> e <option>: Criam menus de opções (combobox). O atributo selected define a opção padrão. O atributo multiple permite múltiplas seleções.
- **Agrupamento e Atributos Importantes**
 - <fieldset> e <legend>: Usados para agrupar visualmente elementos relacionados no formulário. <legend> fornece o título para o grupo.
- **Atributos de Validação/Comportamento:**
 - name: Essencial para identificar o campo no envio dos dados.
 - value: Define o valor padrão ou o valor a ser enviado.
 - checked: Indica que um radio button ou checkbox está selecionado por padrão.
 - required: Torna o campo obrigatório (validação HTML5).
 - placeholder: Texto de orientação que aparece dentro do campo.
 - min e max: Definem o intervalo mínimo e máximo de valores aceitos (ex: type="number").

3) Introdução e Seletores CSS

- **Conceito e Sintaxe CSS**
 - Função: O CSS (Cascading Style Sheets) define a apresentação visual (formatação, layout, cores, fontes) do documento HTML.
 - Regra CSS: É composta por um seletor e um bloco de declarações (propriedade: valor;).
 - Propriedades Comuns: color (cor do texto), background-color, font-family (fonte), font-size, border.
- **Formas de Aplicação de Estilos**
 - Existem três formas de aplicar CSS em um documento HTML:
 - Estilos Inline: Usando o atributo style diretamente no elemento HTML (prática geralmente evitada).
 - Folha de Estilos Interna: Usando a tag <style> dentro do <head> do documento.
 - Folha de Estilos Externa: Usando um arquivo .css separado e vinculado via tag <link> no <head> (método mais recomendado para organização e reutilização).

- **Seletores**

- Seletores permitem direcionar regras CSS a elementos específicos:
 - Seletor por Tagname (Tipo): Aplica o estilo a todos os elementos daquela tag (ex: p { ... }).
 - Seletor por ID (#): Seleciona o elemento único com o ID especificado (ex: #principal { ... }).
 - Seletor por Classe (.): Seleciona todos os elementos com a classe especificada (ex: .destaque { ... }). Um elemento pode ter múltiplas classes.
 - Seletor Universal (*): Aplica estilos a todos os elementos.
- Seletores Hierárquicos:
 - Descendente (Espaço): Seleciona elementos que estão dentro de outro, em qualquer nível (ex: div p { ... }).
 - Filho Direto (>): Seleciona apenas os filhos diretos (ex: div > p { ... }).
 - Pseudo-classes (:): Selecionam elementos baseados em seu estado ou condição (ex: :hover para quando o cursor está sobre o elemento).

4) Layout CSS (Box Model e Flexbox)

- **Tipos de Elementos**

- Block Box (Caixa de Bloco): Ocupam 100% da largura disponível e não aceitam outros elementos ao lado (ex: <div>, <p>).
- Inline Box (Caixa em Linha): Ocupam apenas a largura necessária e permitem que outros elementos fiquem na mesma linha (ex: ,).
- Propriedade display: Permite alterar o comportamento padrão, como definir display: inline-block.

- **Box Model**

- O modelo de caixa define a geometria do elemento, composto por:
 - Conteúdo
 - padding: Espaçamento interno (distância entre o conteúdo e a borda).
 - border: Borda do elemento.
 - margin: Margem externa (espacamento entre elementos).
 - box-sizing: border-box: Altera a forma como a largura e altura são calculadas para incluir o padding e a border.

- **Posicionamento Tradicional**

- float: Usado para tirar um elemento do fluxo normal e fazê-lo flutuar à esquerda ou à direita, permitindo que o conteúdo flua ao redor (base para layouts de múltiplas colunas).
- clear: Usado para interromper o efeito do float, garantindo que o elemento comece abaixo de qualquer flutuante (clear: both;).
- position: Define o esquema de posicionamento (static, relative, absolute). Elementos com position: absolute são posicionados em relação ao ancestral mais próximo que não tenha position: static.

- **Flexbox Layout**

- O Flexbox (display: flex) é um modelo de layout para organizar itens em uma dimensão (linha ou coluna). As propriedades são aplicadas no container (elemento pai) e nos itens (elementos filhos):
 - Container Propriedades:
 - display: flex: Ativa o Flexbox.

- flex-direction: Define a direção (ex: row ou column).
- justify-content: Alinha itens no eixo principal (ex: space-around, space-between).
- align-items: Alinha itens no eixo cruzado (perpendicular) (ex: center, flex-start).
- Item Propriedades:
 - flex: Define a proporção de espaço que o item ocupa dentro do container.
 - align-self: Sobrescreve align-items para um item individual.