



AULA 02

INTRODUÇÃO AO HTML E

ESTRUTURAS BÁSICAS



1. O que é HTML?

O HTML (HyperText Markup Language) é a linguagem padrão para a criação de páginas web. Ele define a estrutura e o conteúdo de um site, organizando os elementos da página em forma de marcação.

O HTML não é uma linguagem de programação, pois não possui lógica de programação como o JavaScript. Ele apenas estrutura e exibe informações, funcionando como a base de qualquer página web.

📌 Origem e Evolução do HTML

O HTML foi criado por Tim Berners-Lee no início dos anos 90, como parte da World Wide Web (www). Desde então, evoluiu por meio de diversas versões, até chegar ao HTML5, que é o padrão atual e trouxe avanços como novas tags semânticas e suporte a áudio e vídeo sem plugins externos.

💡 Exemplo: Imagine um site como uma casa:

- O HTML é a estrutura (paredes, portas, janelas).
- O CSS é a decoração (cores, móveis, iluminação).
- O JavaScript é a parte interativa (interruptores de luz, campainha, portas automáticas).

2. Estrutura Básica de um Documento HTML5

Todo documento HTML segue uma estrutura padrão, que contém os elementos essenciais para o funcionamento da página. Veja a estrutura básica abaixo:

📌 Código: Estrutura Padrão do HTML5

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Minha Primeira Página HTML</title>
</head>
<body>
  <h1>Olá, Mundo!</h1>
  <p>Essa é a minha primeira página HTML.</p>
</body>
</html>
```

2. Estrutura Básica de um Documento HTML5

```
<!DOCTYPE html>
<html lang="pt-BR">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Minha Primeira Página HTML</title>
</head>
<body>
  <h1>Olá, Mundo!</h1>
  <p>Essa é a minha primeira página HTML.</p>
</body>
</html>
```

Explicação dos Elementos:

- <!DOCTYPE html> → Indica que o documento usa HTML5.
- <html lang="pt-BR"> → Define o idioma da página como português do Brasil.
- <head> → Contém informações sobre a página (metadados, título, links para CSS e scripts).
- <meta charset="UTF-8"> → Define a codificação de caracteres (suporte a acentos e caracteres especiais).
- <meta name="viewport" content="width=device-width, initial-scale=1.0"> → Garante que a página seja responsiva em dispositivos móveis.
- <title> → Define o título exibido na aba do navegador.
- <body> → Contém o conteúdo visível da página, como textos, imagens e links.

3. Tags Fundamentais de Conteúdo

O HTML é composto por tags, que são comandos utilizados para definir diferentes tipos de conteúdo na página. Algumas das tags mais utilizadas incluem:

📌 Cabeçalhos (<h1> a <h6>)

Os cabeçalhos definem títulos e subtítulos em diferentes níveis de importância.



```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Document</title>
7    </head>
8    <body>
9      <h1>Este é um título principal</h1>
10     <h2>Este é um subtítulo</h2>
11     <h3>Este é um título menor</h3>
12     <h4>Este é um título menor ainda</h4>
13     <h5>E assim por diante...</h5>
14     <h6>Este é o menor título possível</h6>
15   </body>
16 </html>
17
```

3. Tags Fundamentais de Conteúdo

📌 Parágrafos (<p>)

A tag <p> é usada para criar parágrafos de texto.



```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Document</title>
7  </head>
8  <body>
9      <p>Isso é uma tag de paragrafo e posso alterar o conteúdo</p>
10 </body>
11 </html>
```

3. Tags Fundamentais de Conteúdo

📌 Quebras de linha (
) e espaçamentos

A tag
 insere uma quebra de linha dentro do texto.



```
1  <!DOCTYPE html>
2  <html lang="en">
3    <head>
4      <meta charset="UTF-8" />
5      <meta name="viewport" content="width=device-width, initial-scale=1.0" />
6      <title>Document</title>
7    </head>
8    <body>
9      <p>
10        Este é um texto antes da quebra de linha.<br />Agora estamos em uma nova
11        linha.
12      </p>
13    </body>
14  </html>
15
```

4. Boas Práticas de Organização do Código

Escrever um código organizado é essencial para facilitar a manutenção e a leitura. Algumas boas práticas incluem:

📌 Indentação e Legibilidade

A indentação (recurso das linhas) torna o código mais legível. Exemplo correto:



```
1  <!DOCTYPE html>
2  <html lang="pt-BR">
3      <head>
4          <meta charset="UTF-8" />
5          <title>Exemplo de Indentação</title>
6      </head>
7      <body>
8          <h1>Bem-vindo!</h1>
9          <p>Este é um parágrafo formatado corretamente.</p>
10         </body>
11     </html>
12
```

4. Boas Práticas de Organização do Código

📌 Comentários em HTML (<!-- ... -->)

Os comentários ajudam a documentar o código sem afetar a exibição da página.

📌 Estrutura limpa para facilitar a manutenção

- Sempre utilize títulos e parágrafos para organizar o conteúdo.
- Evite misturar HTML com CSS e JavaScript dentro do mesmo arquivo.
- Comente partes importantes do código para facilitar futuras alterações.



```
1  <!DOCTYPE html>
2  <html lang="pt-BR">
3    <head>
4      <meta charset="UTF-8" />
5      <title>Exemplo de Indentação</title>
6    </head>
7    <body>
8      <!-- Este é um comentário -->
9      <p>Texto visível na página.</p>
10     </body>
11   </html>
12
```

5. Exercícios Práticos

📌 Exercício 1 - Criando uma Página Simples

Crie um arquivo HTML chamado index.html e insira os seguintes elementos:

- ✓ Um título (`<title>`) na aba do navegador.
- ✓ Um cabeçalho (`<h1>`) com o nome da sua página.
- ✓ Dois subtítulos (`<h2>` ou `<h3>`).
- ✓ Dois parágrafos (`<p>`) descrevendo qualquer tema de interesse.
- ✓ Um comentário (`<!-- ... -->`) explicando o objetivo da página.

📌 Exercício 2 - Revisão e Melhoria do Código

Agora, revise seu código e adicione:

- ✓ Mais cabeçalhos e parágrafos para organizar o conteúdo.
- ✓ Utilize indentação correta para melhorar a legibilidade.
- ✓ Inclua comentários para descrever partes do código.

OBRIGADO!



Prof. Rodrigo Medeiros - Curso Programação Full-Stack