

HELLO, GIRLS!

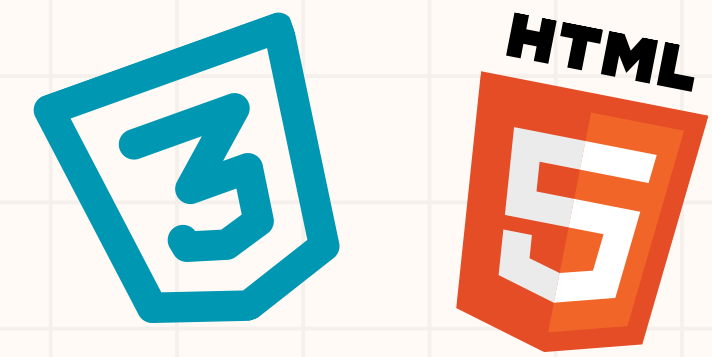
AULA 3 - História da HTML e CSS, estrutura e boas práticas

OBJETIVO

Compreender os conceitos básicos do desenvolvimento Front-end, identificando a função do HTML na estruturação do conteúdo e do CSS na estilização da página, além de aplicar boas práticas de organização do código para criar páginas mais claras, funcionais e de fácil manutenção.



FRONT-END



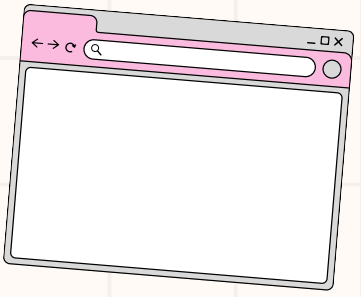
O Front-end é a parte de um site ou aplicação que o usuário consegue ver e interagir. Tudo aquilo que aparece na tela do navegador, como textos, imagens, botões, formulários, menus e animações, faz parte do front-end.

Para funcionar, o front-end utiliza principalmente três tecnologias: o HTML, que cria a estrutura do conteúdo; o CSS, que define a aparência e o estilo da página; e o JavaScript, que adiciona interatividade e dinamismo, como menus que se abrem, carrosséis de imagens ou validações de formulários.

Assim, o front-end tem como principal objetivo criar interfaces agradáveis, responsivas e funcionais, tornando a experiência do usuário mais intuitiva e eficiente.



O QUE É HTML?



HTML é a sigla para HyperText Markup Language, que por sua vez significa Linguagem de Marcação de Hipertexto.

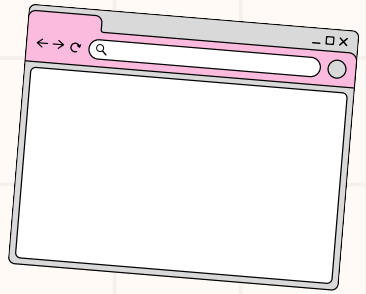
O HTML foi criado por Tim Berners-Lee, como uma ferramenta que permitia acrescentar alguns elementos nos textos usados. Foi a linguagem que permitiu o desenvolvimento dos primeiros sites.

Ele é o responsável por mostrar ao navegador como um conteúdo deve estar estruturado. Se a página tem imagens, é ele que especifica onde e como elas serão apresentadas.

Não é exagero dizer, que boa parte do que temos hoje, deve-se ao surgimento e desenvolvimento da linguagem ao longo de sua história.



O QUE É CSS?



CSS é a sigla para Cascading Style Sheets, traduzido como Folhas de Estilo em Cascata.

Ele foi desenvolvido para complementar o HTML, dando aos desenvolvedores a possibilidade de controlar o visual e o design das páginas da web. Com o CSS, é possível definir cores, fontes, tamanhos de texto, margens, alinhamentos e até criar animações e transições.

Uma de suas maiores vantagens é a separação entre conteúdo e apresentação, o que facilita a manutenção e permite que várias páginas compartilhem o mesmo estilo.

Pode-se dizer que o CSS transformou a web em algo muito mais dinâmico e atraente, sendo essencial para a criação de sites modernos e responsivos.



ESTRUTURA BÁSICA DE UMA PÁGINA

```
<!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</title>
</head>

<body>
  <h1>Hello, girls!</h1>
  <p>Este é um parágrafo de exemplo.</p>
</body>

</html>
```



O QUE SÃO TAGS

Em HTML (HyperText Markup Language), as tags são os elementos que dão estrutura e significado ao conteúdo de uma página da web.

Elas são palavras-chave delimitadas por sinais de menor < e maior >, que dizem ao navegador como aquele conteúdo deve ser exibido.

Exemplo:

<p> → tag de abertura (parágrafo)

</p> → tag de fechamento

Algumas tags são autofechadas (não precisam de fechamento), como:

 (quebra de linha)

ESTRUTURA BÁSICA DE UMA PÁGINA

`<!DOCTYPE html>`

➡ Informa ao navegador que o documento é um arquivo HTML5, que é a versão mais atual do HTML.

`<html lang="pt-BR">`

➡ Abre a tag principal do HTML e define o idioma da página como português do Brasil.

`<head>` e `</head>`

➡ Contém informações de configuração da página, que não aparecem diretamente para o usuário, mas são importantes para o navegador.

`<meta charset="UTF-8">`

➡ Define a codificação de caracteres da página, permitindo que acentos (á, é, ã, ç) e símbolos sejam exibidos corretamente.

`<meta name="viewport" content="width=device-width, initial-scale=1.0">`

➡ Faz a página se adaptar a diferentes tamanhos de tela (computador, celular, tablet), tornando-a responsiva.

`<title>Minha Primeira Página</title>`

➡ Define o título da página, que aparece na aba do navegador.

`<body>` e `</body>`

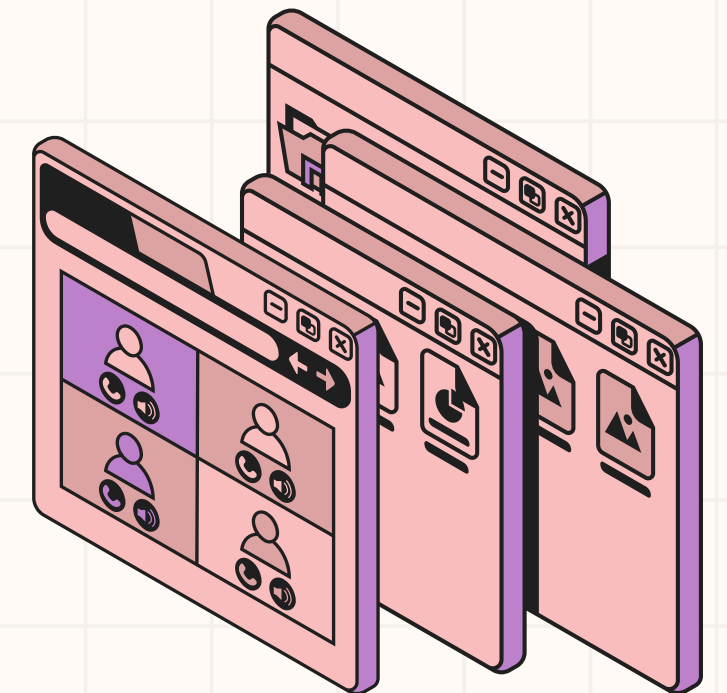
➡ É onde fica o conteúdo visível da página, ou seja, o que o usuário realmente vê na tela.

`<h1>Hello, girls!</h1>`

➡ Um título de nível 1 (o mais importante). Geralmente usado para títulos principais da página.

`<p>Este é um parágrafo de exemplo.</p>`

➡ Um parágrafo de texto comum, usado para conteúdo descritivo.



HIERARQUIA DE TÍTULOS

Em HTML, os títulos (chamados de headings) vão de <h1> até <h6>. Eles são usados para organizar e hierarquizar o conteúdo da página, como em um índice.

<h1>Título principal</h1>

<h2>Subtítulo</h2>

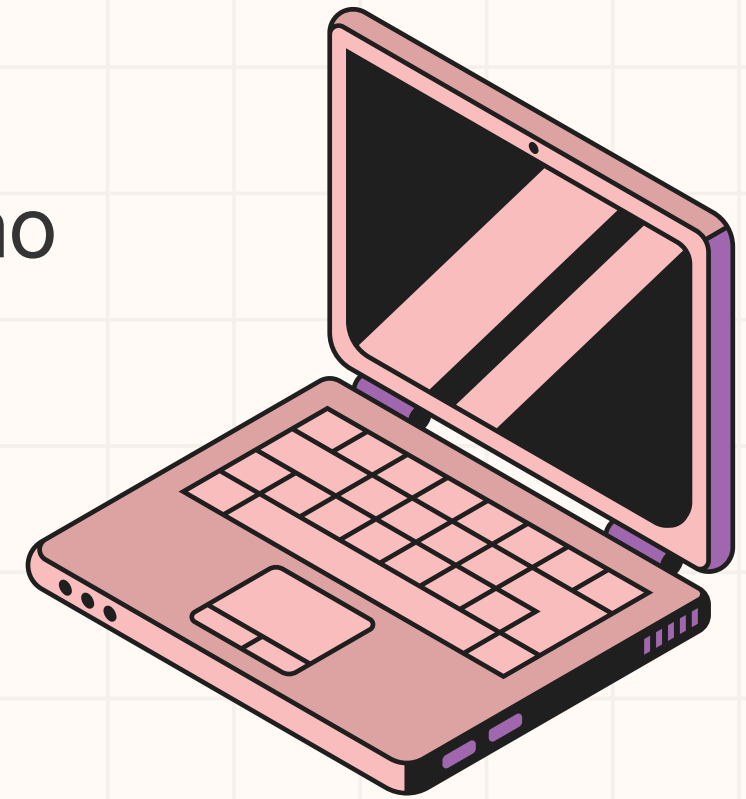
<h3>Seção dentro do subtítulo</h3>

<h4>Subseção</h4>

<h5>Detalhe dentro da subseção</h5>

<h6>Nível mais específico</h6>

O h1 deve ser único, ou seja, é recomendado usar apenas uma vez na página



QUEBRAS DE LINHA E COMENTÁRIOS

No HTML, o navegador ignora espaços extras e enter no código. Ou seja, se você escrever várias palavras com várias quebras de linha no código, elas ainda vão aparecer em uma única linha na página.



Para fazer uma quebra de linha, usamos a tag `
`

Comentários servem para explicar o código e não aparecem no navegador. São muito úteis para organizar o código ou deixar lembretes.

Para comentar no código, usamos `<!-- COMENTÁRIO -->`



FORMATAÇÕES BÁSICAS DE TEXTO

No HTML, podemos utilizar tags textuais específicas que proporcionam diferentes estilos e formatações.

- **Negrito visual:** usamos a tag ``
- **Negrito semântico (ênfase forte):** usamos a tag ``
- **Itálico visual:** usamos a tag `<i></i>`
- **Itálico semântico (ênfase leve):** usamos a tag ``
- **Sublinhado:** usamos a tag `<u></u>`
- **Marcação ou destaque:** usamos a tag `<mark></mark>`, que serve para chamar atenção em um trecho específico.
- **Sobrescrito:** usamos a tag ``, muito usado em marcas registradas, potências matemáticas e equações.
- **Subscrito:** usamos a tag ``, muito usado em fórmulas químicas.
- **Citação longa:** usamos a tag `<blockquote></blockquote>`

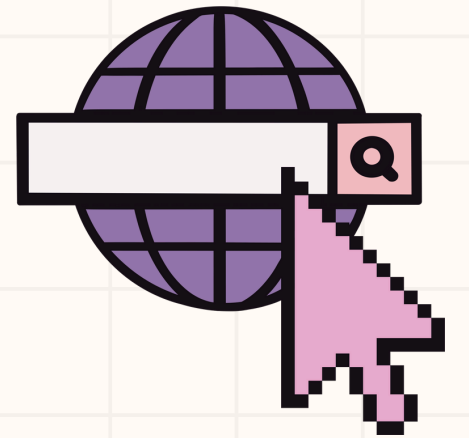


BOAS PRÁTICAS

1. Indentação (espaçamento): Sempre use indentação para mostrar a hierarquia das tags.
2. Comentários para separar seções: Use comentários (`<!-- -->`) para marcar blocos de código e organizar grandes páginas.
3. Fechar todas as tags corretamente: Mesmo que algumas tags funcionem sem fechamento (como `
` e ``), é boa prática sempre fechá-las corretamente quando for aplicável.
4. Usar letras minúsculas para tags e atributos: Apesar de o HTML não diferenciar maiúsculas de minúsculas, é padrão da comunidade usar minúsculas.
5. Espaçamento entre blocos: Deixe linhas em branco entre seções para evitar um código “amontoadado”.



LIÇÃO PROPOSTA PARA A SEMANA



Crie um arquivo chamado index.html no VSCode, Acode ou no editor que você estiver utilizando. Digite o conteúdo a seguir, aplicando boas práticas de código e respeitando a hierarquia de títulos, parágrafos e formatações:

- Título
- Subtítulo
- Conteúdo
- Conclusão

Coloquem na “web” e tirem um print da página e mandem no nosso grupo do WhatsApp com o nome.



**OBRIGADA PELA
ATENÇÃO!**

