



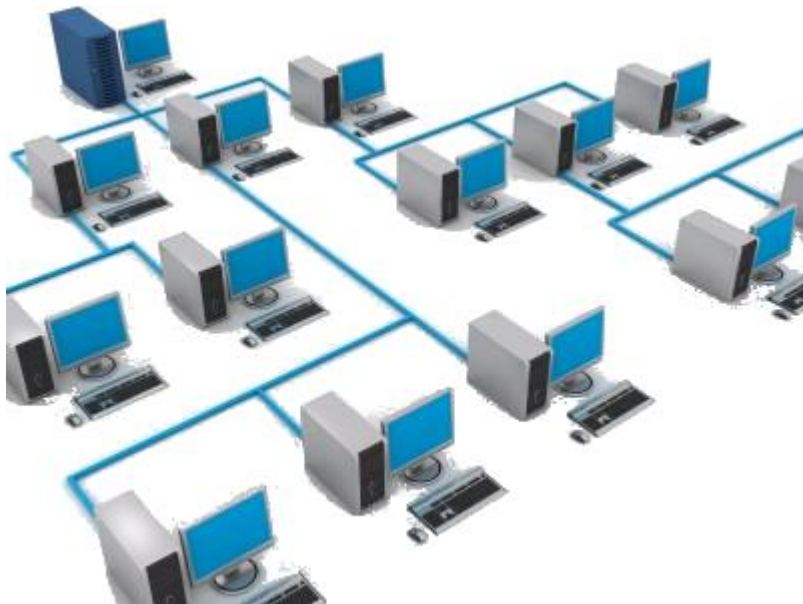
JS4GIRLS

Florianópolis - SC



Introdução ao HTML

Rápida Contextualização



Ao digitar um endereço do site(domínio) no navegador, o computador faz uma requisição para o servidor(outro computador), local onde este site está hospedado/guardado, e então, busca os arquivos necessários para renderizar/carregar na sua tela.

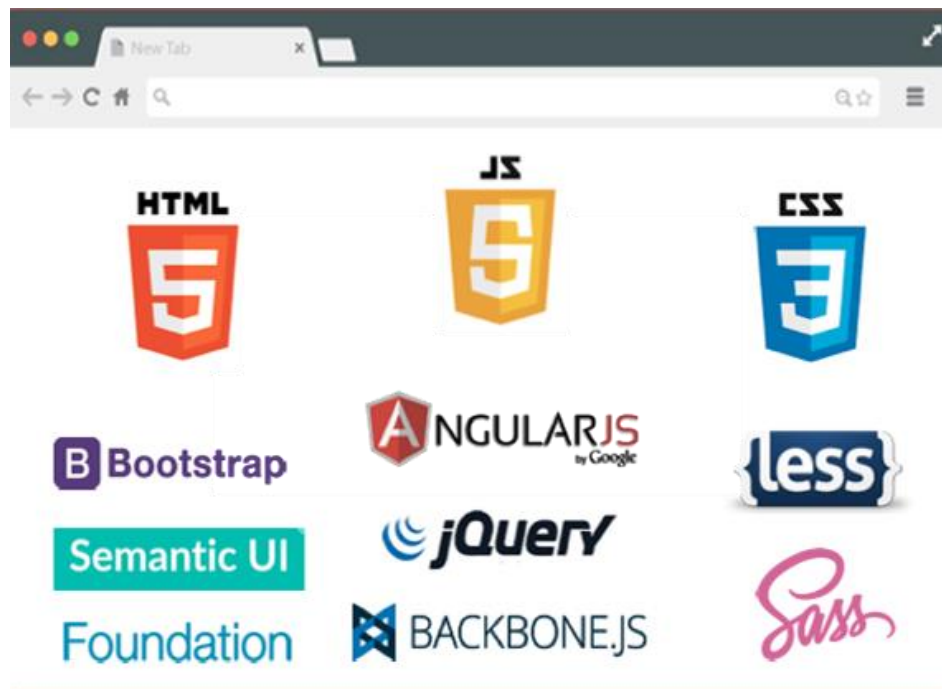
Como funciona um Website

O navegador necessita de um tipo de **linguagem** para poder interpretar a exibição de conteúdo, chama **HTML**(HyperText Markup Language, que significa Linguagem de Marcação de Hipertexto - não de programação), ela é composta por tags, é assim que estruturamos as páginas na WEB.

Um website básico normalmente é composto por 3 linguagens: HTML, CSS e JS.

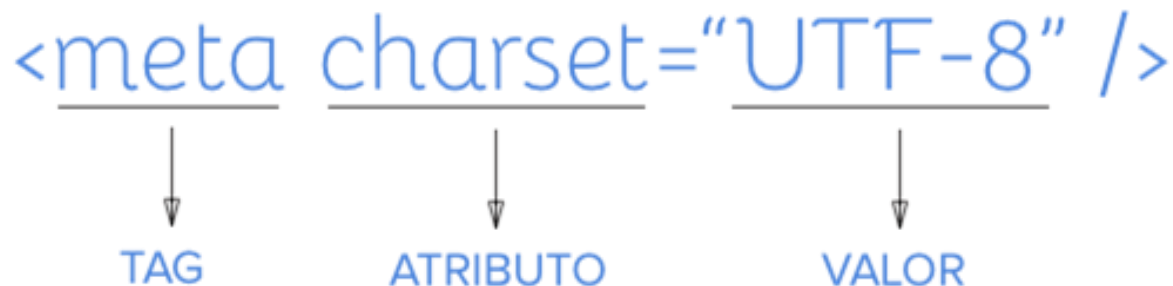


Framework é a junção de códigos para a realização de funcionalidades genéricas e digamos pré-pronta.



Sintaxe HTML - TAGS

Como já dito, o HTML é constituído por TAGs. Utilizamos a seguinte sintaxe para o navegador interpretar esse código:



<meta charset="UTF-8" />


↓ ↓ ↓

TAG ATRIBUTO VALOR

Em pares: <p> Meu parágrafo </p>

Única:

Estrutura básica da arquivo HTML



```
<!DOCTYPE html>
<html lang="pt-br">
  <head>
    <meta charset="utf-8" />
    <title>Minha primeira pagina em HTML</title>
  </head>
  <body>
    <h1>Meu primeiro título</h1>
    <p>Meu primeiro parágrafo</p>
  </body>
</html>
```

Glossário de TAGs

<!DOCTYPE html> Comando especial para que os navegadores interpretem esse documento com o HTML de versão mais atual.

<html lang="pt-br"> </html> Essa tag inicializa um documento HTML, já o atributo lang define o idioma padrão de todo o conteúdo, além de para os leitores de tela, dizer quais termos ele deve ler com o idioma nativo do termo.

<head> </head> Indica o cabeçalho, tudo que estiver entre a abertura e fechamento desta tag não aparecerá para o usuário, aqui dentro do seu escopo, são inseridas informações que fornecemos para nossa página e ou navegador.

<meta charset="utf-8" /> Para os textos da página renderizarem os caracteres como cedilha e acentuação corretamente, um navegador deve saber o conjunto de caracteres usado pela página. Para configurar esse conjunto, usa-se o atributo charset na tag <meta> (tag associada a metadados). O valor mais comuns usado hoje em dia é o UTF-8, também chamado de Unicode. O UTF-8 é a recomendação atual para encoding na Web por ser amplamente suportada em navegadores e editores de código, além de ser compatível com praticamente todos os idiomas do mundo.

<title></title> Título da minha aba do navegador/documento

<body> </body> Aqui entre a abertura e fechamento desta tag inserimos tudo que queremos que o usuário veja na nossa página, do nome inglês body significa corpo, portanto aqui é o corpo do documento.

<link rel="stylesheet" href="css/arquivo.css" /> Inclui arquivo de css na página.

<h1> </h1>, <h2> </h2>, <h3> </h3>, <h4> </h4>, <h5> </h5> e <h6></h6> Tags para criação do texto, o h1 possui uma tamanho de fonte maior e vai diminuindo esse tamanho até o h6 que é o menor.



Glossário de TAGs

`<p> </p>` Tag para inserção de parágrafos.

`` Deixa o texto em negrito.

`<hr />` Cria uma linha.

`<div></div>` Serve para alterar o estilo em partes específicas da página e posicionar objetos.

`<footer></footer>` Tag HTML5 que demarca o rodapé da página.

`` É usado para elementos que ficam alinhados lado a lado, não oferece mudança visual, fornece uma maneira de adicionar algo no meio de um texto ou de uma alguma parte do documento.

`` Inserir imagem.

`` ou `` - Inserir um link

Lista não ordenada:

``

`Neoway`

`Google`

``

Lista Ordenada:

``

`Estudar HTML`

`Fazer Exercícios`

``



Glossário de TAGs

Usado para inserir videos ou músicas

```
<embed src="../../assets/sounds/Bird-music.mp3" volume="20" type="audio/mp3" autostart="true"></embed>
```

Usado para inserir conteúdos externos dentro da página, por exemplo, vídeos do youtube ou conteúdos

```
<iframe src="https://www.youtube.com/embed/xpxYmbAwEZ8" frameborder="0" width="853" height="480"></iframe>
```

```
<iframe src="https://www.google.com" frameborder="0" width="853" height="480"></iframe>
```



Glossário de TAGs

<form></form> Demarca o início e término de um formulário, servirá para entrada de informações que usuário irá inserir, este elemento pode conter um ou mais das seguintes tags de formulário.

<form method="POST" action="enviar_formulario.php"> O atributo method especifica qual o método de envio dos dados do formulários, existem dois meios/valores possíveis: GET ou POST. O formulário de dados podem ser enviados como variáveis de URL (com method = "get") ou via HTTP (com method = "post").

GET: Acrescenta form-data para o URL em pares nome / valor; O tamanho da URL é limitado (cerca de 3000 caracteres); Nunca use GET para enviar dados sensíveis! (Será visível na URL); Útil para envios de formulários onde um usuário deseja marcar o resultado; GET é melhor para dados não-seguros, como seqüências de consulta no Google.

POST: Acrescenta forma de dados no interior do corpo do pedido HTTP (dado não é mostrado na URL); Não tem limitações de tamanho; Envios de formulários com POST não podem ser marcadas.

As tags abaixo devem estar contidas entre a abertura e fechamento da tag form:

Cria um campo de texto:

<input type="text" name="nome" placeholder="Digite seu nome" />

A tag input possui um atributo chamado type que recebe um valor que define o tipo de dado desse campo e transforma no campo de acordo com o tipo, então analisar o tipo de dado, por exemplo: nome, idade, data e etc.

Exemplos de valores para o atributo type:

text, button, checkbox, color, date, email, file, hidden, number, password, radio, range, reset, search e submit.

O atributo placeholder é uma dica que vai aparecer dentro do campo, quando clicar em cima do campo, a dica some.



Glossário de TAGs

`<textarea rows="4" cols="10"></textarea>` Cria uma caixa de texto, o atributo cols é o tamanho da largura em colunas e rows é a altura em linhas da caixa de texto.

Cria uma lista suspensa (dropdown list):

`<select>`

`<option>Selecione</option>`

`<option>Maçã</option>`

`</select>`

`<button>Enviar</button>` ou `<input type="submit" value="Enviar" />` Cria um botão, a diferença é que esse tipo de button é mais utilizado para ações com comandos javascript, agora para mandar enviar um formulário precisa utilizar o input submit, pois ele vai mandar o conteúdo do formulário de acordo com as infos que estão nos atributos da tag form.





Introdução ao CSS

Aqui enfeita as tags

O que é CSS

CSS Cascading Style Sheets (Folhas de Estilo em Cascata) é uma linguagem de folha de estilos criada para **estilizar páginas HTML**.

Com as regras, podemos selecionar um ou mais elementos(tags) em uma página web e aplicar estilos neles.



Sintaxe do CSS

Diagram illustrating the CSS syntax structure for the rule `p { color: red; }`:

- Seletor** (Selector): Points to the `p` (blue).
- Propriedade** (Property): Points to the `color` (purple).
- Valor** (Value): Points to the `red` (orange).

The diagram shows the following structure:

```
p {  
  color: red;  
}
```



Ligando nosso CSS no arquivo HTML

Para que as regras tenham efeito na página HTML, há 3 maneiras de inserir o CSS:

1. Arquivo **externo** criando um arquivo “.css” que é ligado à página HTML através da tag <link> que é inserida entre a tag <head> (**melhor método, utilizar sempre esse**);

Exemplo: `<link rel="stylesheet" href="css/arquivo.css" />`

1. **Incorporado** - Entre a tag <head> da página, utilizando a tag “<style>”. Os estilos em uma tag style tem precedência sobre os estilos em um arquivo externo.

Ex: `<head>`

`<style>`

`p {`

`color: red;`

`}`

`</style>`

`</head>`

1. **Inline** - Direto na tag HTML, utilizando o *atributo* style (estilos em linha).

Exemplo: `<p style="color: red; font-size: 20px;">Texto</p>` - Nunca utilizar esse método.



Seletores Visão geral

Seletores selecionam um (ou vários) elementos, para que você possa aplicar estilos nele(s).

`p { color: red; }`



Seleciona todos os elementos `<p>`

`p a { color: blue; }`



Seleciona todos os elementos `<a>` dentro

de `<p>`



Seletores CSS básicos

Simples ou compostos?

Existem duas categorias básicas de seletores: os simples e os combinados. Um seletor simples consiste em um tipo qualquer de seletor ou o seletor universal seguido por nenhum ou algum seletor de atributo, seletor tipo ID, seletor de classe ou pseudo-classe.

```
p { color: red; }
```

Um seletor combinado (algumas vezes chamado de seletor contextual ou composto) consiste de dois ou mais seletores simples separados por um elemento de combinação. A seguir um exemplo de seletor combinado:

```
div p { font-weight: bold; }
```



Seletor Universal

O seletor universal é representado por um asterisco, “*”, e casa com todos os elementos em seu escopo.

// todos os elementos - Global White Space Reset

* { margin: 0; padding: 0; }

// todos elementos dentro do #menu-lateral

#menu-lateral * { padding-left: 20px; }



Seletores Tipo

Seleciona um tipo de elemento. Seu valor é exatamente o texto da TAG HTML.

```
p { color: red; }
```

```
a { color: blue; }
```

```
img { max-width: 100%; }
```



Seletor – Classe

Classe é representada por um ponto [.] antes do nome, que é o que está no atributo class do html:

```
.pink-text { color: pink; }
```

```
<p> Meu parágrafo normal </p>
```

```
<p class="pink-text"> Meu parágrafo rosa </p>
```

O atributo class no HTML pode conter uma lista de vários nomes separados por espaço em branco:

```
<p class="pink-text large-text">Meu texto rosa grande</p>
```

```
<p class="pink-text small-text"> Meu texto rosa pequeno </p>
```

```
.pink-text { color: pink; }
```

```
.pink-text.large-text { font-size: 20px; }
```

```
.pink-text.small-text { font-size: 10px; }
```



Seletor – ID

ID's tem a função semelhante das classes, porém:

1. São exclusivos. Não deve existir mais de 1 elemento com o mesmo ID no HTML.
2. São representados por # ao invés de .

```
p { font-size: 10px; }
```

```
p.italico { font-style: italic; }
```

```
#main { font-size: 20px; }
```

```
<div id="main">
```

```
    <p class="italico"> Este texto será em itálico com 10px</p>
```

```
    <p id="introducao" class="italico"> Este terá 20px e será em itálico </p>
```

```
    <p> Este será apenas um texto com 10px </p>
```

```
</div>
```



Elementos de combinação

Elementos de combinação de seletores são usados para separar dois ou mais seletores simples que compõem um seletor combinado.

Os elementos de combinação disponíveis são:

- espaço em branco
- >
- +



Seletores Descendentes

- É uma combinação de dois ou mais seletores simples separados por um espaço em branco
- Refere-se à qualquer elemento descendente. Em qualquer nível.

```
p { color:black; }
```

```
div p { color:green; }
```

```
<p> Este parágrafo será preto </p>
```

```
<div>
```

```
<p> Um parágrafo terá a cor verde, pois está dentro do div.</p>
```

```
<article>
```

```
<p>Este também será verde, pois ainda está dentro do div.</p>
```

```
</article>
```

```
</div>
```

```
#nav li { display:inline; }  
#nav a { font-weight:bold; }
```

```
<ul id="nav">  
  <li><a href="#">Link 1</a></li>  
</ul>  
<ul>  
  <li><a href="#">Link 2</a></li>  
  <li><a href="#">Link 3</a></li>  
</ul>
```



Seletores Filho

Um seletor filho tem como alvo um **filho imediato** de um elemento.

```
div > p { font-weight: bold }
```

```
<div>
```

```
  <p>Este terá negrito</p>
```

```
  <article>
```

```
    <p>Este não terá negrito</p>
```

```
  </article>
```

```
</div>
```



Seletores Irmãos adjacentes

Um seletor de irmão adjacente tem como alvo o elemento diretamente ao lado do primeiro.:

p + strong { color: red }

<p>

 Este texto não será afetado mas ficará em negrito devido ao strong.

</p>

Este ficará vermelho pois é irmão do “<p>”.

 Já este não, pois há um elemento entre o “<p>” e este “”. Ele é adjacente ao “” e não ao “<p>”.

“+” se refere ao próximo irmão, ou seja, primeiro irmão seguinte ao elemento.



Agrupando Seletores

Você pode agrupar os seletores em uma lista e separando-os por uma vírgula, ao invés de escrever repetidamente a mesma regra para cada um dos seletores.

`p { color: red }
span { color: red }`



`p,
span { color: red }`

```
div#news h3,  
ul { margin:0 2em;  
}
```

```
div#news h3,  
div#news ul {  
margin:0 2em; }
```

```
<div id="news">  
  <h3>News</h3>  
  <ul>  
    <li>Item 1</li>  
    <li>Item 2</li>  
  </ul>  
</div>
```

Qual a diferença?



Seletores de Atributo

Seletores de atributo atingem elementos **com base no valor de atributo** declarado no seletor.

- [att] Qualquer elemento com o atributo att independente do seu valor.
- [att=val] - Qualquer elemento com o atributo att cujo valor seja “val”.
- [att~=val] - Qualquer elemento com atributo **att** com valores separados entre espaço, dentre eles **val** (<http://codepen.io/romuloctba/pen/GppwJL>)
- [att|=val] - Qualquer elemento com atributo att de valor igual a um valor qualquer separado por um hífen de um valor começando com “val”. *O principal uso deste seletor é o de casar elementos com um valor de idioma especificado no atributo lang (xml:lang em XHTML), por exemplo; “en”, “en-us”, “en-gb”, etc.*

```
input[type=text] { border: 1px solid #000000; }
```

```
<input type="text" />
```



Projeto Final

Links para estudos

Conheça mais Tags HTML:

https://developer.mozilla.org/pt-BR/docs/Web/HTML/HTML5/HTML5_element_list

HTML para iniciantes:

<http://www.w3schools.com/html/>

CSS para iniciantes:

<http://www.w3schools.com/css/>

Fontes gratuitas:

<https://www.google.com/fonts>

Paleta de Cores em hexadecimal

<http://flatcolors.net/palettes>

