

WEB

HTML

- **Hypertext Markup Language** (Linguagem de Marcação de Hipertexto)
- A Linguagem de Marcação de Hipertexto (HTML) é uma linguagem de computador que compõe a maior parte das páginas da internet e dos aplicativos online. Um hipertexto é um texto usado para **fazer referência a outros textos**, enquanto uma linguagem de marcação é composta por uma **série de marcações(TAGS)** que dizem para os servidores da web qual é o estilo e a **estrutura de um documento**.
- O HTML **não é considerado uma linguagem de programação**, já que ele não pode criar funcionalidades dinâmicas. Ao invés disso, com o HTML, os usuários podem criar e estruturar seções, parágrafos e links usando elementos, tags e atributos.

Como funciona

Documentos HTML são arquivos que terminam com uma extensão .html ou .htm. Um navegador lê o arquivo HTML e renderiza o seu conteúdo para que os usuários da internet possam vê-lo.

Todas as páginas HTML possuem uma série de elementos, que consistem num conjunto de **tags e atributos**.

- Uma tag diz para o navegador onde um elemento começa e termina, enquanto um atributo descreve as características de um elemento.

Tag de abertura – usada para dizer onde um elemento começa a ter efeito. A tag é cercada de colchetes angulares para abertura e fechamento. Por exemplo, use a tag de abertura **<p>** para criar um parágrafo.

Conteúdo – essa é a parte que os usuários verão.

Tag de fechamento – igual à tag de abertura, mas com uma barra antes do nome do elemento. Por exemplo, **</p>** para encerrar um parágrafo.

A combinação dessas três partes vai criar um **elemento HTML**.

A maioria dos elementos possui uma tag de abertura e de fechamento, mas alguns não precisam fechar a tag para funcionar. Esse é o caso dos elementos vazios. Eles não usam uma tag de fechamento pois não têm conteúdo:

Essa **tag de imagem** possui dois atributos — um **atributo src** (que é um caminho de imagem) e um atributo alt (que é o texto de descrição). Contudo, ele não tem conteúdo nem uma tag de fechamento.

Estrutura de um documento

```
<!DOCTYPE HTML>
<html>
  <head>
    Informações sobre o documento: autor, título, etc...
  </head>
  <body>
    Corpo do documento (parte a ser visualizada)
  </body>
</html>
```

Elementos básicos

DOCTYPE : Define o tipo de documento

<html> : Define o início de um documento HTML , todos as demais tags devem estar dentro desta tag **</html>**

<head>: É o cabeçalho do documento , onde ficam metadados do documento, como metatags, css e javascript **</head>**

Meta tags : As meta tags são declarações inseridas à seção head do código HTML ;
Descrevem o conteúdo de uma página HTML para os mecanismos de busca

- **funcionam como pequenas instruções para os algoritmos dos sites de busca.**
- **permitem que os mecanismos de pesquisa categorizem o conteúdo do site corretamente.**

<title> : define o título do documento, mostrado na barra de títulos do navegador e como link nos mecanismos de busca **</title>**

<meta> metatags, como charset, description, viewport ou keywords

Charset : O atributo charset serve para indicar o formato de codificação de **caracteres** utilizado no documento para os NAVEGADORES

<link> : utilizado para linkar um arquivo de **imagem ou css** a página html

<style>: utilizada para adicionar um **código css** a página**</style>**

<body>: corpo ou **área principal do documento**, dentro deste deve ficar o conteúdo da sua página**</body>**

<main>: define o **conteúdo principal** da página, deve ser utilizado apenas uma única vez em cada página**</main>**

<h1 a h6>: representam 6 níveis de títulos, sendo o h1 de maior importância**</h>**

<header>: define um **cabeçalho da página ou de uma seção**, normalmente pode conter um menu (nav), um logo ou link**</header>**

****: define a **utilização de uma imagem** (JPG, GIF, PNG e outras). A imagem deve ser apontada com o uso do **atributo src**

<nav>: define uma **área de navegação** contendo links, para a formação de menus com hiperlinks**</nav>**

****: define um elemento ou item da lista não ordenada****

****: define uma lista não ordenada****

****: lista ordenada ****

<a>: define um **link** que deve ser apontado com o **atributo href******

<section>: define uma **seção** do documento **</section>**

<footer>: define o rodapé de uma página ou sessão **</footer>**

<table>: cria um **espaço/bloco** para a inserção de linhas e colunas em forma de uma tabela. A função da tag <table> é apenas delimitar este espaço, ficando reservado à tag **<tr>** a criação das linhas da tabela, e dentro desta às tags **<th>** ou **<td>** a criação das colunas.**</table>**

- Utilizando as tags **<thead>**, **<tbody>** e **<tfoot>**, podemos dar à nossa tabela, respectivamente, um cabeçalho, definir o "corpo" dos dados e adicionar um rodapé.

<tr>: cria as **linhas** da tabela **</tr>**

<td>: significa **table data** é usada para definir uma célula(coluna)**</td>**

<th>: As tabelas podem ter **cabeçalhos**. Para isso usa-se a etiqueta <th> que significa **table header**. Os cabeçalhos podem ser usados nas linhas e/ou nas colunas.**</th>**

<address>: define as informações de contato do autor / proprietário de um documento ou um artigo. Se o elemento <address> está dentro do elemento <body>, representa informações de contato para o documento.**</address>**

<form>: começo e um formulário em html ; . É um elemento de container como o <div> e <p> **</form>**

<fieldset>: "Fieldset Element" ou campo definido é utilizado como diz o próprio nome; **definir os limites visuais (perímetro)** de um formulário

- . Pode-se ou não ser usada com a tag **<LEGEND>****</LEGEND>** quando então o leitor estará diante de um arquivo de aço cheio de divisórias onde pastas de arquivos estão separadas por cartões duros com uma aba de identificação.

<legend>: representa um rótulo para o conteúdo do seu ancestral **<fieldset>**.**</legend>**

<label>: representa **uma legenda para um item em uma interface de usuário**. Ele pode estar associado com um elemento de controle, colocando este dentro do elemento label, ou usando o atributo **for**. Tal controle é chamado o controle etiquetado do elemento etiqueta.

<input>: um **campo básico texto** de linha única

<div>: é um **container genérico** para conteúdo de fluxo, que de certa forma não representa nada. Ele pode ser utilizado para agrupar elementos para fins de estilos (usando class ou id), ou porque eles compartilham valores de atributos, como lang.**</div>**

<select>: representa um controle que apresenta um **menu de opções**. As opções dentro do menu são representadas pelo elemento **option**, que podem ser agrupados por elementos **optgroup** **</select>**

<option>: Em um formulário Web, o elemento HTML **<option>** é usado para criar um controle que representa um item dentro de um elemento HTML5 **select**, **optgroup** ou **datalist**. As **opções** podem ser pré-selecionadas para o usuário.**</option>**

- O que é **Value** em HTML?

O value é a informação que será enviado pro servidor, pode ser usado 1, 2, f, m, masculino, feminino, tanto faz. Porque é tu que vai receber essa informação e tratar a utilização dela.

<textarea>: representa um controle de edição para uma **caixa de texto**, útil quando você quer permitir ao usuário informar um texto extenso em **formato livre**, como um comentário ou formulário de retorno. **</textarea>**

Atributos dos Elementos

Atributos HTML são palavras especiais usadas dentro da tag de abertura para controlar o comportamento do elemento.

- Os atributos HTML são um modificador de um tipo de elemento HTML.
- Com eles podemos identificar melhor um elemento, informar qual arquivo aquela tag deve utilizar, indicar o tipo de um campo de texto, etc...

➤ **Class = "NomeDaClasse"**

Classes são como classificações de uma tag/elemento, para que no CSS estilizar uma tag específica, ou um conjunto de tags. Também é possível usar no JavaScript para selecionar uma tag específica.

Ex: **<h1 class="titulo">Mergulhe em Tecnologia!</h1>**

Seleção no CSS : **.titulo**

➤ **id="NomeDoid"**

Identificador ; Identificar de forma única um elemento naquela página HTML.

Seleção no CSS: **#NomeDoid**

➤ **src="Link ou diretório da mídia"**

Comumente utilizado para indicar para a tag **qual arquivo ou mídia utilizar**.

Recebe valores como links (<https://google.com/minhaimagem.jpeg>) ou o nome de um arquivo já presente no projeto (/minhaimagem.jpeg).

➤ **href="Url"**

Para a tag **<a>**, o atributo **href especifica a URL** da página para a qual o link vai.

➤ **lang="Linguagem"**

O atributo lang especifica **o idioma do conteúdo da tag**

❓ **target="blank"**

Esse atributo abre o link do documento em **uma nova janela ou aba**.

CSS (VISUAL)

1. Salvar o arquivo como **style.css** na sua pasta styles.
2. Aplicar o CSS ao seu documento HTML.

```
<head>
```

```
<link rel="stylesheet" type="text/css" href="css/style.css">
```

```
</head>
```

Seletor(selector)

O nome do elemento HTML no começo do conjunto

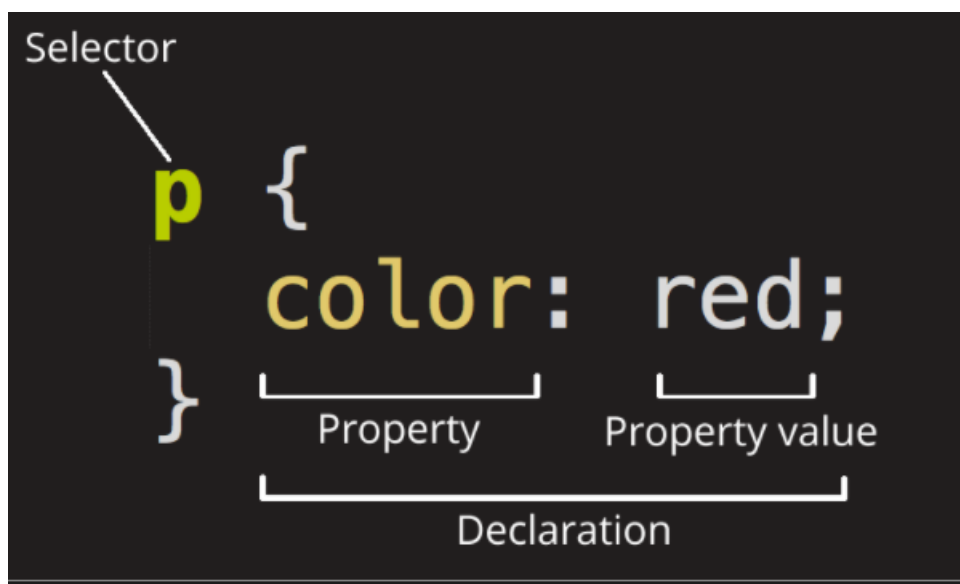
Ex: p{

}

também pode selecionar vários tipos de elementos e aplicar um único conjunto de regras a todos eles. Inclua múltiplos seletores separados por vírgulas.

Ex:

```
p, li, h1 {  
    color: red;  
}
```



}

*

O asterísco (ou estrela) tem como alvo todos os elementos em uma página.

{

margin: 0;

padding: 0;

}

zerar as **margens** e o **preenchimento**

• SELETORES

***{** Seletor universal: qualquer elemento;

e { Seletor de tipo de elemento: seleciona qualquer elemento **<e>**;

e f{ Seletor contextual: seleciona qualquer elemento **<f>** que **estiver contido** num elemento **<e>**;

e:first-child { Pseudo-classe primeiro filho: **seleciona o primeiro elemento <e>** descendente do elemento-pai;

e:hover{ Pseudo-classe hover: aplica-se ao elemento **<e>** **quando o cursor estiver sobre ele**, mas sem ativá-lo;

e:focus{ Pseudo-classe focus: aplica-se ao elemento **<e>** quando o **foco** estiver posicionado nele;

- **TEXTO**

text-align:

Esta propriedade destina-se a definir o alinhamento horizontal do texto. Trata-se de uma propriedade herdada.

- **left**

Alinha o texto à esquerda do box. Este é o valor inicial (ou default).

- **right**

Alinha o texto à direita do box.

- **center**

Alinha o texto no centro do box.

- **justify**

Alinha o texto de modo que cada linha estenda-se da esquerda à direita do box.

text-decoration: none Efeitos decorativos no texto como sublinhado, tachado, linha acima e piscante . Neste caso sem efeito decorativo

- **CORES**

color:

Cor do texto presente no elemento

background-color:

Cor de fundo do elemento

background-image:

Imagem de fundo do elemento

background:

Atalho para definir de uma só vez todas as propriedades de imagem de fundo

- **BORDAS**

border-top-widthpx

border-right-widthpx

border-bottom-widthpx

border-left-widthpx

Largura das bordas em cada um dos quatro lados: superior, direito, inferior e esquerdo

border-width

Atalho para definir de uma só vez larguras de bordas para todos os lados

border-radius

Define bordas arredondadas

border-top-color

border-right-color

border-bottom-color

border-left-color

Cor das bordas em cada um dos quatro lados: superior, direito, inferior e esquerdo

border-color

- **MARGENS**

margin-top: Xpx

margin-right: Xpx

margin-bottom: Xpx

margin-left: Xpx

Tamaho da margem para cada um dos lados: superior, direito, inferior e esquerdo

margin : Xpx

- **ESPAÇAMENTO**

padding-top: xpx

padding-right: xpx

padding-bottom: xpx

padding-left: xpx

Distância utilizada para espaçamento em cada um dos quatro lados: superior, direito, inferior e esquerdo

padding: xpx

- **Modelo de formatação visual**

display

Modos de exibição do conteúdo, [inline | block | inline-block| ... | none]

inline: O elemento renderizado em uma ou mais linhas

block: renderizado como um bloco

inline-block: renderizado em uma linha simples, mas comporta-se como um bloco

none: o elemento não será mostrado no modo visual

position

Esquema de posicionamento , [relative | absolute | ...]

relative: posicionamento é calculado de acordo com o fluxo normal.

absolute: Posicionamento (e possivelmente o tamanho) é especificado com as propriedades '**top**', '**right**', '**bottom**', e '**left**'. Essas propriedades são especificadas em relação ao conteúdo do bloco

top

right

bottom

left

Distância do elemento com relação à extremidade de seu elemento-pai

float

Posicionamento **adjacente** à esquerda ou à direita do elemento, { left | right | none }

left: o elemento será posicionado flutuando à esquerda. O fluxo do conteúdo não integrante do elemento será posicionado à direita a partir do 'top' do elemento que recebe a propriedade 'float'

right: similar à 'left', mas com o elemento flutuando à direita

none: o elemento não flutua

direction:

Define a direção de um texto: esquerda para direita; direita para esquerda, [ltr | rtl]

ltr - direção da esquerda para a direita

rtl - direção da direita para a esquerda

Detalhes do modelo de formatação visual

width, Largura, [<tamanho> | <porcentagem> | auto], Exemplo:
e {width: 25%;}

min-width, Largura mínima, [<tamanho> | <porcentagem>], Exemplo:
e {min-width: 200px;}

max-width, Largura máxima, [<tamanho> | <porcentagem> | none], Exemplo:
e {max-width: 600px;}

height, Altura, [<tamanho> | <porcentagem> | auto], Exemplo:
e {height: 250px;}

min-height, Altura mínima, [<tamanho> | <porcentagem>], Exemplo:
e {min-height: 30%;}

max-height, Altura máxima, [<tamanho> | <porcentagem> | none], Exemplo:
e {max-height: 70%;

line-height, Distância entre as linhas de um texto, ,

- **FONTES**

font-family

font-style

Estilo utilizado na fonte, [normal | italic | oblique]

font-size

Tamanho da fonte

font

Atalho para definir de uma só vez todas as propriedades de fonte

