

**Elementos**

Elementos básicos

# Doctype: `<!DOCTYPE html>`

- não é uma tag HTML, mas sim uma declaração para informar ao navegador qual versão do HTML o arquivo utiliza;
- não é obrigatório, porém seu uso é altamente recomendado;
- deve ser declarado na primeira linha do arquivo;
- é *case insensitive*;
- versão 5:
  - `<!DOCTYPE html>;`
- versão 4 (estrito):
  - `<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 strict//EN" "http://www.w3.org/TR/html4/strict.dtd">;`

**HTML:** `<html></html>`

- elemento raiz do documento;
- envolve todo o conteúdo da página;

Exemplo:

`<html> ... </html>`

# Head: `<head></head>`

- cabeçalho da página;
- recipiente de todo conteúdo que não é visível aos usuários;
- possui instruções para o navegador carregar e mostrar o conteúdo do site;
  - título da página;
  - palavras chave, descrição e outras informações para *crawlers*, SEO (*Search Engine Optimization*) e SEM (*Search Engine Marketing*);
  - declaração de conjunto de caracteres;
  - inclusão de CSS para estilização e JS para manipulação;

Exemplo:

```
<head> ... </head>
```

## Body : `<body></body>`

- corpo da página;
- recipiente de todo conteúdo visível aos usuários;
  - textos;
  - imagens;
  - vídeos;
  - tabelas;
  - etc;

Exemplo:

```
<body> ... </body>
```

## Comment: `<!--...-->`

- *tag* para adicionar comentários ao documento;
- os comentários não são visíveis para os usuário (somente no código fonte);
- utilizado para adicionar anotações ou explicar um trecho de código;

Exemplo:

```
<!-- Isto é um comentário -->
```

## Title: `<title></title>`

- define o título da página: guia do navegador, favoritos, resultado de buscas;
- deve ficar dentro da *tag* `<head>`;
- pode conter apenas texto (não pode conter outros elementos);
- uso obrigatório\*;
- não é recomendado possuir mais de 50 caracteres;

Exemplo:

```
<title>Minha página</title>
```



## Metadata: `<meta>`

- define metadados (dados que descrevem dados) sobre o documento para o navegador;
- deve ficar dentro da *tag* `<head>`;
- geralmente utilizado para definir o conjunto de caracteres, descrição, palavras-chave e autor do documento, além de outras recomendações para o navegador como o *viewport* e outros *pragmas*;
- não é visível para o usuário, mas pode afetar o funcionamento da página;
- utilizado para SEO e SEM;

# Metadata: `<meta>`

Exemplo:

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

```
<meta name="description" content="Uma página para quem gosta de pudim.">
```

```
<meta name="keywords" content="HTML, CSS, JavaScript">
```

```
<meta name="author" content="John Doe">
```

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

## Base: `<base>`

- define a URL base para a página e/ou o *target* para todos os links relativos para o documento;
- pode existir apenas um elemento `<base>` no documento;
- deve ser utilizado no `<head>`;
- atributos `href` e `target`;

Exemplo:

```
<base href="https://www.google.com.br/" target="_blank">
```

## Script: `<script></script>`

- tag utilizada para incluir scripts JavaScript na página;
- pode conter código JS ou apontar para o caminho de um arquivo através do atributo `src`;
- pode ser utilizado tanto no `<head>` quanto no `<body>`;
- atributos `async`, `defer` e `type`;

Exemplo:

```
<script type="text/javascript" src="arquivo.js"></script>
```

## Style: `<style></style>`

- tag utilizada para definir informações de estilização (CSS) para o documento;
- deve conter código CSS;
- pode ser utilizado dentro da *tag* `<body>`, porém é altamente recomendado utilizar na *tag* `<head>`;
- atributos `media` e `type`;

Exemplo:

```
<style type="text/css"> ... </style>
```

## Link: `<link>`

- tag utilizada para definir a relação entre o documento e outro recurso externo;
- geralmente utilizado para incluir arquivos CSS externos ou favicons;
- deve ser utilizado no `<head>`;
- atributos `href`, `rel`, `type`, `media` e `sizes`;

Exemplo:

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

Elementos estruturais

## Header: `<header></header>`

- representa um *container* para conteúdos introdutórios ou um conjunto de links de navegação;
- geralmente utilizado em publicações de blogs, fóruns e notícias;
- um documento pode ter várias *tags* `<header>`;

Exemplo:

```
<header> ... </header>
```



## Footer: `<footer></footer>`

- representa um *container* de fechamento de uma seção do documento;
- normalmente contém informações sobre o autor da seção, direitos autorais ou links para documentos relacionados;
- um documento pode ter várias *tags* `<footer>`;

Exemplo:

```
<footer> ... </footer>
```

## Main: `<main></main>`

- representa o conteúdo principal de um documento;
- seu conteúdo deve ser único;
- não deve conter barras laterais, menus de navegação, logos e formulários que se repetem em outras páginas;
- um documento deve ter apenas uma *tag* `<main>`;

Exemplo:

```
<main> ... </main>
```

## Section: `<section></section>`

- define uma seção do documento;
- cada `<section>` geralmente inclui um cabeçalho relacionado;
- não deve ser utilizado como *container* genérico;

Exemplo:

```
<section> ... </section>
```

## Article: `<article></article>`

- define um conteúdo independente, auto-contido;
- geralmente utilizado em publicações de blogs, fóruns e notícias;
- deve ser utilizado apenas quando faz sentido sem o resto do conteúdo da página;

Exemplo:

```
<article> ... </article>
```

## Aside: `<aside></aside>`

- define um conteúdo a parte do restante do conteúdo em que está inserido;
- geralmente utilizado como barra lateral;
- um documento pode ter várias *tags* `<aside>`;

Exemplo:

```
<aside> ... </aside>
```

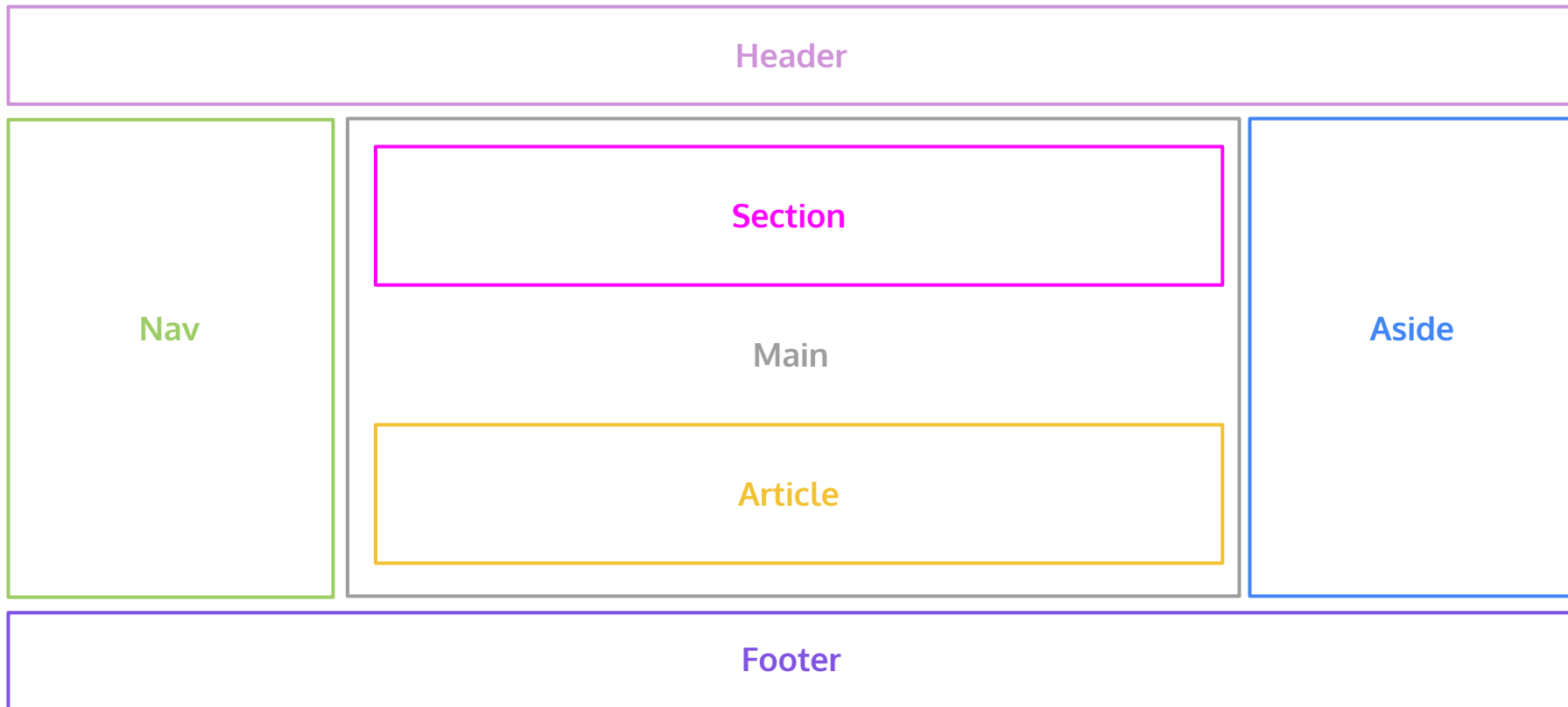
# Navigation: `<nav></nav>`

- define um conjunto de links para navegação;
- um documento pode ter várias *tags* `<nav>`;

Exemplo:

```
<nav>  
  <a href="/html/">HTML</a>  
  <a href="/css/">CSS</a>  
  <a href="/js/">JS</a>  
</nav>
```

# Estrutura:



Elementos de conteúdo



## Division: `<div></div>`

- *tag* utilizada como *container* para um ou mais elementos em forma de bloco;
- geralmente utilizado para envelopar um texto que será estilizado de alguma forma, ou manipulado via JS;
- qualquer tipo de conteúdo pode ser colocado dentro de uma div;
- por padrão, o navegador insere uma quebra de linha antes e depois da div;

Exemplo:

```
<div> ... </div>
```

## Paragraph: `<p></p>`

- elemento de parágrafos de texto;
- utilizado com bastante frequência para demarcar textos;
- por padrão, sempre irá ocupar uma nova linha e todo seu espaço horizontal;

Exemplo:

```
<p>Eu gosto de pudim e de brócolis.</p>
```

# Headings: `<h1></h1>...<h6></h6>`

- elementos de cabeçalho;
- geralmente utilizados para representar títulos e subtítulos da página;
- existem 6 níveis de títulos: `<h1>`, `<h2>`, `<h3>`, `<h4>`, `<h5>` e `<h6>`;
- `<h1>` é o mais importante, enquanto `<h6>` é o menos importante;
- possuem valor semântico, utilizado para acessibilidade, SEO e SEM;
- não deve ser utilizado para simplesmente aumentar o tamanho da fonte ou deixar o texto em negrito;
- a página não deve ter mais de um `<h1>`;
- pode conter mais de um cabeçalho de outros níveis (preferência hierárquica);

**Headings:** `<h1></h1>...<h6></h6>`

Exemplo:

`<h1>Título 1</h1>`

`<h2>Título 2</h2>`

`<h3>Título 3</h3>`

`<h4>Título 4</h4>`

`<h5>Título 5</h5>`

`<h6>Título 6</h6>`

## Unordered List: `<ul></ul>`

- elemento de listas de itens não ordenados;
- utilizado onde a ordem dos itens não é um requerimento;
- possui um "•" como marcador principal, que pode ser alterado;
- pode conter indeterminados itens, subitens e assim por diante;

## Ordered List: `<ol></ol>`

- elemento de listas de itens ordenados;
- utilizado onde a ordem dos itens é um requerimento;
- possui números decimais como marcador principal, que pode ser alterado;
- pode conter indeterminados itens, subitens e assim por diante;

## List Item: `<li></li>`

- elemento de item de lista;
- pode ser utilizado tanto em listas ordenadas quanto em não ordenadas;
- possui números decimais como marcador principal, que pode ser alterado;

# Lists:

Exemplo:

```
<ul>  
  <li>Pudim</li>  
  <li>Brócolis</li>  
  <li>Picanha</li>  
</ul>
```

```
<ol>  
  <li>Picanha</li>  
  <li>Pudim</li>  
  <li>Brócolis</li>  
</ol>
```



## Blockquote: `<blockquote></blockquote>`

- utilizado para representar uma citação longa de uma fonte externa;
- por padrão, sempre irá ocupar uma nova linha e todo seu espaço horizontal;
- geralmente é estilizado de uma forma diferente;
- possui o atributo `cite` que pode ser utilizado para referenciar sua origem;

Exemplo:

```
<blockquote cite="https://www.google.com/">...</blockquote>
```

## Preformatted: `<pre></pre>`

- define um texto pré formatado;
- o texto será apresentado em fonte monoespaçada;
- serão preservados espaços e quebras de linha escritos no documento;

Exemplo:

```
<pre>
```

```
Texto pré formatado
```

```
com várias linhas
```

```
e      também   espaços
```

```
</pre>
```

## Address: `<address></address>`

- utilizado para definir informações de contato para o autor do documento ou de um artigo;
- pode incluir qualquer tipo de informação de contato, como endereço físico, e-mail, telefone, mídia social, etc;
- o texto será apresentado em itálico;

`<address>`

Av. Avelino Talini, 171  
Bairro Universitário,  
Lajeado / RS, 95914-014  
Brasil

`</address>`

## Horizontal Rule: `<hr>`

- utilizado para representar uma mudança de tema/assunto;
- geralmente é representado com uma linha horizontal, utilizada para separar visualmente o conteúdo da página;

Exemplo:

`<hr>`

Elementos de formatação

## Span: `<span></span>`

- *tag* utilizada como *container inline*;
- geralmente utilizado para envelopar um texto que será estilizado de alguma forma, ou manipulado via JS;
- muito parecida com a *tag* `<div>`, porém a *div* possui *display* de bloco, enquanto o *span* é *inline*;

Exemplo:

```
<span>Exemplo</span>
```

## Anchor: `<a></a>`

- elemento de hyperlink;
- utilizado para direcionar o usuário à outra página, arquivo, endereço de e-mail, número de telefone, ou qualquer outra URL válida;
- o atributo `href` deve informar o endereço de destino do link;
- atributos `target`, `rel`, `download` e `type`;

Exemplo:

```
<a href="https://www.google.com/" target="_blank">Google</a>
```

# Strikethrough: `<s></s>`

- representa um texto que não está correto ou não é mais relevante;
- o texto será apresentado tachado, com um risco no meio;

Exemplo:

`<s>Exemplo</s>`



## Deleted: `<del></del>`

- representa um texto que foi excluído da página;
- o texto será apresentado tachado, com um risco no meio;

Exemplo:

`<del>Exemplo</del>`

## Inserted: `<ins></ins>`

- representa um texto que foi adicionado à página;
- o texto será apresentado sublinhado;

Exemplo:

`<ins>Exemplo</ins>`

## Underlined: `<u></u>`

- representa um texto não articulado;
- o texto será apresentado sublinhado;
- utilizado quando se quer dar um destaque a uma palavra específica, ou frase pequena;
- não é recomendado utilizar onde pode ser confundido com um `<a>`;

Exemplo:

`<u>Exemplo</u>`

## Strong: `<strong></strong>`

- representa um texto que possui importância;
- o texto será apresentado negrito;

Exemplo:

`<strong>Exemplo</strong>`

**Bold:** `<b></b>`

- tag para apresentar um texto em negrito, mas sem dar importância semântica às palavras;
- deve ser utilizado como último recurso;

Exemplo:

`<b>Exemplo</b>`

## Emphasis: `<em></em>`

- representa um texto que possui ênfase;
- o texto será apresentado em itálico;

Exemplo:

`<em>Exemplo</em>`

## Italic: `<i></i>`

- define um texto que deve ser lido com uma voz ou humor diferenciado;
- o texto será apresentado em itálico;
- geralmente utilizado para indicar termos técnicos ou de línguas estrangeiras;
- também muito utilizado como *container* para ícones;
- deve ser utilizado como último recurso;

Exemplo:

`<i>Exemplo</i>`

**Mark:** `<mark></mark>`

- define um texto que deve ser marcado;
- o texto será apresentado com uma cor de fundo;

Exemplo:

`<mark>Exemplo</mark>`



## Line Break: `<br>`

- utilizado para forçar uma quebra de texto simples;
- é geralmente útil para escrever endereços ou poemas;
- deve ser usado apenas para inserir quebras de linha, não espaços entre parágrafos;

Exemplo:

`<br>`

## Word Break Opportunity: `<wbr>`

- utilizado para informar uma possível quebra de linha;
- quando uma palavra é muito grande, o navegador pode quebrá-la no lugar errado, e esta tag pode corrigir isso;

Exemplo:

`<wbr>`

## Code: `<code></code>`

- utilizado para definir trechos de código de programação;
- o texto será apresentado em fonte monoespçada;

Exemplo:

`<p>A tag code é legal.</p>`

## Quote: `<q></q>`

- utilizado para representar uma citação curta de uma fonte externa;
- pode ser utilizado dentro de um `<p>` que não irá quebrar o texto;
- o navegador pode adicionar aspas duplas ao redor do texto;
- possui o atributo `cite` que pode ser utilizado para referenciar sua origem;

Exemplo:

```
<q cite="https://www.google.com/">...</q>
```

## Cite: `<cite></cite>`

- utilizada para representar uma referência a um trabalho artístico;
- geralmente utilizado dentro de um `<blockquote>` ou `<q>`, mas pode ser incluído em qualquer elemento;
- geralmente é formatado com estilo em itálico;

Exemplo:

```
<p>Mais informações podem ser encontradas em <cite>[ISO-0000]</cite>.</p>
```

## Time: `<time></time>`

- representa uma data, hora ou data e hora no calendário Gregoriano;
- visa apresentar datas e horas em um formato semântico;
- dispositivos e navegadores podem oferecer a opção de “adicionar a agenda” do usuário;
- utilizado para SEO;

Exemplo:

`<p>Aberto das <time>08:00</time> às <time>18:00</time>.</p>`

## Superscript: `<sup></sup>`

- representa um texto sobrescrito;
- o texto será apresentado em fonte menor, acima da metade da linha;
- geralmente utilizado para notas de rodapé;

Exemplo:

```
<p>Exemplo<sup>1</sup></p>
```

## Subscript: `<sub></sub>`

- representa um texto subscrito;
- o texto será apresentado em fonte menor, abaixo da metade da linha;
- geralmente utilizado para fórmulas químicas;

Exemplo:

```
<p>H<sub>2</sub>O</p>
```



# Small: `<small></small>`

- representa um texto escrito em fonte menor do que o resto da linha;

Exemplo:

```
<p>Um texto <small>pequeno</small>.</p>
```

## Variable: `<var></var>`

- utilizado para definir variáveis matemáticas ou de programação;
- o texto será apresentado em itálico;

Exemplo:

`<p>A área do triângulo é: <var>b</var> * <var>h</var> / 2.</p>`

## Abbreviation: `<abbr></abbr>`

- utilizado para definir abreviações e acrônimos, como "HTML", "CSS", "Dr.", "Dra.", "Sr.", "Sra.", etc;
- o atributo `title` deve informar a descrição completa da abreviação;
- o texto será apresentado em sublinhado pontilhado;

Exemplo:

```
<abbr title="Cascading Style Sheets">CSS</abbr>
```

## Progress: `<progress></progress>`

- elemento de barra de progresso;
- utilizado para mostrar o progresso de algo;
- atributos `value` e `max`;

Progress:

```
<progress value="30" max="100">30%</progress>
```

## Meter: `<meter></meter>`

- define uma medida escalar;
- utilizado para mostrar valores parciais de um total conhecido;
- atributos `min`, `max`, `high`, `low`, `value` e `optimum`;

Progress:

```
<meter value="2" min="0" max="10">2 de 10</meter>
```

## SVG: `<svg></svg>`

- *container* para gráficos SVG;
- o SVG possui vários métodos para desenhar caminhos, caixas, círculos, texto e imagens;

Exemplo:

```
<svg width="100px" height="100px">  
  <circle cx="50" cy="50" r="40" stroke="green" stroke-width="4"/>  
</svg>
```

Elementos multimídia

## Image: `<img>`

- utilizada para exibir imagens no documento;
- pode ser utilizada sozinha, ou em conjunto com os elementos `<figure>` e `<figcaption>` OU `<picture>`;
- o atributo `src` é obrigatório e deve definir o caminho para a imagem;
- o atributo `alt` é obrigatório e deve definir um texto alternativo para caso a imagem não carregue corretamente e acessibilidade;
- atributos `srcset`, `sizes`, `loading`, `width` e `height`;

Exemplo:

```

```



## Source: `<source>`

- utilizada para especificar múltiplas fontes de multimídia para elementos de mídia, como `<video>`, `<audio>` e `<picture>`;
- possibilita trabalhar com formatos de mídia mais novos e deixar o *browser* escolher o mais adequado e suportado;
- atributos `src`, `srcset`, `type`, `media` e `sizes`;

Exemplo:

```
<source src="arquivo.mp4" type="video/mp4">
```

## Video: `<video></video>`

- utilizada para adicionar vídeos à página;
- deve possuir ao menos uma tag `<source>` ou o atributo `src` declarado;
- caso múltiplos `<source>` sejam informados, o *browser* escolherá a primeira fonte com formato suportado;
- o atributo `controls` deve ser adicionado para que o vídeo seja renderizado;
- atributos `autoplay`, `loop`, `muted`, `poster`, `preload`, `width` e `height`;
- pode possuir um texto como conteúdo para *fallback* (caso o navegador não suporte a *tag*, o texto será exibido no lugar);

# Video: `<video></video>`

Exemplo:

```
<video controls>  
  <source src="arquivo.mp4" type="video/mp4">  
  <source src="arquivo.ogg" type="video/ogg">  
  Seu navegador não suporta a tag de vídeo.  
</video>
```

## Audio: `<audio></audio>`

- utilizada para adicionar audios à página;
- deve possuir ao menos uma tag `<source>` ou o atributo `src` declarado;
- caso múltiplos `<source>` sejam informados, o *browser* escolherá a primeira fonte com formato suportado;
- o atributo `controls` deve ser adicionado para que o vídeo seja renderizado;
- atributos `autoplay`, `loop`, `muted` e `preload`;
- pode possuir um texto como conteúdo para *fallback*;

# Audio: `<audio></audio>`

Exemplo:

```
<audio controls>  
  <source src="arquivo.ogg" type="audio/ogg">  
  <source src="arquivo.mp3" type="audio/mpeg">  
  Seu navegador não suporta a tag de áudio.  
</audio>
```

## Figure: `<figure></figure>`

- utilizado para representar conteúdos independentes, como imagens, ilustrações, trechos de código, diagramas, etc;
- geralmente uma de suas *tags* filhas é um `<figcaption>`;

## Figure Caption: `<figcaption></figcaption>`

- define uma legenda para um elemento `<figure>`;
- pode ser o primeiro ou último elemento do `<figure>`;

# Figure:

Exemplo:

```
<figure>  
    
  <figcaption>Figura 1: pudim</figcaption>  
</figure>
```



## Picture: `<picture></picture>`

- utilizado para definir múltiplas fontes para uma imagem;
- dá aos desenvolvedores web uma maior flexibilidade ao trabalhar com imagens de tamanhos diferentes em telas e formatos diferentes;
- geralmente utilizada para *layouts* responsivos;
- deve possuir ao menos uma *tag* `<source>` e uma *tag* `<img>` ao final como *fallback*;
- pode ser especificada uma resolução ideal para cada fonte, e o browser irá exibir a que melhor se adequar;

## Picture: <picture></picture>

Exemplo:

```
<picture>  
  <source srcset="pudim-grande.jpg" type="(min-width:800px)">  
  <source srcset="pudim-pequeno.jpg" media="(min-width:400px)">  
    
</picture>
```

## Inline Frame: `<iframe></iframe>`

- tag utilizada para incorporar outro documento no HTML do documento atual;
- geralmente utilizado para incorporar vídeos do YouTube, Vimeo e outros;
- deve ser utilizado no `<head>`;
- atributos `src`, `allow`, `loading`, `sandbox`, `width` e `height`;

Exemplo:

```
<iframe src="https://www.google.com/"></iframe>
```

# Elementos de formulários

## Label: `<label></label>`

- tag de rótulo para `<input>`, `<textarea>`, `<select>`, `<meter>` e `<progress>`;
- utilizado para uma melhor UX (*User Experience*) e acessibilidade;
- bastante utilizado em formulários;
- o label pode ser referenciado a um determinado elemento através do atributo `for`, ou colocando o elemento dentro da tag `<label>`;

Exemplo:

```
<label for="campo">Campo</label>
```

# Input: `<input>`

- elemento de campo de entrada, onde o usuário pode inserir dados;
- é o elemento de formulário mais comum e importante;
- quando utilizado em formulários, deve possuir um atributo `name` definido;
- o atributo `id` não é obrigatório, porém é recomendado quando utilizado em conjunto com a tag `<label>`;
- pode haver um valor pré definido através do atributo `value`;
- pode possuir um texto de ajuda com o atributo `placeholder`;
- através do atributo `type`, é possível alterar completamente seu funcionamento e *display*;

## Input: `<input>`

- o tipo `text` é o padrão, porém também é possível definir como um `button`, `checkbox`, `color`, `date`, `datetime-local`, `email`, `file`, `hidden`, `image`, `month`, `number`, `password`, `radio`, `range`, `reset`, `search`, `submit`, `tel`, `time`, `url`, `week`;
- os atributos variam de acordo com o tipo, porém os mais comuns são `name`, `id`, `value`, `type`, `required`, `readonly`, `placeholder` e `disabled`;

Exemplo:

```
<input type="text" id="campo" name="campo">
```

## Text Area: `<textarea></textarea>`

- elemento de controle para inserção de textos com múltiplas linhas;
- o número de linhas e colunas pode ser especificado com os atributos `rows` e `cols`, respectivamente;
- quando utilizado em formulários, deve possuir um atributo `name` definido;
- o atributo `id` não é obrigatório, porém é recomendado quando utilizado em conjunto com a *tag* `<label>`;
- pode haver um valor pré definido entre as tags de abertura e fechamento;
- pode possuir um texto de ajuda com o atributo `placeholder`;
- atributos `autofocus`, `disabled`, `maxlength`, `readonly`, `required` e `wrap`;



# Text Area: `<textarea></textarea>`

Exemplo:

```
<textarea rows="4" cols="50">
```

Esse é o valor pré definido do textarea.

```
</textarea>
```

## Select: `<select></select>`

- elemento de controle utilizado para mostrar uma lista de opções;
- as opções são declaradas com a tag `<option>` que podem ser agrupadas por um `<optgroup>`;
- quando utilizado em formulários, deve possuir um atributo `name` definido;
- o atributo `id` não é obrigatório, porém é recomendado quando utilizado em conjunto com a tag `<label>`;
- pode possuir um valor pré definido através da tag `<option>`;
- atributos `autofocus`, `disabled`, `multiple`, `required` e `size`;

## Option: `<option></option>`

- elemento de opção para uma lista de itens;
- pode ser utilizado dentro das *tags* `<select>` e `<optgroup>`;
- quando utilizado em formulários, deve possuir um atributo `value` definido;
- o atributo `selected` define o valor do `<select>`;
- atributos `disabled` e `label`;

## Option Group: `<optgroup></optgroup>`

- elemento para agrupar opções de uma lista de itens;
- pode ser utilizado dentro da *tag* `<select>` e conter inúmeros `<option>`;
- atributos `disabled` e `label`;

# Select:

Exemplo:

```
<select>  
  <option selected value="audi">Audi</option>  
  <option value="mercedes">Mercedes</option>  
  <option value="gm">General Motors</option>  
  <option value="ford">Ford</option>  
</select>
```

# Select:

Exemplo:

```
<select>  
  <optgroup label="Carros alemães">  
    <option selected value="audi">Audi</option>  
    <option value="mercedes">Mercedes</option>  
  </optgroup>  
  <optgroup label="Carros americanos">  
    <option value="gm">General Motors</option>  
    <option value="ford">Ford</option>  
  </optgroup>  
</select>
```

## Button: `<button></button>`

- elemento de botão, clicável;
- pode conter texto e outras tags como `<i>`, `<b>`, `<strong>`, `<em>`, `<br>`, `<img>`, etc;
- é recomendado especificar o tipo do botão para que o navegador saiba o que fazer quando ele for clicado;
- atributos mais comuns `type`, `autofocus` e `disabled`;

Exemplo:

```
<button>Botão</button>
```

## Fieldset: `<fieldset></fieldset>`

- elemento utilizado para agrupar campos relacionados em um formulário;
- possui uma estilização padrão com bordas;
- geralmente utilizado em conjunto com *checkboxes* e *radio buttons*;



## Legend: `<legend></legend>`

- define uma legenda para um `<fieldset>`;
- possui uma estilização padrão alinhado ao `<fieldset>`;

# Fieldset:

Exemplo:

```
<fieldset>
  <legend>Escolha:</label>
  <div>
    <input type="radio" id="radio-1" name="radio-1" value="1">
    <label for="radio-1">Opção 1</label>
  </div>
  <div>
    <input type="radio" id="radio-2" name="radio-2" value="2">
    <label for="radio-2">Opção 2</label>
  </div>
  <div>
    <input type="radio" id="radio-3" name="radio-3" value="3">
    <label for="radio-3">Opção 3</label>
  </div>
</fieldset>
```

## Form: `<form></form>`

- elemento utilizado como *container* para formulários;
- pode conter vários elementos;
- funciona de forma parecida com a `div`, porém possui algumas funcionalidades, como submeter o formulário ao teclar "enter";
- atributos mais comuns `action`, `method`, `enctype`, `novalidate` e `target`;

# Form:

Exemplo:

```
<form action="/salvar" method="post">
  <label for="texto">Campo 1</label>
  <input type="text" id="texto" name="texto">
  <label for="texto-grande">Campo 2</label>
  <textarea rows="4" cols="50" id="texto-grande" name="texto-grande"></textarea>
  <label for="seletor">Campo 3</label>
  <select id="seletor" name="seletor">
    <option selected value="audi">Audi</option>
    <option value="mercedes">Mercedes</option>
    <option value="gm">General Motors</option>
  </select>
  <button type="submit">Enviar</button>
</form>
```

Elementos de tabelas

## Table: `<table></table>`

- elemento de tabela;
- utilizado para representar dados em duas dimensões ou mais;
- deve possuir pelo menos um `<tr>`, `<th>` e `<td>`;
- pode possuir `<thead>`, `<tbody>`, `<tfoot>`, `<colgroup>` e `<caption>`;

## Table Row: `<tr></tr>`

- elemento de linha de tabela;
- pode estar diretamente dentro de uma `<table>`, ou ainda dentro de um `<thead>`, `<tbody>` OU `<tfoot>`;
- deve possuir um ou mais `<th>` OU `<td>`;

## Table Header Cell: `<th></th>`

- célula de cabeçalho de tabela;
- por padrão, possui estilização de destaque em negrito e centralizada;
- atributos mais comuns: `colspan` e `rowspan`;



## Table Data Cell: `<td></td>`

- célula de dado de tabela;
- atributos mais comuns: `colspan` e `rowspan`;

## Table Header: `<thead></thead>`

- utilizada para agrupar células de cabeçalho de tabela;
- deve ser utilizada em conjunto com `<tbody>` e `<tfoot>`;
- deve possuir pelo menos um `<tr>`;
- pode permitir uma rolagem apenas na tabela;

## Table Body: `<tbody></tbody>`

- utilizada para agrupar células de dados de tabela;
- deve ser utilizada em conjunto com `<thead>` e `<tfoot>`;
- deve possuir pelo menos um `<tr>`;
- pode permitir uma rolagem apenas na tabela;

## Table Footer: `<tfoot></tfoot>`

- utilizada para agrupar células de dados de tabela para o rodapé;
- deve ser utilizada em conjunto com `<thead>` e `<tbody>`;
- deve possuir pelo menos um `<tr>`;
- pode permitir uma rolagem apenas na tabela;

**Caption:** `<caption></caption>`

- utilizado para adicionar uma legenda à tabela;
- deve ser a primeira *tag* filha dentro da `<table>`;

# Tables:

```
<table>
  <tr>
    <th>Cabeçalho coluna 1</th>
    <th>Cabeçalho coluna 2</th>
  </tr>
  <tr>
    <td>Linha 1, coluna 1</td>
    <td>Linha 1, coluna 2</td>
  </tr>
  <tr>
    <td>Linha 2, coluna 1</td>
    <td>Linha 2, coluna 2</td>
  </tr>
</table>
```

# Tables:

```
<table>
  <caption>Minha tabela</caption>
  <thead>
    <tr>
      <th>Cabeçalho coluna 1</th>
      <th>Cabeçalho coluna 2</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>Linha 1, coluna 1</td>
      <td>Linha 1, coluna 2</td>
    </tr>
    <tr>
      <td>Linha 2, coluna 1</td>
      <td>Linha 2, coluna 2</td>
    </tr>
  </tbody>
  <tfoot>
    <tr>
      <td>Rodapé coluna 1</td>
      <td>Rodapé coluna 2</td>
    </tr>
  </tfoot>
</table>
```

# Links úteis

- <https://www.w3schools.com/TAgS/default.asp>
- [https://www.w3schools.com/html/html\\_attributes.asp](https://www.w3schools.com/html/html_attributes.asp)