

# CURSO DESENVOLVIMENTO WEB / HTML

Fundamentos de HTML

**CURSO DESENVOLVIMENTO WEB**  
**André Fontenelle**

<https://www.udemy.com/course/curso-desenvolvimento-website-responsivo-completo/>

# O que vamos aprender

- Introdução à HTML
  - Compreender os principais elementos.
  - Linguagem de marcação.
- Elementos do HEAD
  - Elemento Meta.
  - Aplicar CSS.
    - Incorporado.
    - Externo.
  - Aplicar Javascript.
    - Incorporado.
    - Externo.
- Elementos do BODY.
  - Elementos de Texto.
    - Parágrafo.
    - Títulos.
    - Listas.
    - Hyperlinks.
- Elementos de Layout.
  - Novos elementos semânticos.
  - Elementos não semânticos.
  - ID e Classes.

# Introdução à HTML

**HTML** (Linguagem de Marcação de HiperTexto) é o bloco de construção mais básico da web. Define o significado e a estrutura do conteúdo da web.

Outras tecnologias além do HTML geralmente são usadas para descrever a aparência/apresentação (**CSS**) ou a funcionalidade/comportamento (**JavaScript**) de uma página da web.



**HTML**

[ conteúdo ]



**CSS**

[ forma ]

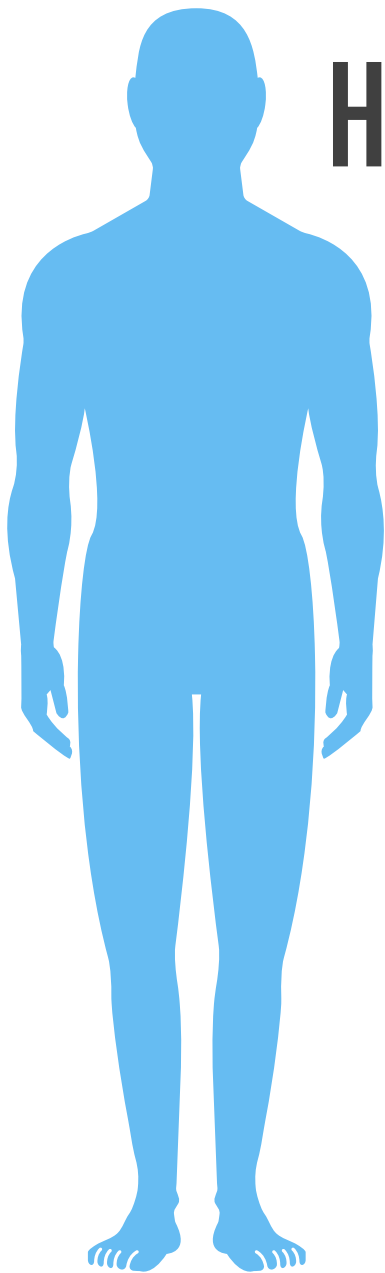


**JS**

[ comportamento ]

Principais linguagens de desenvolvimento Web.

# Compreender os principais elementos do HTML



**HEAD**

**BODY**

Tal como o corpo humano, um arquivo HTML também é dividido entre cabeça e corpo.

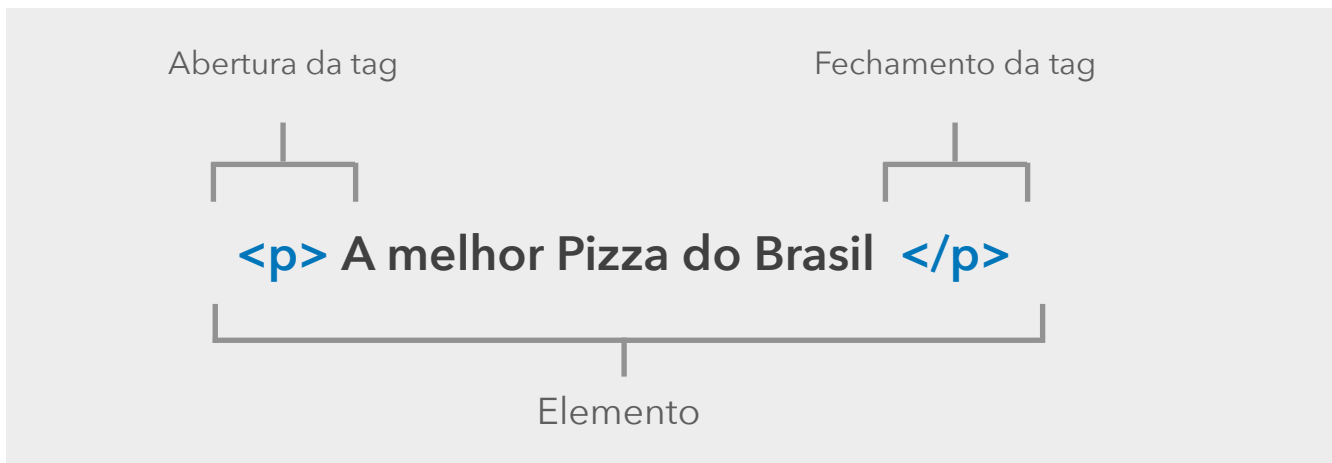
Na cabeça, ficam todos os elementos estruturais da página web.

No corpo, ficam todos os elementos que o usuário vê na página.

# Linguagem de Marcação

Programar em HTML não requer nenhuma lógica de programação, não é necessário conhecimento de algoritmos, tão pouco estruturas condicionais ou de repetição. É uma linguagem simples e fácil compreensão. HTML é uma linguagem de marcação, isto é, formada por tags.

## Anatomia de um tag.



Atributos contém informação extra sobre o elemento. Muitos destes atributos são vitais para o real funcionamento desta tag.



**Importante:** É comum que em uma tag seja necessário o uso de mais de um atributo.

# Elemento HEAD

O **head** de um documento HTML é a parte que não é exibida no navegador da Web quando a página é carregada. Ele contém informações como **title**, links para `<css>` (se você deseja modelar seu conteúdo HTML com CSS), links para favicons personalizados e outros metadados (dados sobre o HTML, como quem o escreveu, e palavras-chave importantes que descrevem o documento.) Neste artigo, abordaremos todas as coisas acima e mais. Dando-lhe uma boa base para lidar com marcação.

## Algumas Tags comuns que ficam no HEAD.

- Title.
- Meta.
- Style.
- Link.
- Script.

## Exemplo

```
<html>
  <head>
    <meta charset="UTF-8">
    <meta name="viewport"
      content="width=device-width, initial-scale=1.0">

    <title>Pizzaria Fontenelle</title>

    <link href="_estilos/layout.css"      rel="stylesheet"/>
    <link href="_estilos/tipografia.css" rel="stylesheet"/>

    <style>
      h1 {
        font-size:200%;
        color:#F00;
      }
    </style>
  </head>

  <body>

    </body>
</html>
```

## Elemento Meta

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

Este elemento especifica a codificação de caracteres do documento — o conjunto de caracteres que o documento está autorizado a usar. `utf-8` é um conjunto de caracteres universal que inclui praticamente qualquer caractere de qualquer linguagem humana. Isso significa que sua página web poderá lidar com a exibição de qualquer idioma; portanto, é uma boa idéia configurar isso em todas as páginas web que você cria.

## Adicionando um autor e descrição

Muitos elementos **<meta>** incluem atributos de **name** e **content**:

- O **name** especifica o tipo de elemento meta que é; que tipo de informação contém.
- O **content** especifica o conteúdo real do meta.

Dois desses meta-elementos que são úteis para incluir na sua página definem o **autor** da página e fornecem uma **descrição** concisa da página. Vejamos um exemplo:

```
<meta name="author" content="André Fontenelle">  
<meta name="description" content="Site de uma pizzeria para uso  
exclusivo de um curso.">
```



## Aplicar CSS

Todos os sites que você usar nos dias atuais empregarão o CSS para torná-los legais e bonitos. No HEAD, o CSS pode ser aplicado diretamente (incorporado) ou carregado de um arquivo externo.

### Incorporado

```
<style>
  div {
    width:400px;
  }
</style>
```

### Externo

```
<link href="layout.css" rel="stylesheet">
```

## Aplicar Javascript

JavaScript ativa funcionalidades interativas, como players de vídeo, mapas, jogos e muito mais. Da mesma forma do CSS, pode ser aplicado incorporada à página ou através de um arquivo externo.

### Incorporado

```
<script>  
    window.open("http://www.adobe.com","_blank");  
</script>
```

### Externo

```
<script src="script.js"></script>
```

# Elemento BODY

O estudo do elemento **body** pode ser dividido em duas etapas:

- Elementos de texto.
- Elementos de layout.

O body por ser a região onde estão os elementos que aparecerão na página web.

O número de tags é muito maior que do head.

Este guia não tem a pretensão de detalhar todas as tags, entretanto abordaremos as mais utilizadas.

Para maiores detalhes, sugiro duas fontes:

- <https://developer.mozilla.org/pt-BR/docs/Aprender/HTML>
- <https://www.w3.org/TR/html52/>

## Elementos de Texto

Há vários elementos de texto no HTML, parágrafos, cabeçalhos, listas, ênfase e citações. O texto mais estruturado é composto por títulos e parágrafos, esteja você lendo uma história, um jornal, um livro da faculdade, uma revista, etc.

### Parágrafo

O conteúdo estruturado torna a experiência de leitura mais fácil e mais agradável.

Em HTML, cada parágrafo deve ser envolvido em um elemento `<p>`, assim:

```
<p>A melhor pizza é servida para você todo  
domingo.</p>
```

## Títulos

Existe seis elementos de título:

- `<h1>`
- `<h2>`
- `<h3>`
- `<h4>`
- `<h5>`
- `<h6>`

Cada elemento representa um nível diferente de conteúdo no documento; `<h1>` representa o título principal, `<h2>` representa subtítulos, `<h3>` representa sub-subtítulos, e assim por diante.

Não adicione `<h7>` ou superior em suas páginas, porque não há a existência desta tag.

No exemplo abaixo, há quatro elementos de texto: **H1**, **H3**, **P** e **LI**. Outro elemento a destacar é o uso da tag **<strong>**, dentro da tag **<p>**. Se quiséssemos afirmar que o sabor calabresa é **muito** importante para destacar o texto, poderíamos envolver a palavra "calabresa" em um elemento **<strong>**, o que significa que a palavra deve ser fortemente enfatizada:

```
<html>
  <head>
  </head>

  <body>
    <h1>Pizza de Calabresa</h1>
    <p>
      É especialmente preparada com recheio
      generoso de linguiça calabresa de primeira qualidade, queijo
      muçarela e orégano. A linguiça <strong>calabresa</strong>
      agrega um sabor único e levemente picante à pizza.
    </p>
    <h3>Vantagens</h3>
    <ul>
      <li>Borda recheada</li>
      <li>Queijo extra</li>
      <li>Massa sem gluten</li>
    </ul>
  </body>
</html>
```

## Listas

Há 3 tipos de listas que você pode fazer no HTML, ordenadas, não ordenadas e descritivas. Neste guia, abordaremos as duas primeiras opções por serem mais utilizadas.

### Lista ordenada (ol)

Na lista ordenada traz algum indicativo de ordem, seja numérico ou de algarismo, o atributo **type** determina qual tipo de caractere simboliza a ordem. A ausência dele, determina que os números serão o elemento ordenador.

```
<ol>
  <li>HTML 5</li>
  <li>CSS 3</li>
  <li>Javascript</li>
</ol>
```

1. HTML 5
2. CSS 3
3. JAVASCRIPT

```
<ol type="a">
  <li>HTML 5</li>
  <li>CSS 3</li>
  <li>Javascript</li>
</ol>
```

- a) HTML 5
- b) CSS 3
- c) JAVASCRIPT

```
<ol type="I">
  <li>HTML 5</li>
  <li>CSS 3</li>
  <li>Javascript</li>
</ol>
```

- I. HTML 5
- II. CSS 3
- III. JAVASCRIPT

## Lista não ordenada (ul)

Na lista não ordenada, não há nenhum indicativo de ordem, seja numérico ou de algarismo. O disco preenchido pela cor preta é o elemento padrão senão for atribuído nenhuma personalização.

Usando o atributo **style**, pode-se determinar via CSS (propriedade list-style-type) o forma da imagem da lista.

```
<ul>
  <li>HTML 5</li>
  <li>CSS 3</li>
  <li>Javascript</li>
</ul>
```

- HTML 5
- CSS 3
- JAVASCRIPT

```
<ul style="list-style-type: square">
  <li>HTML 5</li>
  <li>CSS 3</li>
  <li>Javascript</li>
</ul>
```

- HTML 5
- CSS 3
- JAVASCRIPT

```
<ul style="list-style-type: circle">
  <li>HTML 5</li>
  <li>CSS 3</li>
  <li>Javascript</li>
</ul>
```

- HTML 5
- CSS 3
- JAVASCRIPT

## Criando Hiperlinks

Os hiperlinks são uma das inovações mais interessantes que a Web oferece. Bem, eles são uma característica da Web desde o início, mas são o que torna a Web como ela é — eles nos permitem vincular nossos documentos a qualquer outro documento (ou outro recurso) que queremos. Também podemos vincular para partes específicas de documentos e podemos disponibilizar aplicativos em um endereço web simples.

### Anatomia de um link

Um link básico é criado envolvendo o texto (ou outro conteúdo, veja Block level links) que você quer transformar em um link dentro de um elemento `<a>`, e dando-lhe um atributo `href`, (também conhecido como Hypertext Reference, ou `target`) que conterá o endereço da Web para o qual você deseja que o link aponte.

```
<a href="https://www.mozilla.org/pt-BR/">  
  a página inicial da Mozilla  
</a>
```



# Elementos de Layout

O HTML contém inúmeros elementos para organizar o layout de uma página. A versão 5 da linguagem foi um salto qualitativo nas opções que anteriormente resumia-se apenas a Tag **<div>** e **<span>**.

Para implementar essa marcação semântica, o HTML fornece tags dedicadas que você pode usar para representar essas seções, por exemplo:

cabeçalho: **<header>**.

barra de navegação: **<nav>**.

conteúdo principal: **<main>**, com várias subseções de conteúdo representadas por **<article>**, **<section>**, e elementos **<div>**.

rodapé: **<footer>**.

## Elementos de layout semânticos

**<main>** é para o conteúdo único dessa página. Use **<main>** apenas uma vez por página, e o coloca diretamente dentro do **<body>**. Não é ideal aninhar ele dentro de qualquer outro elemento senão o elemento **<body>**.

**<article>** inclui um bloco de conteúdo relacionado o qual faz sentido por si só, sem o restante da página (por exemplo, uma postagem singular de um blog).

**<section>** é similar com **<article>**, mas ele é mais para agrupar uma única parte de página que constitui em um único pedaço de funcionalidade (por exemplo, um minimapa, ou um conjunto de manchetes e resumo). É considerado boa prática começar cada seção com um título; observe também que você pode dividir um **<article>**s em diferentes **<section>**s, ou **<section>**s em diferentes **<article>**s, dependendo do contexto. **<section>** é menos semântico dos novos elementos do HTML 5.

**<aside>** é para conteúdo que não está relacionados diretamente com os conteúdos principais, mas que podem providenciar informações complementares a esses (entradas de glossários, biografia do autor, links relacionados, etc.).

**<header>** representa um grupo de conteúdo introdutório. Se ele for um elemento filho do **<body>**, Ele é um header global da página do site, mas se for um elemento filho de um **<article>** ou **<section>**, é definido como um header específico para essa seção (tenta não confundir isso com títulos e cabeçalhos).

**<nav>** contém a funcionalidade principal de navegação da página. Links secundários, etc., não iria na navegação

**<footer>** representa um grupo de conteúdo final da página.

Na ilustração abaixo, uma síntese do uso das principais tags de layout.



## Elementos de layout não-semânticos

Além destas tags que são bastante semânticas, há também `<div>`, `<span>` que não tem uma finalidade específica, entretanto são bastante usadas para organização de elementos de layout.

## ID e Classes

No uso de Tags é muito importante o desenvolvedor usar **ID's** e **Classes** para diferenciar uma das outras em casos de haver repetição da ocorrência da mesma tag.

O uso de **ID** é único, não pode haver dois elementos com o mesmo ID na mesma página.

```
<section id="pizza" >  
  
</section>
```

Ao contrário de **classes** a qual você pode aplicar em vários elementos que compartilham das mesmas características.

```
<section class="plano" >  
  
</section>  
  
<section class="plano" >  
  
</section>  
  
<section class="plano" >  
  
</section>
```

# Biografia

- <https://developer.mozilla.org/pt-BR/docs/Aprender/HTML>
- <https://www.w3.org/TR/html52/>