

HTML

Hugo Fumero

- [Introdução](#)
- [Estrutura HTML5](#)
- [Novos Elementos Semânticos](#)
- Formatação
- Formulários
- Elemento Canvas
- Players de áudio e vídeo
- Vínculos (links) e microdata
- Listas
- Tabela
- Tableles
- Introdução ao CSS3
- Montagem de Layout

Introdução

Noções básicas sobre desenvolvimento para ambientes
Web

A *World Wide Web* é uma grande rede de computadores interligados e capazes de compartilhar informações e recursos. Para disponibilizar de modo mais acessível são criadas páginas de *hypertexto* para que o usuário possa interagir com estas informações.

Hipertexto são conjuntos de elementos ou nós ligados por conexões. Estes

elementos podem ser palavras, imagens, vídeos, áudio, documentos dentre outros.

Segundo a *World Wide Web Consortium* (**W3C**) a Web é baseada em 3 pilares:

- Esquema de nomes para localização (URL)
- Protocolo de acesso (HTTP)
- Linguagem de Hypertexto (HTML)

Quem está acostumado com programação voltada para desktop, geralmente trabalha com uma única linguagem. Se você está fazendo um programa em Java, Visual Basic, C# ou Delphi usa a mesma linguagem para a interface quanto para o funcionamento do programa.

Quando trabalhamos na criação de projetos voltados para ambientes web temos uma mudança de paradigma. Trabalhamos com mais de uma linguagem onde cada uma tem um papel a cumprir. Por exemplo o HTML estrutura o seu documento enquanto o CSS é responsável pela formatação ou seja a aparência de uma página da internet.

Existem diversas linguagens mas vamos citar abaixo as mais usadas hoje.

- **HTML** *(atualmente na versão 5)*

Estrutura e organização do documento

- **CSS** *(atualmente na versão 3)*

Formatação e aparência do documento

- **Javascript** *(atualmente na versão 3)*

Responsável pela programação de toda funcionalidade que é executada no navegador por esta razão chamada de client-side ou seja executa do lado do cliente. Permite validar formulários, criar banners, slideshow e processar dados.

- **PHP** *(atualmente na versão 5)*

Responsável pela programação de toda funcionalidade que é executada no servidor por esta razão chamada de server-side ou seja executa do lado do servidor. Permite validar formulários, manipular banco de dados, e processar dados.

HTML é uma abreviação de Hypertext Markup Language - Linguagem de Marcação de Hipertexto, e é de acordo com a World Wide Web Consortium (W3C), é uma linguagem para publicação de conteúdo (texto, imagem, vídeo, áudio e etc) na Web.

Segundo Lee (1994), dentre as linguagens de marcação, ainda é a mais utilizada na Internet, mesmo que muitos a considerem simples e limitada. Sua utilização está voltada tanto para a estruturação de documentos quanto na apresentação visual destes documentos em um navegador.

Embora simples, ela pode representar documentos na Internet da mesma forma que as linguagens mais complexas. Ainda que existam formas de representação mais evoluídas, não se usando apenas tags predefinidas a HTML continua sendo utilizada em larga escala.

Estrutura do HTML5

A estrutura básica do HTML é relativamente simples. Utiliza-se marcadores

chamados de tag's. Cada tag é representada da seguinte forma:

`<html>` - Tag de abertura

`</html>` - Tag de fechamento

A maioria das tags é aberta e fechada como mostrado acima, com o conteúdo entre as duas. Para aqueles que já utilizavam HTML4 é importante ressaltar que houveram algumas mudanças na estrutura básica. Seguem os exemplos do código nas versões 4 e 5 para comparar.

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0  
Transitional//EN"  
http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd>
```

```
<html xmlns="http://www.w3.org/1999/xhtml">
```

```
  <head>
```

```
    <meta http-equiv="Content-Type" content="text/html; charset=utf-8"  
    />
```

```
    <title>Título do Documento</title>
```

```
  </head>
```

```
  <body>
```

Corpo do Documento

```
  </body>
```

```
</html>
```

```
<!doctype html>
```

```
<html>
```

```
  <head>
```

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

```
    <title>Titulo da Página</title>
```

```
  </head>
```

```
  <body>
```

Corpo da Página

```
  </body>
```

```
</html>
```

Vamos começar entendendo as tag's presentes no exemplo.

<!doctype html> Indica que se trata de um documento do tipo HTML ressaltando que se trata da versão 5. Deve ser a primeira tag do código antes mesmo da tag <html>.

<html> Indica que todo seu conteúdo deve ser tratado como um código HTML.

<head> Indica o cabeçalho do documento HTML onde muito de seu conteúdo não é visível ao usuário. Nesta parte ficam os metadados que são informações sobre a página e o conteúdo ali publicado.

<meta charset="utf-8"> Existem mais de uma tag meta. São chamadas de metatags e são responsáveis por passar informações de configuração. A propriedade charset informa a tabela de caracteres. No nosso caso passamos o valor utf-8, que representa o padrão europeu já que nosso idioma tem acentos e outros caracteres especiais de origem europeia.

<title> Esta tag guarda o título do documento geralmente exibido na guia do navegador.

<body> Tudo que está dentro desta tag faz parte do corpo do documento e será exibido na tela.

Conhecendo as Tags

Os elementos HTML são divididos em algumas categorias de acordo com suas similaridades.

- *Metadata content*
- *Flow content*
- *Sectioning content*
- *Heading content*
- *Phrasing content*
- *Embedded content*
- *Interactive content*

As listas que serão mostradas a seguir, estão organizadas de acordo as informações da W3C.

Os elementos desta categoria vem antes da apresentação, formando uma relação com o documento e seu conteúdo com outros documentos que distribuem informação por outros meios.

base

command

link

meta

noscript

script

style

title

A maioria dos elementos utilizados no body e aplicações são categorizados como Flow Content. Existem alguma exceções em que elementos de outra categoria passa a ser considerados Flow Content. Veja a lista abaixo:

- *area (se for um descendente de um elemento de mapa)*
- *style (Se o atributo scopedfor utilizado)*
- *link (Se o atributo itempropfor utilizado)*
- *meta (Se o atributo itempropfor utilizado)*

Basicamente elementos que seu modelo de conteúdo permitem inserir qualquer elemento que se encaixa no Flow Content, devem ter pelo menos um descendente de texto ou um elemento descendente que faça parte da categoria embedded.

Flow Content

<i>a</i>	<i>canvas</i>	<i>fieldset</i>	<i>header</i>	<i>keygen</i>
<i>abbr</i>	<i>cite</i>	<i>Figure</i>		
<i>address</i>	<i>code</i>	<i>Footer</i>	<i>hgroup</i>	<i>label</i>
<i>area</i>	<i>command</i>		<i>hr</i>	
<i>article</i>	<i>datalist</i>	<i>form</i>		
<i>aside</i>	<i>del</i>	<i>h1</i>	<i>i</i>	
<i>audio</i>	<i>details</i>	<i>h2</i>	<i>iframe</i>	
<i>b</i>	<i>dfn</i>	<i>h3</i>	<i>img</i>	
<i>bdo</i>	<i>div</i>		<i>input</i>	
<i>blockquote</i>	<i>dl</i>	<i>h4</i>		
<i>br</i>	<i>em</i>	<i>h5</i>	<i>Ins</i>	
<i>button</i>	<i>embed</i>	<i>h6</i>	<i>Kbd</i>	

<i>Link</i>	<i>object</i>	script	table
<i>map</i>	<i>o</i>	section	textarea
<i>mark</i>	output	select	time
<i>math</i>	p	small	ul
<i>Menu</i>	pre	span	var
<i>Meta</i>	progress	strong	video
<i>meter</i>	q	sub	wbr
<i>nav</i>	ruby	sup	Text
<i>noscript</i>	samp	svg	

Estes elementos definem um grupo de cabeçalhos e rodapés. São elementos que juntam grupos de textos no documento

- article
- aside
- nav
- section

Estes elementos definem uma seção de cabeçalhos, que podem estar contidos em um elemento na categoria Sectioning.

- h1
- h2
- h3
- h4
- h5
- h6
- hgroup

Fazem parte desta categoria elementos que marcam o texto do documento, bem como os elementos que marcam este texto dentro do elemento de parágrafo.

Existem alguns elementos que só se enquadram nestas categorias em condições especiais.

- *area* (se ele for descendente de um elemento de mapa)
- *del* (se ele contiver um elemento da categoria de Phrasing)
- *ins* (se ele contiver um elemento da categoria de Phrasing)
- *link* (se o atributo *itemprop* for utilizado)
- *map* (se apenas ele contiver um elemento da categoria de Phrasing)
- *meta* (se o atributo *itemprop* for utilizado)

<i>a</i>	<i>command</i>	<i>keygen</i>	<i>ruby</i>	<i>textarea</i>
<i>abbr</i>	<i>datalist</i>	<i>Label</i>	<i>samp</i>	<i>time</i>
<i>audio</i>	<i>dfn</i>	<i>mark</i>	<i>script</i>	<i>var</i>
<i>b</i>	<i>em</i>	<i>Math</i>	<i>select</i>	<i>video</i>
<i>bdo</i>	<i>embed</i>	<i>meter</i>	<i>small</i>	<i>wbr</i>
<i>br</i>	<i>i</i>	<i>noscript</i>	<i>span</i>	<i>Text</i>
<i>button</i>	<i>iframe</i>	<i>object</i>	<i>strong</i>	
<i>canvas</i>	<i>img</i>	<i>output</i>	<i>sub</i>	
<i>cite</i>	<i>input</i>	<i>progress</i>	<i>sup</i>	
<i>code</i>	<i>kbd</i>	<i>q</i>	<i>svg</i>	

Nessa categoria há elementos que importam outra fonte de informação para o documento.

- audio
- canvas
- embed
- iframe
- img
- math
- object
- svg
- video

Os elementos desta classe fazem parte da interação de usuário. Alguns elementos no HTML podem ser ativados por um comportamento. Isso significa que o usuário pode ativá-lo de alguma forma.

O início da sequência de eventos mecanismo de depende do teclado, mouse, ativação por voz e outros dependendo do user-agente, que permite que usuário ative manualmente.

São eles:

- *a*
- *audio* (se o atributo *control* for utilizado)
- *button*
- *details*

- *Embed*
- *frame*
- *img* (se o atributo *usemap* for utilizado)
- *input* (se o atributo *type* não tiver o valor *hidden*)
- *keygen*
- *label*
- *menu* (se o atributo *type* tiver o valor *toolbar*)
- *object* (se o atributo *usemap* for utilizado)
- *select*
- *textarea*
- *video* (se o atributo *control* for utilizado)

Elementos Semânticos

Novidades do HTML5

Uma das novidades do HTML5 é que agora temos novas tags que funcionam como elementos semânticos. Isto além de organizar o código permite que os buscadores como o Google possa ler o conteúdo de uma página com mais eficiência.

As versões anteriores do HTML não continham um padrão para a criação de seções comuns e específicas como rodapé, cabeçalho, sidebar, menus e etc. Frequentemente era necessário usar uma tag `<div>` para cada elemento onde trabalhávamos com ID e classes para identificá-los. Também não havia um padrão de nomenclatura de IDs, classes ou tags. Veja o exemplo em HTML 4.

```
<!doctype html>
```

```
<html>
```

```
  <head>
```

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

```
    <title>Titulo da Página</title>
```

```
  </head>
```

```
  <body>
```

```
    <div id="header">
```

Cabeçalho

```
    <div id="navbar"> Menu </div> <!-- FIM DE NAVBAR -->
```

```
  </div> <!-- FIM DE HEADER -->
```

```
  <div id="content"> Conteúdo </div> <!-- FIM DE CONTENT -->
```

```
  <div id="sidebar"> Barra lateral </div> <!-- FIM DE SIDEBAR -->
```

```
  <div id="footer"> Roda Pé </div> <!-- FIM DE FOOTER -->
```

```
</body>
```

```
</html>
```

*Exemplo de Estrutura utilizando apenas
A tag <div>.*

O HTML5 modifica a forma de como escrevemos código e organizamos a informação na página. Organizada com novas tags de forma mais semântica com menos código. Deste modo conseguimos mais interatividade sem a necessidade de instalação de plugin's o que pode levar a perda de performance.

Ainda sim podemos ter problemas de compatibilidade de código que podem ocorrer com mais frequência o que depende dos fabricantes dos navegadores. Por exemplo, um navegador pode adotar bordas arredondadas e outro não.

Estas são algumas das novas tags:

<article> Define uma artigo ou conteúdo de texto

<aside> Esta tag demarca a área para uma barra lateral

<header> Define um cabeçalho de conteúdo

<nav> Marca uma barra de navegação seja ela principal ou secundária.

<section> Define uma nova seção genérica no documento

<hgroup> Este elemento consiste em um grupo de títulos.

<footer> Define um rodapé de conteúdo

<time> Marca parte do texto que exibe um horário ou uma data

Elementos Semânticos – Exemplo 1



Esta é a organização do código do exemplo 1, que você verá no próximo slide.

Elementos Semânticos – Exemplo 1

```
<!doctype html>
```

```
<html>
```

```
  <head>
```

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

```
    <title>Titulo da Página</title>
```

```
  </head>
```

```
  <body>
```

```
    <header> Cabeçalho </header>
```

```
    <nav> Menu </nav>
```

```
    <article> Conteúdo </article>
```

```
    <aside> Barra lateral </aside>
```

```
    <footer> Roda Pé  
    </footer>
```

```
  </body>
```

```
</html>
```

Agora observe o exemplo utilizando as novas tags e observe como ficou mais limpo e organizado. Podemos ainda dizer que está semanticamente correto.

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title> Título da Página </title>
  </head>
  <body>
    <header>
      Cabeçalho

      <nav> Menu </nav>
    </header>
    <hgroup>
      <h1> Título 1 </h1>
      <h2> Título 2 </h2>
    </hgroup>
    <section> Secção 1 </section>
    <section> Secção 2 </section>
    <footer> Roda Pé </footer>
  </body>
</html>
```

Observe este outro exemplo onde trabalhamos com a tag `section` para dividir secções de conteúdo.

Também utilizamos a tag `hgroup` para organizar os grupos de cabeçalhos , onde haverão títulos e subtítulos.

Elementos Modificados

`` Passa a ter o mesmo nível semântico que um SPAN, mas ainda mantém o estilo de negrito no texto.

`<i>` também passa a ser um SPAN. O texto continua sendo itálico e para usuários de leitores de tela, a voz utilizada é modificada para indicar ênfase.

`<a>` Quando estiver sem o atributo href representa um placeholder no lugar que este link se encontra.

`<address>` Agora é tratado como uma seção no documento.

`<hr>` agora tem o mesmo nível que um parágrafo, mas é utilizado para quebrar linhas e fazer separações.

`` Ganhou mais importância.

`<head>` não aceita mais elementos child como seu filho.