

[< VOLTAR](#)

HTML5: Princípios importantes

Cada vez mais sistemas têm sido desenvolvidos para a web, é um novo paradigma de programação e uma nova forma de pensar, além de ser uma tendência mundial. O HTML5 chegou para ficar e é uma das principais linguagens modernas para desenvolvimento de interfaces web. Veremos, neste tópico os principais conceitos e características do famoso HTML5.

NESTE TÓPICO

- › Introdução
- › O que saiu e o que entrou no HTML5
- › HTML5 - Um pouco de código
- › Referências



Introdução



É chegada a hora de colocar um pouquinho a mão em código fonte para desenvolvermos interfaces de qualidade.

Cada vez mais sistemas têm sido desenvolvidos para a web, com recursos de portabilidade. Este é um novo paradigma de programação e uma nova forma de pensar, além de ser uma tendência mundial.

O HTML5 chegou para ficar, mas é importante salientar que a linguagem está em constante evolução, ou seja, provavelmente ainda teremos novas variações dessa linguagem, como HTML 6 etc.

O HTML5 é uma das principais linguagens modernas para desenvolvimento de interfaces web. Lembre-se que usamos o HTML, JavaScript e CSS normalmente para a programação na camada de visualização (**View**) em modelos MVC (**Model - View - Control**), ou seja, o HTML5 tem se tornada cada vez mais a linguagem para desenvolvimento de interfaces.

Parece bobeira, mas é importante que você se familiarize com alguns logotipos importantes que envolvem essa tecnologia, como:



Logotipos das novas tecnologias

Mas qual a diferença do HTML5 para as versões anteriores? Precisaréi aprender uma linguagem totalmente nova? Não, muita calma, não será preciso aprender uma linguagem totalmente nova, assim como as diferenças não são tão absurdas ou assustadores. Não se preocupe. HTML continua significando HyperText Markup Language (Linguagem de Marcação de Hipertexto). Apenas a versão mudou, não a linguagem inteira.

Como toda evolução de linguagem de programação, o HTML5 nos traz novas TAGS e retirou algumas defasadas. Vamos entender melhor as novas tags e as retiradas no decorrer deste tópico.

Mas é importante salientar que uma noção de HTML clássico (HTML4, por exemplo) é importante para o desenvolvimento e acompanhamento deste tópico. É importante ter em mente que você não aprenderá programar usando HTML de forma avançada aqui, pois é uma habilidade que você desenvolverá ao longo do tempo, praticando bastante.

Caso você queria aprender HTML5 completo (desde o básico, para lembrar, por exemplo) com todas as TAGS e recursos, um bom site é o W3Schools, uma referência mundial em padrões de programação, desenvolvido pela W3C, ou seja, *World Wide Web Consortium* um dos comitês máximos de padrões Web. O site para o HTML5 é <http://www.w3schools.com/html/> (<http://www.w3schools.com/html/>). Uma coisa muito interessante é que o próprio site possui um pequeno editor de código que você pode simular seus exercícios no próprio browser. Vale a penas conferir.

E qual editor usar? Bem, o bom e velho “bloco de notas” continua sendo um poderoso editor. Sinta-se livre para utilizar qualquer editor, dos mais simples até os mais rebuscados. Alguns exemplos de editores:

- Bloco de notas - O bom e velho “notepad” do Windows é um excelente editor, pois é simples, leve, rápido e gratuito (no Windows);
- Notepad++ - É um excelente editor gratuito também, bastante leve e que permite uma visualização melhor do código;



- TextWrangler - É um editor gratuito para sistemas operacionais Apple (MAC), bastante parecido com o Notepad++, com possibilidade de colocação do código, organização dos arquivos etc.;
- <https://html-online.com/editor/> (<https://html-online.com/editor/>) - É um simples editor online, bastante interessante pois o código é compilado diretamente no browser. É excelente para uma visualização e testes rápidos;
- GDit - Aplicativo nativo do Gnome (Linux), bastante poderoso e leve;
- etc.



Exemplo de editor HTML - Bloco de notas e execução

O HTML5 também ajudou no processo de extinção de recursos em Flash. Além de uma tendência criada pelas maiores empresas de internet e tecnologia do mundo (Google, Apple, Microsoft, Oracle etc.) que declararam, abertamente, o abandono ao suporte a essa tecnologia. O HTML5 já não prevê mais a inclusão de recursos em Flash. O que é bom pois são recursos ultrapassados (obsoletos) que consomem banda, memória, processamento e nem sempre funcionam, adequadamente em todos os dispositivos. A própria Adobe abandonou o desenvolvimento da ferramenta, em 2011.



O que saiu e o que entrou no HTML5

Como mencionado anteriormente, algumas TAGS foram descontinuadas (tornadas obsoletas) e outras foram acrescentadas. Isso não quer dizer que as TAGS antigas pararam de funcionar completamente, pois os browsers ainda conseguem interpretar as versões anteriores do HTML, ou seja, as TAGS removidas ainda funcionam, apenas não são mais recomendadas.

Alguns pontos importantes a se destacarem sobre o que entrou ou saiu da nova versão da linguagem:

- **Entrou:** Algumas TAGs especiais novas, como < audio >, < video >, < canvas >, etc., para adicionar mais suporte a recursos multimídia e gráficos. Isso ajudou a tornar algumas ferramentas obsoletas como Flash, por exemplo;
- **Entrou:** Novos recursos para suportar fórmulas matemáticas;
- **Entrou:** Novas TAGs especiais para divisão semântica do conteúdo na página, como < section >, < article >, < header >, < footer >, < aside >, < nav >

etc. Essas TAGs novas não necessariamente alteram a visualização na tela, mas auxiliam muito os desenvolvedores web a organizar o código. Alguns frameworks modernos (como Bootstrap, por exemplo) possuem funcionalidades que requerem essas TAGs novas.

- **Entrou:** Novidade! Agora o HTML5 suporta algumas propriedades que antes eram nativas do *jQuery*, como o DOM (*Document Object Model*).
- **Mudou:** A performance do processamento de elementos de *input* para formulários, onde o carregamento se tornou mais eficiente e mais preciso;
- **Manteve:** TAGs clássicas como a de link (< a >), citação (< cite >), menus (< menu >) foram mantidas, apenas ganharam uma nova forma de serem processadas, para garantia de mais eficiência no processo;
- **Saiu:** Algumas TAGs defasadas como < center > e < font > se tornaram obsoletas, obrigando a formatação ser feita através e exclusivamente pelo CSS (*Cascading Style Sheets*).

Muitas mudanças foram conseguidas com o HTML5, que é, de fato, uma revolução. De qualquer forma, a forma de escrever código, a carinha do código e os recursos clássicos foram mantidos. A nova versão não tornou a linguagem completamente diferente, mas sim mais robusta e mais adequada às novas tecnologias.

Para lhe situar um pouco na história, veja a tabela abaixo onde são mostrados os anos os quais cada versão do HTML foi homologada:

Versão	Ano
HTML	1991
HTML 2.0	1995
HTML 3.2	1997
HTML 4.01	1999
XHTML	2000
HTML5	2014

Ano de homologação
das versões do HTML

Como você pode notar, o HTML5 teve sua homologação em 2014, o que é bastante recente e hoje já é um padrão para desenvolvimento web (lado cliente).

LEMBRE-SE!

É importante lembrar que o HTML é uma linguagem interpretada exclusivamente pelo browser, ou seja, não requer um servidor de aplicação para ser executado, apenas para ser hospedado. Por isso falamos que HTML roda no lado do cliente, ou seja, quem solicitou a página (normalmente o usuário). Todos os browsers modernos (Chrome, Edge, Firefox, Safari etc.) já estão preparados para o HTML5!

HTML5 - Um pouco de código

Bem, depois de aprendido um pouco sobre as grandes novidades do HTML5, é hora de colocar um pouco a mão na massa e criar algumas coisas.

Vamos, por enquanto, evitar de usar o CSS pois este é um assunto para outro tópico, ou seja, vamos, nessa parte, ver (ou rever) alguns conceitos importantes da linguagem.

Declaração DOCTYPE

A declaração DOCTYPE raramente é usada pelos desenvolvedores web, mas é muito importante. Ela deve ser a primeira declaração em um arquivo HTML, ou seja, deve constar na primeira linha, somente uma vez em cada documento e antes de todas as TAGs.

Essa declaração representa o tipo de documento que é descrito, ajudando os browsers a exibir corretamente e de forma mais eficiente as páginas.

A sintaxe da declaração é:

```
1. <!DOCTYPE html>
```

TAGs

É muito importante lembrar que todas as TAGs HTML andam em pares, ou seja, elas precisam ser abertas e fechadas. Sempre que você abrir uma TAG, não se esqueça de fechá-la. Para não errar, abra e feche a TAG antes de colocar elementos dentro dela.

Por exemplo:

```
1. <p>Olá mundo!</p>  
2. <h1>Título</h1>
```

Dizemos que uma TAG é aberta quando o símbolo de “menor” aparece seguido do nome da TAG e o símbolo de “maior”. O fechamento deve sempre ocorrer com o mesmo nome da TAG (em minúsculo) com uma barra invertida no início (depois do símbolo de “menor”).

Algumas TAGs especiais não possuem uma de fechamento como, por exemplo:



```
1. <br />
```

Como você pode observar, essa TAG, que serve para quebrar uma linha, não possui um par, por isso foi aberta e fechada dentro da própria TAG.

Estrutura de uma página

É sempre bom lembrar a estrutura básica de uma página HTML. Ela é composta por delimitadores da linguagem, um cabeçalho e o corpo. Assim:

```
1. <!DOCTYPE html>
2.     <html lang="pt-br">
3.         <head>
4.             <title>Título da página</title>
5.         </head>
6.         <body>
7.             <!-- Essa é a área onde desenvolvemos o código -->
8.         </body>
9.     </html>
```

Repare, no código acima, que fora usando um comentário. O HTML implementa comentários há muito tempo, mas é sempre bom lembrar suas sintaxe:

```
1. <!-- Comentários em HTML são assim! -->
```

Repare também que na primeira TAG de abertura do documento (< html >) utilizou-se o parâmetro "lang". Este é um parâmetro importante para sistemas de busca e leitores de tela para acessibilidade, por exemplo. Ele define que a linguagem do documento.

Os comentários são muito úteis quando estamos desenvolvendo pois nos ajudam a identificar coisas, lembrar onde colocamos determinada declaração, documentar o código etc. Além, claro, de servir para retirar elementos (TAGs) HTML que não devem ser renderizadas num determinado momento.

Repare, também, na indentação do código. Embora em HTML não existam boas práticas de programação que definem como o código deve ser indentado, o fazer pode lhe ajudar bastante na hora de organizar os elementos, especialmente sabendo que seu código pode (e normalmente vai) ficar bastante extenso. Indentação correta nunca é demais!

Títulos e parágrafos, links e imagens. Nada de novo aqui.

O HTML5 ainda implementa os títulos de 1 a 6, ou seja, a TAG de Headings. Tente implementar o seguinte código para testar:



```
1. <!DOCTYPE html>
2. <html lang="pt-br">
3.     <head>
4.         <title>Teste de títulos</title>
5.     </head>
6.     <body>
7.         <h1>Este é um Título 1</h1>
8.         <h2>Este é um Título 2</h2>
9.         <h3>Este é um Título 3</h3>
10.        <h4>Este é um Título 4</h4>
11.        <h5>Este é um Título 5</h5>
12.        <h6>Este é um Título 6</h6>
13.    </body>
14. </html>
```

Além dos títulos, nada mudou em relação à clássica TAG de parágrafo, como exemplo:

```
1. <p>Este é um parágrafo!</p>
2. <p>Este é outro parágrafo</p>
```

Nada mudou, também, em relação aos links, como exemplo:

```
1. <a href="http://www.uninove.br">Clique aqui para ir para o site da Uninove</a>
```

Repare que o parâmetro do endereço do link é colocado dentro da TAG. O endereço (destino) é colocado no parâmetro “href”.



Os parâmetros são usados para fornecer ao browser informações adicionais da TAG específica.

Bem, as imagens continuam com a mesma forma de declaração, como exemplo:

```
1. 
```

Repare que foram usados três parâmetros na TAG acima, o “src” que indica onde está a imagem (caminho dela no servidor ou no projeto), o “alt” que acrescenta um texto alternativo caso a imagem não seja encontrada pelo browser e as especificações de tamanho. Lembre-se que ao alterar as especificações de tamanho a imagem pode ficar distorcida, portanto, pode-se usar apenas um deles para realizar o ajuste, neste caso, em pixels.

Mas quando usar aspas simples ou duplas? Bem, esta é uma dúvida comum! Bem, para atributos é preciso utilizar aspas duplas, contudo, se o conteúdo do atributo possui aspas, então utilize aspas simples. Assim:

Por padrão:

```
1. <p title='A "Uninove" com você é dez'>Uninove</p>
2.
3. <!-- ou vice-versa: -->
4.
5. <p title="A 'Uninove' com você é dez">Uninove</p>
```

De qualquer forma é muito importante não se esquecer de colocar as aspas, na abertura e no fechamento, para que o browser não tenha problema em interpretar o código.

Para fechar o conceito de atributos, veja a tabela abaixo com os atributos mais comuns do HTML5:

Atributo	Descrição
alt	Texto alternativo para imagens quando a imagem não puder ser carregada.
disabled	Torna um elemento de input desabilitado
href	Endereço destino de um link (para a TAG < a >
id	O ID único de um determinado elemento
src	Endereço de uma imagem
style	Aplica o estilo <i>inline</i> (no código) a uma TAG
title	Elemento extra para TAGs, normalmente para ToolTipText

Atributos mais comuns em HTML

Se quiser ver a lista completa de atributos e TAGs relacionadas do HTML5, acesse este link (http://www.w3schools.com/tags/ref_attributes.asp) (http://www.w3schools.com/tags/ref_attributes.asp)



Chegamos ao final de mais um importante tópico, este sobre HTML5 seus princípios, TAGs que entraram (novas), TAGs que saíram (foram apreciadas) e TAGs que foram atualizadas. Conforme mencionado, o HTML5 é uma linguagem que chegou para ficar e é uma tendência mundial que perdurará por bastante tempo.

Quiz

Exercício Final

HTML5: Princípios importantes

INICIAR ➤

Referências

W3C. HTML, The Web Core Language. Disponível em: <<http://www.w3.org/html/>>. Acesso em: 4 de Novembro de 2016

BATISTA, E. A.; MANOCHIO, K. D. Disponibilizando Dados de um sistema escrito em COBOL para serem consultados em ambiente WEB: Um Estudo de Caso. **Revista Network Technologies**, v. 3, n. 1, p. 1-12, 2009. ISSN: 1809-5240.

BIBEAULT, B.; KATZ, Y. *jQuery in Action*. 2 ed. Greenwich, CT: Manning Publications Co., 2008. p. 475. ISBN 1-935182-32-3.

DE SOUZA, Edson Melo. *Desenvolvimento de um Sistema de Apoio à Decisão e Operacional para a Otimização dos Parâmetros de Corte em Usinagem (COPPISYS)*. 2013. 149 f. Dissertação (Mestrado em Engenharia de Produção) – Programa de Mestrado em Engenharia de Produção, Universidade Nove de Julho, São Paulo, 2013

TUTORIAL WEB DESIGN. Tutorial Bootstrap. Disponível em: <<http://www.tutorialwebdesign.com.br>>. Acesso em 4 de Novembro de 2016



Avalie este tópico



ANTERIOR

Técnicas de Apresentação



Índice

Biblioteca
(<https://www.uninove.br/conhec-a-uninove/biblioteca/sobre-a-biblioteca/apresentacao/>)
Portal Uninove
(<http://www.uninove.br>)
Mapa do Site

CSS3: Princípios importantes

© Todos os direitos reservados

Ajuda?
(<https://ava.uninove.br/curso/>)

