

HTML5

~ em ~

15 minutos

O que você precisa saber para migrar do
HTML4 / xHTML ao HTML5 rapidamente

< 2ª edição >
<autor>Canha</autor>

Sobre este eBook

Primeiramente, gostaria de agradecer a todos que fizeram o download da primeira edição e ajudaram a compartilhar ela. Quero também agradecer a todos os leitores fiéis do Design Blog, seguidores do Twitter e fãs do Facebook, além dos estudantes e entusiastas do design que caem de para-que-das e ficam pra ver o restante do show. Por último, mas não menos importante, quero agradecer a você, leitor, por ter feito o download deste eBook.

O objetivo do eBook “HTML5 em 15 minutos” é de fazer um resumo rápido do que o HTML5 tem a oferecer. Nesta segunda edição, vou expandir um pouco mais o texto sem sacrificar o tempo necessário para ler ele.

Neste eBook você vai encontrar código HTML. Eu separei os códigos por cor para melhor entender o que é novo, o que é já conhecido e o que é conteúdo:

Códigos HTML nesta cor são novidades do HTML5;

Códigos nesta cor são conhecidos das versões anteriores;

Códigos nesta cor são comentários e nesta, é conteúdo.

Para finalizar, gostaria de lembrar que este eBook é gratuito e ele é distribuído sob a [Licença Creative Commons - Atribuição - Não Comercial - Compartilha Igual 3.0 Não Adaptada](#). Ou seja, você pode editar (desde que distribua gratuitamente sob a mesma licença e mantenha os créditos ao autor original), pode compartilhar com seus amigos, famílias, no seu site, blog ou portal desde que você não ganhe dinheiro com isto.

Se você comprou este eBook de alguém ou teve que pagar por ele de qualquer forma, por favor denuncie imediatamente entrando em contato comigo pelo e-mail **canha@design.blog.br**.

Sejamos breves e continuemos.

Abraços,
Canha.

Introdução ao HTML5

Estamos na era do “layout responsivo” (ou “responsive design”), onde sites precisam adaptar os seus layouts para funcionarem em quaisquer tipos de dispositivos; desde as telas menores dos *smartphones*, passando pelas telas das *tablets* um pouco maiores e fechando com os *desktops* que vem em várias resoluções diferentes.

Não é prático desenvolver um layout para cada resolução, mas com o advento do HTML5, criar um layout que se adapte a qualquer resolução de tela ficou muito mais fácil.

No HTML5, o código fica mais limpo e mais legível – aumentando a produtividade dos web designers e diminuindo as dores de cabeça para caçar eventuais erros.

O HTML5 já é suportado pelos navegadores mais populares nas suas versões atuais; até o Internet Explorer na sua versão 9 roda o HTML5 relativamente bem! (Nunca sonhei que iria falar bem desse navegador)

A sintaxe do HTML5 é compatível com o HTML4 e XHTML. Quer fechar elementos com uma barra? Beleza. Não quer? Sem problemas. Prefere escrever em maiúsculas? Minúsculas? Tanto faz! Ou seja, você não precisa mudar o seu jeito de codificar, então não se preocupe. O HTML5 é altamente adaptável e pra quem já possui conhecimento de XHTML / HTML4, aprender as novidades “vai ser sopa”.

Para que você consiga aprender HTML5 em 15 minutos, você vai precisar de alguns itens antes:

- Um computador (dã)
- Um navegador atualizado (Chrome, Safari, Firefox, Opera ou o IE9)
- Um editor HTML simples
- Conhecimento básico de HTML4 ou XHTML1 (opcional)
- 15 minutos

Doctype

Vamos começar pelo topo, no Doctype. Ele não é uma tag ou elemento mas sim uma declaração que você encontra no topo de qualquer documento HTML. Em termos simplificados, ele serve para avisar o navegador que tipo de documento ele é.

Para cada versão do HTML (HTML 4.01, XHTML 1.0, XHTML 1.1), existe um Doctype específico. A declaração usada no HTML5 é a mais simples de todas, consistindo de apenas 15 caracteres:

Versões antigas (neste exemplo, HTML 4.01):

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

HTML5:

```
<!doctype html>
```

Esse código sempre vai no topo da sua página, antes de qualquer outra coisa. Isto é importante para que o navegador saiba como ele vai interpretar o seu HTML.

Dica importante:



O Doctype deve ficar sempre no topo da página, antes de qualquer outro conteúdo. Isto é importante para o navegador saber como interpretar seu site visualmente.

HTML

Logo após o *Doctype*, é onde vamos abrir a tag `<html>` para que o navegador saiba que é a partir daqui que ele vai interpretar os códigos do arquivo e mostrar na tela do usuário.

É nessa tag também que você pode definir a língua do seu conteúdo e a direção de texto (esquerda pra direita ou direita pra esquerda).

No HTML 4.01, o padrão é similar ao HTML5. Mas no xHTML, existe uma leve diferença, sendo que a maior parte do código é removida.

Versão antiga (neste exemplo, xHTML):

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

HTML5:

```
<html lang="pt-BR" dir="ltr">
```

O “pt-BR” indica que a página está em português do Brasil. Se fosse português de Portugal, seria apenas “pt”. Em inglês americano, “en-US”. Inglês internacional, “en”. A lista completa de códigos de língua você encontra no [site da W3C](#).

Já o “ltr” que indica a direção do texto, significa “left to right” (da esquerda pra direita, o padrão ocidental), mas pode ser substituído por “rtl” (direita para esquerda) para línguas orientais.

Logo após a tag `<html>`, vamos inserir a tag `<head>` que vai conter as informações de cabeçalho do site. Essa tag não muda no HTML5.

Meta Charset

O *Meta Charset* define quais caracteres abstratos fazem parte do HTML. É graças ao charser que, se você digitar uma palavra acentuada no seu site, ele vai garantir que a palavra apareça de forma correta.

Se você escrever a palavra “além” e o Charset estiver definido como “ISO-8859-1” (por exemplo) , você vai notar que no seu navegador ele pode mostrar a palavra como “al[]m”. Neste caso, você terá que repor a acentuação para que fique “além”.

Para evitar isto, tradicionalmente usamos o *Charset* “UTF-8” em sites latinos ou não-ingleses.

O código do *Meta Charset* no HTML5 ficou mais simples e deve ser inserido logo após a abertura do `<head>`.

Versões antigas:

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

HTML5:

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

Outros elementos

Os outros elementos de cabeçalho das versões anteriores do HTML ainda funcionam, como as chamadas para JavaScript e folha de estilos CSS:

```
<script type="text/javascript" src="javascript.js" />
<link rel="stylesheet" href="estilo.css" />
<script type="text/javascript">
    alert("Código JS inline");
</script>
<style type="text/css">
    #exemplo {border:1px solid red;}
</style>
```

E assim por diante.

Estrutura

Antes do HTML5, DIVs eram usadas para estruturar um site. Agora, ele pode voltar a ser usado como um elemento de estilização, pois é possível substituí-lo pelos novos elementos que o HTML5 dispõe.

O HTML5 inclui vários elementos estruturais que podem ajudar a definir as partes de um documento. Vou focar nos elementos suportados pelos navegadores mais populares.

Principais elementos estruturais do HTML5:

- `<header>`
- `<nav>`
- `<section>`
- `<article>`
- `<aside>`
- `<footer>`

Esses elementos servem mais para agrupar elementos do que para posicionar eles. Ou seja, você pode ter várias `<section>` em uma página, cada uma tendo a sua própria `<header>` e `<footer>`. São como classes no CSS: podem ser utilizadas mais de uma vez na página.

Os nomes destes elementos são auto-explicativos (assumindo que você saiba o básico de inglês), mas vamos lá definir eles:

- `<header>` – Cabeçalho. Aqui você pode colocar informações importantes como o nome ou logo do site.
- `<nav>` – Navegação. Ou seja, o menu.
- `<section>` – Seção. Você pode ter uma seção para o artigo em destaque, outra para artigos de uma determinada categoria do site, etc.
- `<article>` - Artigo. Em um blog seria o equivalente a um post.
- `<aside>` - Lateral. Representa o conteúdo relacionado da página.
- `<footer>` - Rodapé. Pode conter informações sobre o autor, copyright, ou até mesmo uma navegação de rodapé.

Isto não quer dizer que cada elemento deve obrigatoriamente ser usada para o fim pela qual ela foi criada. O uso é livre. O <aside> por exemplo não precisa ser usado como barra lateral, podendo ser usado como apenas uma seção para colocar informações adicionais pertinentes.

Esses elementos todos servem apenas para facilitar a vida de quem está criando o conteúdo do site e facilitar a vida de quem está criando o estilo do site pela folha de CSS.

Cada elemento também pode ter uma ID ou classe atribuída a ele, como você faria tradicionalmente com as DIVs. Vale notar que a mesma regra de ID se aplica no HTML5: só pode existir um ID único para cada elemento. Se quiser estilizar vários elementos com um mesmo estilo, utilize a classe.

Até agora não tem muito mistério, né? Para que nenhuma dúvida fique no ar, vou criar um exemplo de site simples com um cabeçalho, menu, área com dois artigos, uma barra lateral e um rodapé:

Dica importante:



Não confunda o <head> com <header>. O primeiro é um elemento informativo, o segundo é elemento de estilo. O <title>, por exemplo, só pode ir no <head> e não no <header>.

Exemplo de HTML5

```
<!doctype html>
<html lang="pt-br">
<head>
    <!-- É nesta área que vamos inserir todas as tags meta,
    title, script e style -->
    <meta charset="utf-8">
    <title>Título do seu site</title>
</head>
<body>
    <!-- ##### Cabeçalho: ##### -->
    <header><h1>Nosso site de exemplo</h1></header>
    <!-- ##### Nossa navegação (menu) ##### -->
    <nav>
        <ul>
            <li><a href="#inicial">Inicial</a></li>
            <li><a href="#sobre">Sobre</a></li>
            <li><a href="#contato">Contato</a></li>
        </ul>
    </nav>
    <!-- ##### Área para artigos ##### -->
    <section>
        <article><p>Um pouco de texto</p></article>
        <article><p>Mais um pouco de texto</p></article>
    </section>
    <!-- ##### Rodapé ##### -->
    <footer>
        <p>Informações de rodapé.</p>
    </footer>
</body>
</html>
```

Estilizando com CSS

Para estilizar os novos elemento na sua folha de estilos CSS é mais fácil ainda. Basta digita o nome do elemento sem nada procedendo ele, como se fosse um elemento normal mesmo.

Por exemplo:

```
body {background:#FFF; color:#000}
header h1 {font-size:42px;}
nav {background:#AAA; width:100%}
nav ul li a {color:#666; text-decoration:underline}
nav ul li a:hover {text-decoration:none;}
```

E assim por diante.

Você também pode adicionar uma classe ou ID aos elementos para facilitar na hora de aplicar estilos diferentes ou manipular via JavaScript:

```
<article class="um_estilo estilo_global" id="primeiro">
</article>

<article class="outro_estilo estilo_global" id="segundo">
</article>
```

O seu CSS pode ficar assim:

```
article .um_estilo {color:#AAA;}
article .outro_estilo {color:#BBB;}
article .estilo_global {background:#FFF}
article #primeiro {float:left}
article #segundo {float:right;}
```

Compatibilidade

Todo web designer sabe dos dois tipos de desenvolvimento que ele precisa fazer: a programação que vai funcionar nos navegadores, e a programação que vai funcionar no Internet Explorer.

Felizmente, a versão 9 do navegador parece ser bem mais compatível com o HTML5. Mas já te dou um motivo para não ficar excitado demais: a maioria das pessoas ainda usam versões desatualizadas do navegador.

Ou seja, sites em HTML5 podem parecer cheios de problemas para usuários do IE7 e IE8 (eu me recuso a aceitar que ainda existam usuários de IE6). Isto deveria ser um sério problema, mas por sorte alguém conseguiu resolver.

O código abaixo, que deve ser inserido entre o <head> e </head> é um script hospedado no Google Code. Ele vai fazer com que o site fique como deveria ficar quando o usuário está usando a versão 7 ou 8 do pseudo-navegador da Microsoft, fazendo com que você não precise se preocupar se o seu site funciona no navegador deles.

```
<!--[if IE]>
<script src="http://html5shiv.googlecode.com/svn/trunk/
html5.js">
</script>
<![endif]-->
```

Finalizando

O HTML5 tem ainda outros elementos interessantes, como a fácil implementação de vídeo e áudio (com <video> e <audio> respectivamente), além de outros atributos novos. Mas não é o objetivo deste eBook explicar todos eles.

O que você aprendeu aqui foi o básico para poder fazer a transição do HTML4 ou xHTML1 para o HTML5, além de entender mais sobre como ele funciona e como ele é muito mais limpo que as versões anteriores.

Existem centenas de outros sites que se aprofundam mais no HTML5, então confio que se a sua curiosidade está desperta, você vai buscar por eles.

Espero que este eBook tenha sido instrutivo e te inspire a querer aprender mais sobre o que o HTML5 pode fazer por você.

Se você tiver algum comentário, observação ou crítica, você pode entrar em contato pelo email **canha@design.blog.br**.

Caso queira ficar de olho em futuros eBooks lançados no Design Blog, assine o feed em design.blog.br ou siga nosso perfil no Twitter em twitter.com/design_blog.

Aproveite para compartilhar este eBook com quem você quiser. Só peço duas coisas: 1) Não cobre ou venda este eBook; 2) Não altere os créditos originais.

Muito obrigado por ler!