

Tecnologias para Internet I

Aula 5: Tabelas, Áudio e Vídeo no HTML 5

INTRODUÇÃO



Nesta nova versão do HTML, temos uma determinação de não usar comandos (tags) que tornem rígidas as páginas de web, isto é, temos que sempre levar em conta que ela aparecerá em qualquer tipo de mídia e de qualquer tamanho. Com isso, o uso de comandos como table foram destinados a usos específicos, deixando as páginas mais simples e dinâmicas.

Em contrapartida, cada vez mais o uso de vídeo e som nas páginas se tornam características marcantes e o novo código destina comandos específicos para o uso de formatos mais simples e com menos tamanho, facilitando seu uso.

OBJETIVOS



Identificar as tabelas do HTML5;

Usar vídeo e áudio no HTML5.

A TABELA NO HTML5



Fonte da Imagem: Happy Art / Shutterstock

Até o aparecimento do HTML5, as tabelas eram utilizadas como recurso para diagramação de layouts para páginas. Os designers usavam tabelas como uma grade para exibir imagens e textos, e foram amplamente utilizadas, tornando-se a forma predominante de projeto de sites, criando assim um visual rico.

Entretanto, hoje, o uso de tabelas está, de fato, interferindo na construção de sites melhores, mais acessíveis, flexíveis e funcionais, principalmente para as novas mídias, como o telefone.

Essas tabelas ainda continuam existindo no HTML 5 como recurso para exibição de dados tabulares, informações vindas de base de dados. Com a nova versão do HTML, podemos criar sites feitos completamente sem tabelas (*tableless*), criando assim layouts dinâmicos que aparecem em telas grandes como no computador e se adaptam bem para uma tela pequena do smartphone.

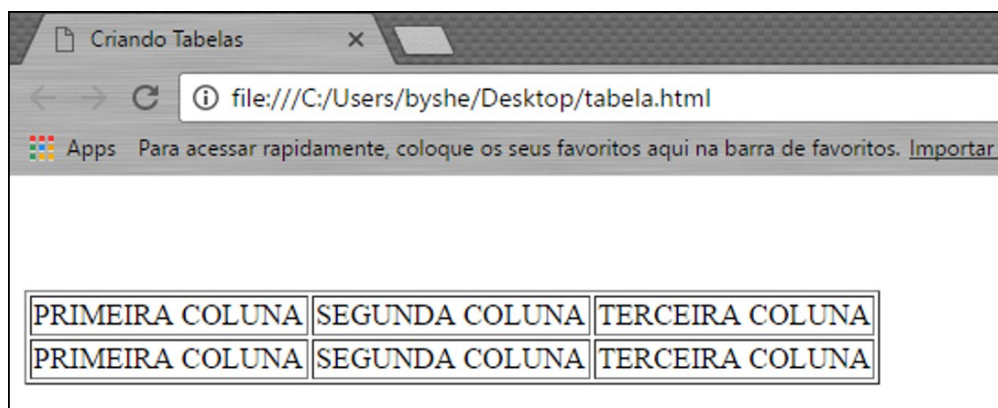
O comando tabela não foi retirado da linguagem, mas ficou restrito o seu uso. Em contrapartida, ainda temos vários sites que utilizam esse recurso, portanto, é importante entendermos como funciona até para poder, se for necessário, dar manutenção nesse tipo de página.

O comando para se inserir uma tabela é **<TABLE>**. Para iniciar uma linha, devemos introduzir a tag **<TR>** e para uma célula **<TD>**. Todos esses comandos são encerrados como **</TABLE>**, **</TR>** e **</TD>** respectivamente.



Veja o código da imagem [aqui \(glossário\)](#).

Neste exemplo foram criadas duas linhas com 3 colunas cada uma, como vemos a seguir.



PRIMEIRA COLUNA	SEGUNDA COLUNA	TERCEIRA COLUNA
PRIMEIRA COLUNA	SEGUNDA COLUNA	TERCEIRA COLUNA

O comando TABLE tem uma série de atributos que ajudam na formatação da tabela. Os principais são:

align

Alinha a tabela nas 3 posições – esquerda, centro e direita.

bgcolor

Determina a cor da tabela.

border

Determina o tamanho da borda, se for zero não mostra a borda.

cellpadding

Determina o espaçamento entre o texto e a borda da célula.

cellspacing

Determina o espaçamento entre células.

Além dos atributos básicos de todo comando de objeto:

Height

Altura da tabela.

Width

Largura da tabela.

Title

Texto que aparece quando passamos o mouse na tabela.

Comandos internos da tabela:**<TR>...</TR>**

Cria uma linha na tabela.

<TD>...</TD>

Divide a linha da tabela.

Exemplo:

```
<tr>
<td> aqui vai o texto ou imagem ou qualquer objeto da primeira divisão</td>
<td> aqui vai o texto ou imagem ou qualquer objeto da segunda divisão </td>
<td> aqui vai o texto ou imagem ou qualquer objeto da terceira divisão </td>
</tr>
```

Este código irá gerar uma linha com três colunas.

<TH>...</TH>

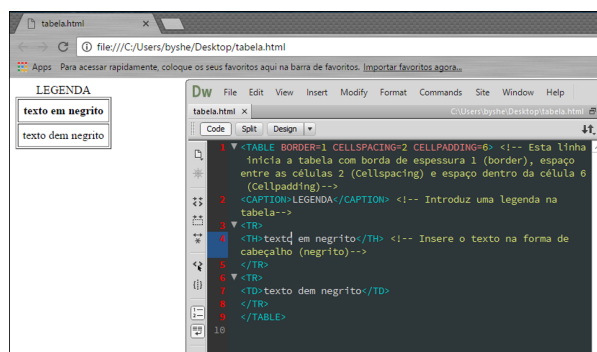
Divide a linha da tabela colocando o conteúdo em negrito.

<CAPTION>...</CAPTION>

Exibe um texto centralizado em relação à tabela, como se fosse uma legenda.

Exemplo:

```
<TABLE BORDER=1 CELLSPACING=2 CELLPADDING=6> <!-- Esta linha inicia a tabela com borda
de espessura 1 (border)
, espaço entre as células 2 (Cellspacing) e espaço dentro da célula 6 (Cellpadding)-->
<CAPTION>LEGENDA</CAPTION> <!-- Introduz uma legenda na tabela-->
<TR>
<TH>Conteúdo em negrito</TH> <!-- Insere o texto na forma de cabeçalho (negrito)-->
</TR>
<TR>
<TD>Conteúdo em negrito</TD>
</TR>
</TABLE>
```



O ATRIBUTO RULES

Para escolher as linhas internas que serão mostradas dentro da tabela, usamos o atributo RULES, dentro da tag . Os complementos desse atributo são:

none

Nenhuma linha interna.

rows

Para as linhas horizontais entre cada linha da tabela.

cols

Para as linhas verticais entre cada coluna da tabela.

all

Para mostrar todas as linhas entre cada coluna e linha na tabela (default).

groups

Para linhas entre grupos de colunas e seções horizontais, definidas por tags especiais como COLGROUP e THEAD.

Veja alguns exemplos de tabelas:

<TABLE BORDER RULES=all>

Nome	Idade	Área
José Maria	35	Gerencia
Maria José	28	Recursos Humanos
Marcos Aurélio	24	C.P.D.

<TABLE BORDER RULES=none>

Nome	Idade	Área
José Maria	35	Gerencia
Maria José	28	Recursos Humanos
Marcos Aurélio	24	C.P.D.

<TABLE BORDER RULES=rows>

Nome	Idade	Área
José Maria	35	Gerencia
Maria José	28	Recursos Humanos
Marcos Aurélio	24	C.P.D.

<TABLE BORDER RULES=cols>

Nome	Idade	Área
José Maria	35	Gerencia
Maria José	28	Recursos Humanos
Marcos Aurélio	24	C.P.D.

, O USO DO VÍDEO NA INTERNET

O Flash sempre foi a alternativa mais viável para quem precisava adicionar vídeos e músicas nas páginas Web. Ele estava e ainda está presente em quase todos os navegadores e tínhamos um formato que podia ser usado sem maiores problemas.

Bastava adicionar uma chamada para o plugin e funcionava.

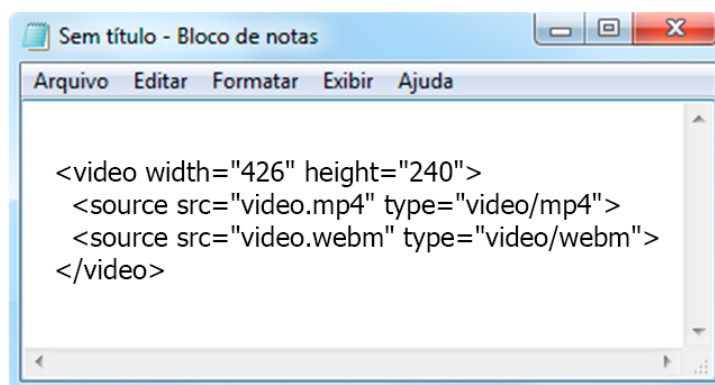
Em virtude das inconsistências dos navegadores, era preciso adicionar tanto o comando **<object>** quanto o comando **<embed>**, comandos específicos para utilização do plugin, duplicando muitos atributos. Isso sem falar que plugins como o Flash causam muitas instabilidades, ocasionando erros que levam até ao fechamento do navegador.

Devido a tudo isso, o HTML 5 vem com um novo suporte, nativo na linguagem, para evitar o uso de plugins e comandos desnecessários.

Atualmente, basta colocar a chamada simples, como se fosse uma imagem e o arquivo é acionado.

O problema apresentado pelo novo comando é o uso do atributo **src** que não funciona bem em alguns navegadores. E se pretendemos que todos possam abrir a página, com qualquer navegador, é necessário usar outro tipo de atributo. Por isso, é comum o uso do atributo **<source>** na tag **<video>**.

O elemento **<source>** pode ser usado múltiplas vezes e permite definir um formato de vídeo para cada navegador, fazendo com que você tenha um alcance maior.



Existem diversos **codecs** ([glossário](#)), possíveis e você precisará exportar pelo menos duas versões para ter um alcance maior.

A seguir, encontramos os formatos aceitos em cada navegador sem a necessidade de ter que instalar **plugins/codecs**:



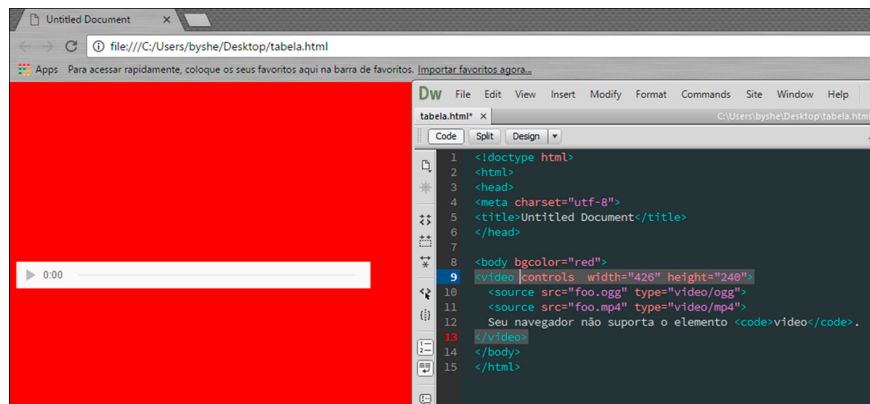
Analisando os formatos aceitos pelos navegadores, percebemos que precisamos apenas de dois formatos para ter um alcance maior: **.mp4 (glossário)** e **.webm. (glossário)** O formato **Ogg (.ogv) (glossário)** tem uma relação entre qualidade vídeo e tamanho do arquivo que não é tão boa quanto o formato WebM (a própria Mozilla recomenda o uso de WebM).

RESUMINDO...

A principal diferença entre os navegadores é, provavelmente, o suporte a diversos formatos de vídeo. O vídeo HTML5 é um padrão emergente e não está vinculado a nenhum formato de vídeo. Observe na tabela as configurações possíveis:

Arquivo de mídia	Config. de extensão	Config. de tipo MIME
Áudio mp3	mp3	audio/mpeg
Áudio mp4	m4a	audio/mp4
Áudio webm	webm	audio/webm
Vídeo mp4 (H.264)	mp4	video/mp4
Vídeo webm	webm	video/webm

Veja um exemplo de código:



Glossário

CÓDIGO

```
<HTML>
<HEAD>
<TITLE>Criando Tabelas</TITLE>
</HEAD>
<BODY>
<CENTER><FONT COLOR=BLUE SIZE=6>TABELA</FONT></CENTER>
<BR>
<TABLE BORDER=1>
<TR> <!-- Cria a primeira linha da tabela-->
<TD>PRIMEIRA COLUNA </TD> <!-- Aqui foi criada uma célula-->
<TD>SEGUNDA COLUNA </TD>
<TD>TERCEIRA COLUNA </TD>
</TR> <!-- Fecha a primeira linha da tabela-->
<TR> <!-- Abre a segunda linha da tabela-->
<TD> PRIMEIRA COLUNA
<TD>SEGUNDA COLUNA </TD>
<TD>TERCEIRA COLUNA </TD>
</TR> <!-- Encerra a Segunda linha da tabela>
</TABLE> <!-- Encerra a tabela-->
</BODY>
</HTML>
```

EXEMPLO

```
<object width="426" height="240">
<param name="movie" value=" https://youtu.be/S2Vty9srpFw"></param>
<param name="allowFullScreen" value="true"></param>
<param name="allowsriptaccess" value="always"></param>
<embed src= https://www.youtube.com/embed/S2Vty9srpFw
type="application/x-shockwave-flash"
allowsriptaccess="always"
allowfullscreen="true"
width="426"
height="240">
</embed>
</object>
```

EXEMPLO

```
<video controls src="video.mp4" width="426" height="240"></video>
```

CODECS

Programas utilizados para codificar e decodificar arquivos de mídia. Eles compactam o formato original, favorecendo o armazenamento, e descompactam na hora da reprodução, transformando novamente em imagem ou áudio.

MP4 H.264

O container (aplicativo utilizado para apresentação de vídeo) **MP4** utiliza o codec (plugin de áudio e vídeo) de vídeo **H.264** e pode usar o **MP3** ou **AAC** como codecs de áudio.

Sua relação compressão–qualidade é muito superior aos outros formatos como WebM e Ogg.

Ele pode ser executado sem a necessidade de instalações adicionais no Internet Explorer, Microsoft Edge, Safari e Chrome (todos os navegadores em suas últimas versões).

Os formatos baseados no MPEG são cobertos por patentes e não podem ser distribuídos livremente, pois exigem o pagamento da licença de uso. O **mime type** (do inglês *Multipurpose Internet Mail Extensions* é uma norma da internet para o formato das mensagens de correio eletrônico), utilizado para o formato MP4 é video/mp4.

WEBM

O formato WebM é baseado em uma versão mais restrita do container Matroska (.mkv). Ele usa o codec de vídeo VP8 e o codec de áudio Vorbis. Ele pode ser executado sem a necessidade de instalações adicionais no Firefox, Chrome e Opera.

O **mime type** utilizado para o formato WebM é video/webm.

OGG THEORA VORBIS

O container Ogg, que utiliza o codec de vídeo Theora e o codec de áudio Vorbis, pode ser executado sem a necessidade de instalações adicionais no Firefox, Chrome e Opera.

O mime type utilizado para o formato Ogg é video/ogg.