



AULA 4: Introdução a Páginas Estáticas

Tabelas, Áudio, Vídeo, Formulários e Noções de Estilo



Noções de Estilo – Uso de CSS inline

Até o momento, tratamos de organizar o conteúdo de nossas páginas HTML utilizando as tags corretas para otimizar a manutenção de nossa página.

Daremos início ao uso do CSS para trazer alguma noção de estilo ao nosso trabalho. O CSS assume a formatação, estilo e design das páginas web, permitindo uma codificação mais leve das páginas, favorecendo a indexação (SEO*) e permitindo layouts como páginas de revistas.

A regra CSS é a unidade básica de uma folha de estilo. Nessa definição, o termo unidade básica significa a menor porção de código capaz de produzir efeito de estilização.

Uma regra CSS é composta de duas partes: o seletor e a declaração.

Inicialmente, iremos trabalhar com CSS inline, ou seja, o CSS aplicado na tag HTML. Esse modelo é aplicado utilizando o texto `'style="seletor:valor;'`

Exemplo: `<p style="color: white ; background:black;"> teste </p>`

*Voltaremos a abordar esse tópico na próxima aula



Tabelas

São elementos que representam dados de uma ou mais dimensões. Podem ser utilizadas para exibição de tabelas de preços de um site.

<TABLE>: Elemento básico para criação da tabela.

<CAPTION>: Elemento que indica o título da tabela.

<THEAD>: Neste elemento, encontram-se as linhas que formam o título de cada coluna da tabela.

<TBODY>: Neste elemento, encontram-se os dados de cada coluna.

<TFOOT>: Neste elemento, encontram-se as informações finais de cada coluna.

<TR>: Elemento que cria uma linha na tabela.

<TD>: Elemento que cria uma coluna na tabela.

<TH>: Este elemento é idêntico ao <TD>, só é utilizado dentro da TAG de título.



Tabelas

Exemplo de utilização:

```
<!DOCTYPE html>
<html>
  <header>
    <title>Exemplo de tabela</title>
  </header>
  <body>
    <table>
      <caption>Lista de preços</caption>
      <thead>
        <tr>
          <th>Serviço</th>
          <th>Tempo</th>
          <th>Valor</th>
        </tr>
      </thead>
      <tbody>
        <tr>
          <th>Corte de Cabelo</th>
          <td>10 Min</td>
          <td>R$ 5,00</td>
        </tr>
```

```
      <tr>
        <th>Manicure</th>
        <td>30 Min</td>
        <td>R$ 20,00</td>
      </tr>
      <tr>
        <th>Pedicure</th>
        <td>30 Min</td>
        <td>R$ 10,00</td>
      </tr>
    </tbody>
  </table>
</body>
</html>
```



Vídeo e Áudio

Anteriormente, para a exibição de vídeos, em um documento HTML, os desenvolvedores utilizavam a TAG `<object>` que é um container genérico para incorporar objetos estrangeiros (plugins de terceiros) à página. Devido a inconsistência dos navegadores, era necessário replicar várias linhas de código para que o vídeo pudesse ser exibido.

O HTML 5 provê a TAG `<video>` para eliminar os problemas de incompatibilidade entre os navegadores.

Pesquisa em Aula:

- **O que é necessário para executar um vídeo ou um áudio?**
- **Como identificamos os diferentes tipos? É necessário um elemento externo?**



CODEC's

“Os elementos <audio> e <video> fornecem suporte para a reprodução de mídias de áudio e vídeo sem necessitar de plug-ins.

Codecs de áudio e vídeo são usados para manipular arquivos de áudio e vídeo, diferentes codecs oferecem diferentes níveis de compressão e qualidade.

Um formato do repositório é usado para armazenar e transmitir o codec de áudio e vídeo (ambos juntos, no caso de um vídeo com tilha sonora).

Existem muitas combinações de codecs e formatos de containers, embora apenas alguns são relevantes para a internet.”

Fonte: https://developer.mozilla.org/pt-BR/docs/Web/HTML/formatos_midia_suportados

Orientações sobre codecs suportados no HTML5 e compatibilidade com os navegadores:

- Vídeo: https://www.w3schools.com/Tags/tag_video.asp
- Áudio: https://www.w3schools.com/Tags/tag_audio.asp



Formatos de Vídeo (https://www.w3schools.com/html/html_media.asp)

Format	File	Description
MPEG	.mpg .mpeg	MPEG. Developed by the Moving Pictures Expert Group. The first popular video format on the web. Used to be supported by all browsers, but it is not supported in HTML5 (See MP4).
AVI	.avi	AVI (Audio Video Interleave). Developed by Microsoft. Commonly used in video cameras and TV hardware. Plays well on Windows computers, but not in web browsers.
WMV	.wmv	WMV (Windows Media Video). Developed by Microsoft. Commonly used in video cameras and TV hardware. Plays well on Windows computers, but not in web browsers.
QuickTime	.mov	QuickTime. Developed by Apple. Commonly used in video cameras and TV hardware. Plays well on Apple computers, but not in web browsers. (See MP4)
RealVideo	.rm .ram	RealVideo. Developed by Real Media to allow video streaming with low bandwidths. It is still used for online video and Internet TV, but does not play in web browsers.
Flash	.swf .flv	Flash. Developed by Macromedia. Often requires an extra component (plug-in) to play in web browsers.
Ogg	.ogg	Theora Ogg. Developed by the Xiph.Org Foundation. Supported by HTML5.
WebM	.webm	WebM. Developed by the web giants, Mozilla, Opera, Adobe, and Google. Supported by HTML5.
MPEG-4 or MP4	.mp4	MP4. Developed by the Moving Pictures Expert Group. Based on QuickTime. Commonly used in newer video cameras and TV hardware. Supported by all HTML5 browsers. Recommended by YouTube.



Formatos de Áudio (https://www.w3schools.com/html/html_media.asp)

Format	File	Description
MIDI	.mid .midi	MIDI (Musical Instrument Digital Interface). Main format for all electronic music devices like synthesizers and PC sound cards. MIDI files do not contain sound, but digital notes that can be played by electronics. Plays well on all computers and music hardware, but not in web browsers.
RealAudio	.rm .ram	RealAudio. Developed by Real Media to allow streaming of audio with low bandwidths. Does not play in web browsers.
WMA	.wma	WMA (Windows Media Audio). Developed by Microsoft. Commonly used in music players. Plays well on Windows computers, but not in web browsers.
AAC	.aac	AAC (Advanced Audio Coding). Developed by Apple as the default format for iTunes. Plays well on Apple computers, but not in web browsers.
WAV	.wav	WAV. Developed by IBM and Microsoft. Plays well on Windows, Macintosh, and Linux operating systems. Supported by HTML5.
Ogg	.ogg	Ogg. Developed by the Xiph.Org Foundation. Supported by HTML5.
MP3	.mp3	MP3 files are actually the sound part of MPEG files. MP3 is the most popular format for music players. Combines good compression (small files) with high quality. Supported by all browsers.
MP4	.mp4	MP4 is a video format, but can also be used for audio. MP4 video is the upcoming video format on the internet. This leads to automatic support for MP4 audio by all browsers.



Tag <Video>

Elemento que exibe vídeos ou filmes.

Atributos:

autoplay = "autoplay" ou "" (string vazia) ou sem atributo. Todas vez que a página for recarregada, o vídeo se iniciará.

controls = "controls" ou "" (string vazia) ou sem atributo. Será exibido a interface de controle de vídeo.

loop = "loop" ou "" (string vazia) ou sem atributo. O vídeo será passado continuamente.

poster = Endereço de uma imagem será exibida enquanto o vídeo não carrega.

height = Altura do vídeo em pixels

width = Largura do vídeo em pixels

muted = "muted" or ""ou "" (string vazia) ou sem atributo. Estado do vídeo default com o áudio desligado.

src = URL do vídeo

```
<video controls src="http://media.w3.org/2010/05/bunny/movie.ogv">
```

Seu navegador não suporta vídeos em HTML 5

```
</video>
```



Tag <Audio>

Elemento que permite tocar um arquivo de áudio. Possui atributos semelhantes a tag <video>.

Atributos:

autoplay - Todas vez que a página for recarregada, o áudio se iniciará.

controls - Será exibido a interface de controle de áudio.

loop - O áudio será reproduzido continuamente.

muted - Define que o áudio não deve ser reproduzido automaticamente.

preload - Apresenta os valores: "auto", "metadata" e "none". Especifica se e como áudio deve ser reproduzido quando a página é carregada.

src - URL do áudio.

<audio controls

src="http://media.w3.org/2010/07/bunny/04-Death_Becomes_Fur.oga">

Seu navegador não suporta áudio em HTML 5

</audio>



Exercícios de Fixação

1) Monte uma página que apresente uma tabela sobre estilos de música e alguns representantes dos estilos selecionados.

Em pelo menos um representante, deve haver um vídeo (Vídeo do YouTube – orientações de como usar em https://www.w3schools.com/html/html_youtube.asp)

2) Usando uma das páginas construídas até o momento, insira alguns elementos de estilo (borda, formatação, cor)

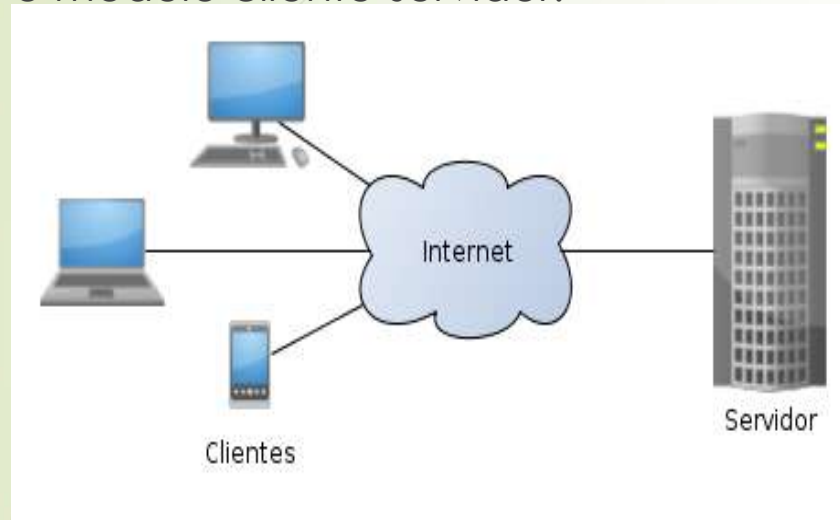
Orientações:

- https://www.w3schools.com/html/html_tables.asp
- https://www.w3schools.com/html/html_styles.asp



Formulários

Como já dito anteriormente, o modelo de aplicação mais comum na Internet é o modelo cliente-servidor.



Fonte: <https://pt.wikipedia.org/wiki/Cliente-servidor>

Tag <form>

Formulários HTML são usados para transmitir dados para um servidor. Com eles é possível criar elementos de entrada, como campos de texto, caixas de seleção, botões e listas de seleção, áreas de texto, legendas, e elementos de rotulagem.

Atributos:

action = url onde será enviada as informações.

method = método de envio das informações GET ou POST

(O que significam esses métodos de envio?)

```
<form action="http://www.google.com/search" method="get">  
<label>Google: <input type="search" name="q"></label>  
<input type="submit" value="Search...">  
</form>
```



Formulários

Tag <fieldset>

Elemento que permite um agrupamento de informações geralmente utilizado como seção dentro de um formulário.

```
<fieldset name="userinfo">  
  <legend>Informações do usuário</legend>  
  <label for="nome">Nome</label>  
  <input type="text" name="nome" size="40">  
  <label for="address">Endereço</label>  
  <input type="text" name="endereco" size="40">  
  <label for="phone">Phone</label>  
  <input type="text" name="phone" size="40">  
</fieldset>
```




Formulários

Tag <legend>

Elemento que indica o título da TAG <fieldset>.

```
<fieldset name="userinfo">  
  <legend>User information</legend>  
  <label for="name">Name</label><input type="text" name="name" size="40">  
  <label for="address">Address</label><input type="text" name="address" size="40">  
  <label for="telefone">Telefone</label><input type="tel" name="telefone" size="40">  
</fieldset>
```

Tag <label>

Elemento que representa o título de uma TAG de entrada. O atributo for= Indica qual elemento está relacionado.

```
<input type="checkbox" name="carro"> <label for="carro">Carro</label>  
<input type="checkbox" name="moto"> <label for="carro">Moto</label>  
<input type="checkbox" name="bicicleta"> <label for="carro">Bicicleta</label>
```



Formulários

Tag <input>:

O elemento <input> é usado para a criação de campos do formulário, em função do tipo (atributo type) pode ser de tipo campo de texto, caixa de seleção, botão de opção de senha, botão enviar, dentre outros.

Criando campos de texto:

Nome: <input type="text" name="nome">

Sobrenome: <input type="text" name="sobrenome">

Criando botões de seleção (radio):

<form>

Selecione o sexo:

<input type="radio" name="sex" value="masc">Masculino

<input type="radio" name="sex" value="fem">Feminino

</form>

Criando campo de senha (password):

Digite sua senha abaixo com 6 dígitos:

<input type="password" maxlength="6" name="senha">



Formulários

Tag `<input>`:

Criando caixas de seleção (checkbox):

Marque os esportes você pratica:

`<input type="checkbox" name="esporte" value="volei" checked>Volei
`

`<input type="checkbox" name="esporte" value="futebol"> Futebol
`

`<input type="checkbox" name="esporte" value="basquete">Basquete
`

`<input type="checkbox" name="esporte" value="outros" disabled>Outros
`

Criando campo de envio múltiplo(file):

`<input type="file" name="arquivo" multiple>Selecione o arquivo:`

Criando campo de data (date):

Nascimento: `<input type="date" name="data">`

Criando campo de e-mail (email):

E-mail: `<input type="email" name="email" placeholder="e-mail">`

Criando campo numérico com intervalo (number):

Valor entre 5 e 10: `<input type="number" name="quantity" min="5" max="10">`



Formulários

Tag <button>:

Criando botões:

```
<button type="submit" value="submit">submit</button>  
<button type="reset" value="reset">reset</button>
```

Tag <select>:

Criando caixa de seleção:

```
<select name="profissao">  
  <option value="advogado">advogado</option>  
  <option value="medico">médico</option>  
  <option value="professor">Professor</option>  
  <option value="coveiro">Coveiro</option>  
</select>
```



Formulários

Tag <optgroup>:

Criando caixa de seleção com grupos <select>:

```
<select>
  <optgroup label="Humanas">
    <option value="direito">Direito</option>
    <option value="filisofia">Filosofia</option>
  </optgroup>
  <optgroup label="Exatas">
    <option value="mat">Matemática</option>
    <option value="est">Estatística</option>
  </optgroup>
</select>
```

Tag <datalist>:

Criando caixa de seleção usando <datalist>:

```
<input list="carros" name="carros">
  <datalist id="carros">
    <option value="Ferrari">
    <option value="Porche">
    <option value="BMW">
    <option value="Jeep">
    <option value="Outros">
  </datalist>
```

Tag <textarea>:

Criando caixa de texto:

```
<textarea rows="5" cols="25"> Comentários / Sugestões</textarea>
```




Questionamentos e Pesquisas

Nestas aulas vimos diversas tags HTML que permitem a identificação de seções, blocos, campos e áreas dos objetos que compõem a página HTML.

1. Por que isso é importante?
2. Como podemos utilizar essas características?

Tente responder essas questões....

Seguem algumas dicas para enriquecer sua pesquisa:

- Definição de linguagens scripts:
<http://www.tecmundo.com.br/programacao/1185-o-que-e-script-.htm>
- Uso das linguagens script:
<http://www.infoq.com/br/news/2009/04/top-scripting-languages-php-ruby>

Em breve utilizaremos algumas delas...

