



Aula 3: HTML 5

Prof. Dr. Rafael Moreno

HTML 5

- Um dos **principais objetivos do HTML5** é **facilitar a manipulação do elemento** possibilitando o desenvolvedor **a modificar as características dos objetos** de forma não intrusiva e de maneira que seja transparente para o usuário final.
- Ao contrário das versões anteriores, o **HTML5 fornece ferramentas para a CSS e o Javascript fazerem seu trabalho da melhor maneira possível**. O HTML5 permite por meio de suas APIs a manipulação das características destes elementos, de forma que o website ou a aplicação continue leve e funcional.

HTML 5

- O HTML5 também cria novas tags e modifica a função de outras
 - As versões antigas do HTML não continham um padrão universal para a criação de seções comuns e específicas **como rodapé, cabeçalho, sidebar, menus e etc.** Não havia um padrão de nomenclatura de IDs, Classes ou tags. Não havia um método de capturar de maneira automática as informações localizadas nos rodapés dos websites.

HTML 5

- O HTML5 **modifica a forma de como escrevemos código e organizamos a informação na página.**
- Mais semântica com menos código.
- Mais interatividade sem a necessidade de instalação de plugins e perda de performance.

Estrutura básica

```
<!DOCTYPE HTML>  
<html lang="pt-br">  
<head>  
<meta charset="UTF-8">  
<title></title>  
</head>  
<body>  
</body>  
</html>
```

- Doctype
 - O Doctype indica para o navegador e para outros meios qual a especificação de código utilizar.
 - O Doctype não é uma tag do HTML, mas uma instrução para que o browser tenha informações sobre qual versão de código a marcação foi escrita.
- Lang
 - O atributo LANG é necessário para que os user-agents saibam qual a linguagem principal do documento

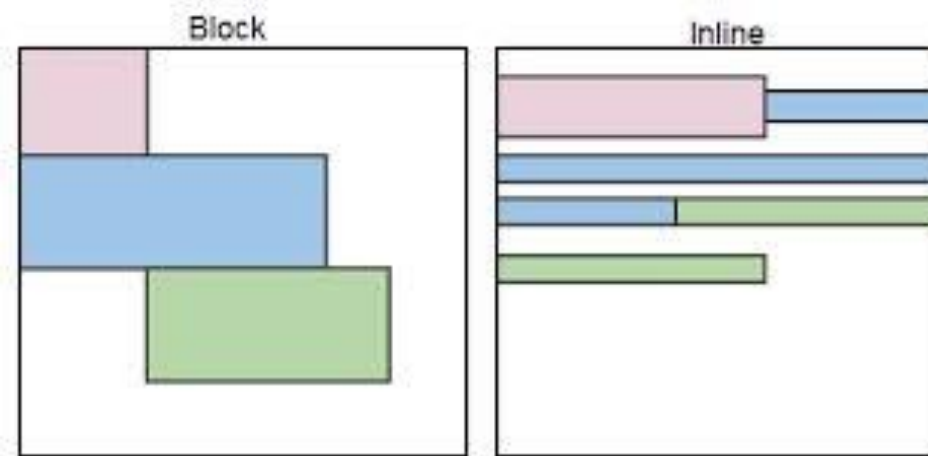
Modelos de conteúdo

- Há pequenas regras básicas estão no HTML desde o início.
- Estas regras definem onde os elementos podem ou não estar.
- Se eles podem ser filhos ou pais de outros elementos e quais os seus comportamentos.
- Dentre todas as categorias de modelos de conteúdo, existem dois tipos de elementos: **elementos de linha e de bloco**.

Modelos de conteúdo

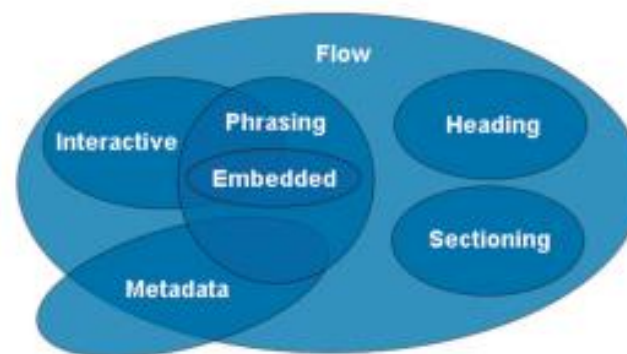
- **Elementos de linha marcam em sua maioria texto**
 - a, strong, em, img, input, abbr, span.
- **Elementos em bloco** são como caixas que dividem o conteúdo em seções do layout.
- Os elementos de linha podem conter outros elementos de linha, dependendo da categoria que ele se encontra. Por exemplo: o elemento **a** não pode conter o elemento **label**.
- Os elementos de linha nunca podem conter elementos de bloco.
- Elementos de bloco sempre podem conter elementos de linha.
- Elementos de bloco podem conter elementos de bloco, dependendo da categoria que ele se encontra. Por exemplo, um parágrafo não pode conter um DIV. Mas o contrário é possível.

Modelos de conteúdo



Modelos de conteúdo

- Estes dois grandes grupos podem ser divididos em categorias.
- **Estas categorias dizem qual modelo de conteúdo o elemento trabalha e como pode ser seu comportamento:**
 - Metadata content
 - Flow content
 - Sectioning content
 - Heading content
 - Phrasing content
 - Embedded content
 - Interactive content



Metadata content

- Este conteúdo **vem antes da apresentação**, formando uma relação com o documento e seu conteúdo com outros documentos que distribuem informação por outros meios.
- Os elementos que compõe a categoria Metadata são:
 - base
 - command
 - link
 - meta
 - noscript
 - script
 - style
 - title

Flow content

- **Devem ter pelo menos um descendente de texto ou um elemento descendente que faça parte da categoria embedded.**
- **A maioria dos elementos utilizados no body e aplicações são categorizados como Flow Content.**
 - a • abbr • address • area (se for um decendente de um elemento de mapa) • article • aside • audio • b • bdo • blockquote • br • button • canvas • cite • code • command • datalist • del • details • dfn • div • dl • em • embed • fieldset • figure • footer • form • h1 • h2 • h3 • h4 • h5 • h6 • header • hgroup • hr • i • iframe • img • input • ins • kbd • keygen • label • link (Se o atributo itemprop for utilizado) • map • mark • math • menu • meta (Se o atributo itemprop for utilizado) • meter • nav • noscript • object • ol • output • p • pre • progress • q • ruby • samp • script • section • select • small • span • strong • style (Se o atributo scoped for utilizado) • sub • sup • svg • table • textarea • time • ul • var • video • wbr • Text

Sectioning content

- Basicamente são elementos que juntam grupos de textos no documento.
- Estes elementos definem um grupo de cabeçalhos e rodapés.
 - article
 - aside
 - nav
 - section

Heading content

- Os elementos da categoria Heading definem uma seção de cabeçalhos, que podem estar contidos em um elemento na categoria Sectioning.
- h1
- h2
- h3
- h4
- h5
- h6
- hgroup

Phrasing content

- Fazem parte desta categoria elementos que **marcam o texto do documento**, bem como os elementos que **marcam este texto dentro do elemento de parágrafo**.
 - a • abbr • area (se ele for descendente de um elemento de mapa) • audio • b • bdo • br • button • canvas • cite • code • command • datalist • del (se ele contiver um elemento da categoria de Phrasing) • dfn • em • embed • i • iframe • img • input • ins (se ele contiver um elemento da categoria de Phrasing) • kbd • keygen • label • link (se o atributo itemprop for utilizado) • map (se apenas ele contiver um elemento da categoria de Phrasing) • mark • math • meta (se o atributo itemprop for utilizado) • meter • noscript • object • output • progress • q • ruby • samp • script • select • small • span • strong • sub • sup • svg • textarea • time • var • video • wbr • Text

Embedded content

- Na categoria Embedded, há elementos que importam outra fonte de informação para o documento.
 - audio
 - canvas
 - embed
 - iframe
 - img
 - math
 - object
 - svg
 - video

Interactive content

- Interactive Content são elementos que fazem parte da interação de usuário.
 - a
 - audio (se o atributo control for utilizado)
 - button
 - details
 - embed
 - iframe
 - 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)

Novos elementos e atributos

- O HTML5 trouxe uma série de elementos que nos ajudam a **definir setores principais no documento HTML**.
- Com a ajuda destes elementos, podemos por exemplo **diferenciar diretamente pelo código HTML5 áreas importantes do site** como **sidebar, rodapé e cabeçalho**. Conseguimos seccionar a área de conteúdo indicando onde exatamente é o texto do artigo.
- Estas mudanças **simplificam o trabalho de sistemas como os dos buscadores**. Com o HTML5 os buscadores conseguem vasculhar o código de maneira mais eficaz. Procurando e guardando informações mais exatas e levando menos tempo para estocar essa informação.



Novos elementos e atributos

Elemento	Descrição
section	Define uma nova seção genérica no documento. Por exemplo, a home de um website pode ser dividida em diversas seções: introdução ou destaque, novidades, informação de contato e chamadas para conteúdo interno.
nav	Representa uma seção da página que contém links para outras partes do website. Nem todos os grupos de links devem ser elementos nav, apenas aqueles grupos que contém links importantes. Isso pode ser aplicado naqueles blocos de links que geralmente são colocados no Rodapé e também para compor o menu principal do site.
article	Representa uma parte da página que poderá ser distribuído e reutilizável em FEEDs por exemplo. Isto pode ser um post, artigo, um bloco de comentários de usuários ou apenas um bloco de texto comum.

Novos elementos e atributos

Elemento	Descrição
aside	O aside pode ser representado por conteúdos em sidebars em textos impressos, publicidade ou até mesmo para criar um grupo de elementos nav e outras informações separados do conteúdo principal do website.
hgroup	Este elemento consiste em um grupo de títulos. Ele serve para agrupar elementos de título de H1 até H6 quando eles tem múltiplos níveis como título com subtítulos e etc.
header	Representa um grupo de introdução ou elementos de navegação. O elemento header pode ser utilizado para agrupar índices de conteúdos, campos de busca ou até mesmo logos.
footer	Representa literalmente o rodapé da página. Seria o último elemento do último elemento antes de fechar a tag HTML. O elemento footer não precisa aparecer necessariamente no final de uma seção.
time	Este elemento serve para marcar parte do texto que exibe um horário ou uma data precisa no calendário gregoriano.



Facens

Atributos descontinuados

- align como atributo da tag caption, iframe, img, input, object, legend, table, hr, div, h1, h2, h3, h4, h5, h6, p, col, colgroup, tbody, td, tfoot, th, thead e tr.
- alink, link, text e vlink como atributos da tag body.
- background como atributo da tag body.
- bgcolor como atributo da tag table, tr, td, th e body.
- border como atributo da tag table e object.
- cellpadding e cellspacing como atributos da tag table.
- char e charoff como atributos da tag col, colgroup, tbody, td, tfoot, th, thead e tr.
- clear como atributo da tag br.
- compact como atributo da tag dl, menu, ol e ul.
- frame como atributo da tag table.
- frameborder como atributo da tag iframe.
- height como atributo da tag td e th.
- hspace e vspace como atributos da tag img e object.
- marginheight e marginwidth como atributos da tag iframe.
- noshade como atributo da tag hr.
- nowrap como atributo da tag td e th.
- rules como atributo da tag table.
- scrolling como atributo da tag iframe.
- size como atributo da tag hr.
- type como atributo da tag li, ol e ul.
- valign como atributo da tag col, colgroup, tbody, td, tfoot, th, thead e tr.
- width como atributo da tag hr, table, td, th, col, colgroup e pre.

Novos tipos de campos

- Tel
- Search
- Email
- url
- Datas e horas
 - datetime • date • month • week • time • datetime-local
- Number
- Range
- Color

Tipos de dados e validadores

- autofocus
 - `<input name="login" autofocus >`
- placeholder
 - `<input name="q" placeholder="Search here">`
- required
 - `<input name="login" required>`
- maxlength

Tipos de dados e validadores

- Validadores

- **pattern**

- O atributo pattern nos permite definir expressões regulares de validação, sem Javascript:

```
<!DOCTYPE html>
```

```
<form>
```

```
<label for="CEP">CEP:
```

```
<input name="CEP" id="CEP" required pattern= "\d{5}-?\d{3}" />
```

```
</label>
```

```
<button type="submit" >Enviar</button>
```

```
</form>
```


Tipos de dados e validadores

- novalidate e formnovalidate
 - Podem haver situações em que você precisa que um formulário não seja validado. Nestes casos, basta incluir no **elemento form o atributo novalidate**.
 - Outra situação comum é querer que o formulário não seja validado dependendo da ação de submit. Nesse caso, você pode usar no **botão de submit o atributo formnovalidate**. Veja um exemplo:

<form>

<label>nome: <input name="nome" required></label>

<label>email: <input name="email" type="email" required></label>

<label>mensagem: <textarea name="mensagem" required></textarea></label>

<input type="submit" name="action" value="Salvar rascunho" formnovalidate>

<input type="submit" name="action" value="Enviar">

</form>

Detalhes e conteúdo editável

- Detalhes e sumário

- Veja um exemplo de uso dos novos elementos details e summary:

```
<details>
```

```
<summary>Copiando <progress max="39248" value="14718"> 37,5%</summary>
```

```
<dl>
```

```
<dt>Tamanho total:</dt>
```

```
<dd>39.248KB</dd>
```

```
<dt>Transferido:</dt>
```

```
<dd>14.718</dd>
```

```
<dt>Taxa de transferência:</dt>
```

```
<dd>127KB/s</dd>
```

```
<dt>Nome do arquivo:</dt>
```

```
<dd>HTML5.mp4</dd>
```

```
</dl>
```

```
</details>
```

Conteúdo editável

- **Para tornar um elemento do HTML editável**, basta incluir nele o atributo contenteditable, assim:

```
<div contenteditable="true">
```

Edite-me...

```
</div>
```

Drag-n-drop

- A API de Drag and Drop é relativamente simples. Basicamente, inserir o atributo **draggable="true"** num elemento o torna arrastável. E há uma série de eventos que você pode tratar. Os eventos do objeto sendo arrastado são:
 - **dragstart** O objeto começou a ser arrastado. O evento que a função recebe tem um atributo target, que contém o objeto sendo arrastado.
 - **drag** O objeto está sendo arrastado
 - **dragend** A ação de arrastar terminou
- O objeto sobre o qual outro é arrastado sofre os seguintes eventos:
 - **dragenter** O objeto sendo arrastado entrou no objeto target
 - **dragleave** O objeto sendo arrastado deixou o objeto target
 - **dragover** O objeto sendo arrastado se move sobre o objeto target
 - **drop** O objeto sendo arrastado foi solto sobre o objeto target

Drag-n-drop Exemplo

```
<!DOCTYPE HTML>
<html>
<head>
<style>
#div1,#div2 {
  width: 100px;
  height: 100px;
  padding: 10px;
  border: 1px solid #aaaaaa;
}
</style>
<script>
function allowDrop(ev) {
  ev.preventDefault();
}
function drag(ev) {
  ev.dataTransfer.setData("text", ev.target.id);
}
```

```
function drop(ev) {
  ev.preventDefault();
  var data = ev.dataTransfer.getData("text");
  ev.target.appendChild(document.getElementById(data));
}
</script>
</head>
<body>
<div id="div1" ondrop="drop(event)" ondragover="allowDrop(event)"></div>
<br>

<div id="div2" ondrop="drop(event)" ondragover="allowDrop(event)"></div>
</body>
</html>
```

Correção Ortográfica

- Os agentes de usuário podem oferecer recursos de revisão ortográfica e gramatical, dependendo do que houver disponível em cada plataforma.
- Os desenvolvedores podem controlar o comportamento dessa ferramenta através do atributo spellcheck. **Inserir spellcheck="true" num elemento faz com que a revisão esteja habilitada para ele.** Você também pode desabilitar a revisão para determinado elemento, inserindo spellcheck="false".

Correção Ortográfica

```
<!DOCTYPE html>
```

```
<html >
```

```
<body>
```

```
<p contenteditable="true" spellcheck="true">This is a praggagraph. It is  
editable. Try to change the text.</p>
```

```
First name: <input type="text" name="fname" spellcheck="true">
```

```
</body>
```

```
</html>
```

Elementos audio e video, e codecs

- Áudio

- `<audio src="mus.oga" controls="true" autoplay="true" />`

- Origens alternativas de áudio:

- `<audio controls="true" autoplay="true">`

- `<source src="mus.oga" />`

- `<source src="mus.mp3" />`

- `<source src="mus.wma" />`

- `</audio>`

Elementos audio e video, e codecs

- Vídeo

O uso de vídeo é muito semelhante ao de áudio:

```
<video src="u.ogv" width="400" height="300" />
```

E com vários elementos source:

```
<video controls="true" autoplay="true" width="400" height="300">
```

```
<source src="u.ogv" />
```

```
<source src="u.mp4" />
```

```
<source src="u.wmv" />
```

```
<p>Faça o <a href="u.mp4">download do vídeo</a>.</p>
```

```
</video>
```

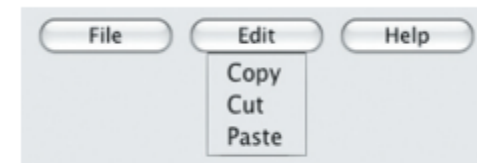
Menus e toolbars

- O elemento menu é usado para definir menus e barras de ferramenta. Dentro do menu, você pode inserir submenus ou comandos. Para inserir submenus, basta inserir outros elementos menu. Para definir comandos, você pode inserir:
 - Um link, um elemento a com atributo href;
 - Um botão, um elemento button;
 - Um botão, um elemento input com o atributo type contendo button, submit, reset ou image;
 - Um radiobutton, um elemento input com o atributo type contendo radio;
 - Um checkbox, um elemento input com o atributo type contendo checkbox;
 - Um elemento select, contendo um ou mais options, define um grupo de comandos
 - Um elemento qualquer com o atributo accesskey
 - Um elemento command

Menus e toolbars (não fica certo no Chrome)

```
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8" />
<title>Menus</title>
</head>
<body>
<menu type="toolbar">
<li>
<menu label="File">
<button type="button" onclick="fnew()">New...</button>
<button type="button" onclick="fopen()">Open...</button>
<button type="button" onclick="fsave()">Save</button>
<button type="button" onclick="fsaveas()">Save as...</button>
</menu>
</li>
```

```
<li>
<menu label="Edit">
<button type="button" onclick="ecopy()">Copy</button>
<button type="button" onclick="ecut()">Cut</button>
<button type="button" onclick="epaste()">Paste</button>
</menu>
</li>
<li>
<menu label="Help">
<li><a href="help.html">Help</a></li>
<li><a href="about.html">About</a></li>
</menu>
</li>
</menu>
</body>
</html>
```



Documentação completa

- <https://www.w3c.br/pub/Cursos/CursoHTML5/html5-web.pdf>