

O objetivo do uso de tabelas em documentos HTML é a apresentação de dados tabulares. Para tanto, existem diversos elementos para formatar as tabelas de forma clara, tanto para o usuário comum, quanto para os usuários portadores de deficiência.

No entanto, uma simples verificação por algumas páginas mostra que em várias tabelas, o uso de elementos é reduzido ao básico: TABLE, TR, TD; dificultando o seu acesso.

As orientações que seguem visam esclarecer o uso dos elementos de tabela visando a melhoria da acessibilidade da apresentação de dados em forma de tabelas.

### **1. Lendo uma tabela**

---

Para pessoas sem problemas de visão, as informações contidas na tabela são compreendidas apenas avaliando o conteúdo da tabela, dados específicos são encontrados cruzando visualmente colunas e linhas.

No entanto para pessoas portadoras de deficiência visual, a compreensão e obtenção de dados de uma tabela não é uma tarefa fácil se essa tabela for construída de forma não-acessível. Pessoas portadoras de deficiência visual utilizam-se geralmente de leitores de tela. Os leitores de tela não lêem a 'tela' e sim decodificam a sua estrutura HTML.

Dessa forma, o uso de elementos HTML que estruturam de forma clara a tabela auxiliam a sua leitura por pessoas que se utilizam leitores de tela.

### **2. Aviso: tabelas não são para diagramar**

---

Apesar de criado com a finalidade de apresentar dados tabulares, o elemento TABLE é muito utilizado para a diagramação dos elementos visuais das páginas. Com o advento do CSS (folhas de estilo em cascata - linguagem voltada para a formatação visual dos elementos HTML) o uso de tabelas para solucionar o desenho de páginas deveria ter sido descartado. No entanto, tabelas continuam sendo usadas com esse objetivo.

O uso de tabelas para diagramar as páginas causa problemas não só a acessibilidade, mas a performance de carregamento da página.

Atualmente há um movimento de esclarecimento sobre o uso correto das linguagens CSS e HTML. Dessa forma tem-se observado um decréscimo de páginas que utilizam tabelas na diagramação.

Lembre-se: os elementos visuais e a diagramação das páginas devem ser controlados a partir do uso adequado das folhas de estilo (CSS).

### 3. Elementos de uma tabela

---

A seguir veremos os elementos e atributos disponíveis para a formatação de tabelas acessíveis:

#### **CAPTION para o título da tabela**

O elemento CAPTION é o título da tabela. O uso de outros elementos como H1, P, TD ou TH pode ter “visualmente” o sentido de título, mas não são semanticamente corretos, e tampouco, acessíveis.

No código HTML o elemento CAPTION deve ser colocado após a marcação TABLE e antes de qualquer outra marcação que seguir.

Por padrão, o elemento CAPTION é mostrado centralizado logo acima da tabela. Para modificações no visual deve ser usado o CSS.

Ex:

```
<table sumary="" >  
<caption>Material escolar</caption>  
<tr>...segue o resto da tabela...
```

#### **Atributo summary para a finalidade da tabela**

O atributo “summary” deve vir dentro do elemento TABLE servindo como informação auxiliar para entendimento da tabela para leitores de tela e displays Braille, não sendo visível em navegadores de interface gráfica.

O atributo “summary” descreve a finalidade da tabela e dá uma indicação da estrutura geral, sendo necessário para compreensão de tabelas complexas.

Ex:

```
<table summary="Material escolar - Levantamento comparativo de preços.">  
<caption>Material escolar </caption>  
<tr>...segue o resto da tabela...
```

### **TH para identificar os cabeçalhos, TR para linha e TD para dado de tabela**

Para tabelas simples, o uso apropriado do elemento TH é essencial para tornar a tabela acessível. Utilize o elemento TH para a identificação de cabeçalhos em linhas e colunas pelos leitores de tela. O elemento TR marca a linha da tabela e o elemento TD marca o conteúdo da célula como dado.

Deve-se evitar o uso de colunas com células vazias, pois elas atrapalham a leitura dos dados pelos leitores de tela. As folhas de estilo devem ser utilizadas para fins de apresentação diferenciada da tabela ao invés de colunas e linhas de células vazias.

Ex:

```
<table summary="Material escolar - Levantamento comparativo de preços.">  
<caption>Material escolar</caption>  
  <tr>  
    <th>Estojo caneta hidrocor / loja</th>  
    <th>6 unidades</th>  
    <th>12 unidades</th>  
    <th>24 unidades</th>  
    <th>36 unidades</th>  
  </tr>  
  <tr>  
    <th>Lisboa papelaria</th>  
    <td>R$ 6,20</td>  
    <td>R$ 9,99</td>  
    <td>R$ 11,90</td>  
    <td>R$ 16,90</td>  
  </tr>  
<tr>...segue o resto da tabela...
```

## Abbr

O atributo 'abbr' permite a abreviação de um cabeçalho longo de modo que ele não seja lido por inteiro toda vez que o leitor de tela o encontrar, lendo apenas a abreviação nas demais vezes.

Ex:

```
<table sumary="Material escolar - Levantamento comparativo de preços.">  
<caption>Material escolar</caption>  
<tr>  
<th abbr="hidrocor">Estojo caneta hidrocor / loja</th>  
<th>...segue o resto da tabela...
```

## THEAD, TBODY e TFOOT grupos de linhas para tabelas extensas

- Os elementos THEAD, TBODY E TFOOT são necessários em tabelas extensas, que ocupam mais de uma página.

O elemento THEAD agrupa uma ou mais linhas de cabeçalho da tabela. O elemento TFOOT agrupa linhas com informações de rodapé. No código HTML o elemento TFOOT deve aparecer antes do elemento TBODY.

O elemento TBODY define o corpo da tabela que contém as células de dados. Uma tabela pode conter mais de um elemento TBODY separando os diferentes grupos de dados.

O uso desses elementos permite que:

- As linhas contidas nos elementos THEAD e TFOOT sejam fixas no navegador, permitindo que as células de dados contidas no TBODY 'rolem' entre as duas;
- Que quando impressas, as tabelas mostrem o cabeçalho e o rodapé em todas as páginas.

Os elementos THEAD, TFOOT e TBODY de uma tabela devem ter o mesmo número de colunas.

Ex:

```
<table summary="....">
<caption>.....</caption>
<thead>
<tr> ...informação contida no cabeçalho... </tr>
</thead>
<tfoot>
<tr>...informação contida no rodapé... </tr>
</tfoot>
<tbody>
<tr> ...primeira linha de dados do bloco um... </tr>
<tr> ...segunda linha de dados do bloco um...</tr>
</tbody>
<tbody>
<tr> ...primeira linha de dados do bloco dois...
<tr> ...segunda linha de dados do bloco dois...</tr>
<tr> ...terceira linha de dados do bloco dois...</tr>
</tbody>
</table>
```

### **Agrupando colunas – elementos COLGROUP e COL**

O elemento COLGROUP cria um grupo/estrutura de colunas, permitindo sua diferenciação. A tabela do próximo exemplo contém dois grupos de colunas. O primeiro grupo de colunas contém 10 colunas e o segundo contém 5 colunas.

Ex:

```
<table summary="...">
<caption>...</caption>
<colgroup span="10">
<colgroup span="5">
<thead>
<tr><td> ...
</table>
```

O número de colunas contidas no grupo pode se especificado de duas maneiras diferentes:

- Utilizando o atributo "span";
- Utilizando o elemento COL;

Ex: Um grupo de 60 colunas com as mesmas características:

- Utilizando o atributo "span"  

```
<colgroup span="60" id="cor">  
</colgroup>
```
- Utilizando o elemento COL  

```
<colgroup>  
<col id="cor">  
<col id="cor">...segue até completar 60 elementos col...  
</colgroup>
```

O atributo "span" costuma ser mais vantajoso em grupos de colunas com as mesmas características. O elemento COL é um elemento vazio, sem função estrutural e serve como suporte para os atributos. O elemento COL pode estar no interior ou no exterior de um grupo explícito de colunas - COLGROUP.

O atributo "span" também pode ser utilizado no elemento COL, sempre que seja necessário isolar uma coluna no interior de um grupo.

### **Atributos id e headers - Uma forma de associar cabeçalhos e células de dados**

O atributo "headers" é utilizado nas células de tabelas (<td></td>) em conjunto com o atributo id na célula de cabeçalho (<th></th>) para associar as células de dados aos seus respectivos cabeçalhos.

O atributo headers é requerido sempre que os cabeçalhos estiverem situados em posições irregulares, em relação aos dados aos quais eles se referem.

Ex:

```
<table summary="Oferta de cursos - tabela estruturada em: nome do curso,
professor, resumo da disciplina e carga horária">
  <caption>Oferta de cursos - Primeiro semestre</caption>
  <tr>
    <th id="t1" abbr="Nome">Nome do Curso</th>
    <th id="t2" >Professor</th>
    <th id="t3" >Resumo</th>
    <th id="t4" abbr="carga">Carga horária</th>
  </tr>
  <tr>
    <td header="t1">Gravura - conceitos básicos</td>
    <td header="t2">Eunice Lobato</td>
    <td header="t3">História da gravura, estilos, gráfica. Visita aos ateliês.
</td>
    <td header="t4">40h</td>
  </tr>
  <tr>
    <td header="t1">Gravura em metal</td>
    <td header="t2">Ione Rocha</td>
    <td header="t3">Trabalhos práticos de gravura em metal. </td>
    <td header="t4">160h</td>
  </tr>
  <tr>
    <td header="t1">Xilogravura</td>
    <td header="t2">Heli Santos</td>
    <td header="t3">Trabalhos práticos em xilogravura.</td>
    <td header="t4">160h</td>
  </tr>
</table>
```

### **Atributo scope -Uma outra forma de associar cabeçalhos e células de dados**

De forma semelhante aos atributos "id" e "header", o uso do atributo scope é uma outra forma de se agrupar cabeçalhos de colunas com suas respectivas informações e assim melhorar a acessibilidade das tabelas de dados.

O atributo "scope" especifica o grupo de células de dados para qual a célula de cabeçalho (<th></th>) está associada. Este atributo é usado no lugar do atributo "header", preferencialmente em tabelas de dados simples. O atributo possui os seguintes valores:

- row: Fornece informação de cabeçalho referente a linha que a contém;
- col: Fornece informação de cabeçalho referente a coluna que a contém;
- rowgroup: Fornece informação de cabeçalho referente ao grupo de linhas que a contém;
- colgroup: Fornece informação de cabeçalho referente ao grupo de colunas que a contém.

O atributo "scope" também pode ser associado, quando necessário, ao elemento de célula de dados (<td></td>). Os leitores de tela entenderão a célula como o cabeçalho da linha.



Ex:

```
<table summary="Oferta de cursos - tabela estruturada em: nome do curso,
professor, resumo da disciplina e carga horária">
  <caption>Oferta de cursos - Primeiro semestre</caption>
  <tr>
    <th scope="col" abbr="Nome">Nome do Curso</th>
    <th scope="col" >Professor</th>
    <th scope="col">Resumo</th>
    <th scope="col" abbr="carga">Carga horária</th>
  </tr>
  <tr>
    <td scope="row">Gravura – conceitos básicos</td>
    <td>Eunice Lobato</td>
    <td>História da gravura, estilos, gráfica. Visita aos ateliês. </td>
    <td>40h</td>
  </tr>
  <tr>
    <td scope="row">Gravura em metal</td>
    <td>Ione Rocha</td>
    <td>Trabalhos práticos de gravura em metal. </td>
    <td>160h</td>
  </tr>
  <tr>
    <td scope="row">Xilogravura</td>
    <td>Heli Santos</td>
    <td>Trabalhos práticos em xilogravura.</td>
    <td>160h</td>
  </tr>
</table>
```

## 4. Tabelas simples X tabelas complexas

---

Tabelas contendo níveis simples de cabeçalhos de colunas e/ou linhas são mais fáceis de serem acessadas que tabelas complexas com múltiplos níveis de cabeçalhos (ex: três níveis de cabeçalhos para colunas e dois níveis para linhas).

Mesmo com todos os elementos que proporcionem a acessibilidade, tabelas complexas são de difícil compreensão para o usuário que utiliza leitores de tela. Sempre que possível devem ser usadas tabelas simples.

Quando possível, as tabelas complexas devem ser divididas em tabelas simples.

## 5. Leitura recomendada

---

### **HTML 4.01 Specification - Tables**

<http://www.w3.org/TR/html4/struct/tables.html>

### **Tabelas de dados acessíveis**

<http://www.maujor.com/tutorial/actables.php>