Creating Table Styles in CSS

Matthew Ellison
matthew@ellisonconsulting.com
What we’ll cover in this session

- Components of an HTML-based table
- Why is table formatting a challenge?
- What are the basic formatting options for tables?
- To which elements do we apply the formatting properties?
- Descendant selectors in CSS
- Putting it altogether to create “table styles”
## Working with Source Documents

There are two types of source documents in Doc-To-Help: Word or HTML. You may edit these documents in Microsoft® Word, Microsoft® FrontPage®, or Adobe® Dreamweaver®. This makes it possible for you to work in the environment you prefer while efficiently creating the output you need.

<table>
<thead>
<tr>
<th>Source Document</th>
<th>File Type(s)</th>
<th>Editor</th>
<th>Ribbons/Toolbars</th>
</tr>
</thead>
<tbody>
<tr>
<td>Microsoft® Word</td>
<td>.doc, .docx</td>
<td>Microsoft Word</td>
<td>ComponentOne Doc-To-Help and Doc-To-Help Special Formatting (added when Doc-To-Help is installed)</td>
</tr>
<tr>
<td>HTML</td>
<td>.html, .htm</td>
<td>Microsoft FrontPage®, Adobe® Dreamweaver®, your choice</td>
<td>ComponentOne Doc-To-Help D2HML Styles (added when Doc-To-Help is installed; Microsoft FrontPage and Adobe Dreamweaver only)</td>
</tr>
</tbody>
</table>

See [Editing Word Documents](#) and [Editing HTML Documents](#) for more on working with Word and HTML documents.

**D2HML styles** are used to create links, inline text, etc. in Word and HTML documents. For more on D2HML see [Using D2HML](#).

**More:**
- [Editing Word Documents](#)
- [Editing HTML Documents](#)
- [Setting Document Properties](#)
- [Creating a Glossary](#)
# Tables in Help

## Topic Toolbars

The **Topic Toolbars** give you quick access to commonly used commands in the authoring environment.

Commands for working with topic content are located on the left of the Topic Editor toolbars.

<table>
<thead>
<tr>
<th>This button</th>
<th>Does this</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Save" /></td>
<td>Saves the topic</td>
</tr>
<tr>
<td><img src="image" alt="Undo" /></td>
<td>Reverses the last action in the topic</td>
</tr>
<tr>
<td><img src="image" alt="Redo" /></td>
<td>Reverses the last action applied by the Undo button</td>
</tr>
<tr>
<td><img src="image" alt="Insert New Link" /></td>
<td>Creates a new Hyperlink, or edit a Hyperlink applied to selected text</td>
</tr>
<tr>
<td><img src="image" alt="Insert Existing Link" /></td>
<td>Inserts an existing Hyperlink</td>
</tr>
<tr>
<td><img src="image" alt="Urlink" /></td>
<td>Removes a Hyperlink from selected text</td>
</tr>
<tr>
<td><img src="image" alt="Insert Image" /></td>
<td>Creates a new image, or edit the selected image</td>
</tr>
<tr>
<td><img src="image" alt="Insert Existing Image" /></td>
<td>Inserts an existing image</td>
</tr>
<tr>
<td><img src="image" alt="Insert a File Object from Thumbnails" /></td>
<td>Inserts an existing image using an image preview</td>
</tr>
<tr>
<td><img src="image" alt="Cut" /></td>
<td>Removes the selected content from the topic</td>
</tr>
<tr>
<td><img src="image" alt="Copy" /></td>
<td>Copies the selected content to the clipboard</td>
</tr>
<tr>
<td><img src="image" alt="Paste" /></td>
<td>Pastes the content from the clipboard into the topic</td>
</tr>
<tr>
<td><img src="image" alt="Find" /></td>
<td>Finds characters or words in your topic</td>
</tr>
<tr>
<td><img src="image" alt="Find/Replace" /></td>
<td>Finds and replaces characters or words in your topic</td>
</tr>
</tbody>
</table>

Commands for adding custom characters and viewing the topic are located on the right of the Topic Editor.
## Components of an HTML-based table

**Table** `<table>`

**Header cell** `<th>`

**Caption** `<caption>`

### Code-page support in Microsoft Windows

<table>
<thead>
<tr>
<th>Code-Page ID</th>
<th>Name</th>
<th>ACP</th>
<th>OEMCP</th>
<th>Windows NT 3.1</th>
<th>Windows NT 3.51</th>
<th>Windows 95</th>
</tr>
</thead>
<tbody>
<tr>
<td>1200</td>
<td>Unicode (EUC of ISO/IEC 10646)</td>
<td>X</td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>1250</td>
<td>Windows 3.1 Eastern European</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>1251</td>
<td>Windows 3.1 Cyrillic</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1252</td>
<td>Windows 3.1 US (ANSI)</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>1253</td>
<td>Windows 3.1 Greek</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>1254</td>
<td>Windows 3.1 Turkish</td>
<td>X</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>1255</td>
<td>Hebrew</td>
<td></td>
<td></td>
<td>X</td>
<td></td>
<td></td>
</tr>
<tr>
<td>437</td>
<td>MS-DOS United States</td>
<td></td>
<td></td>
<td>X</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>708</td>
<td>Arabic (ASMO 708)</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>709</td>
<td>Arabic (ASMO 449/82/CON V4)</td>
<td></td>
<td></td>
<td></td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

**Header section** `<thead>`

**Body section** `<tbody>`

**Row** `<tr>`

**Data cell** `<td>`

**Column** `<col>`

**Column group** `<colgroup>`
Why is table formatting a challenge?

- Tables contain multiple nested elements
- Each nested element has its own formatting properties
  - **Table**: width, borders, background, etc.
  - **Caption**: font, colour, margin, position, etc.
  - **Column**: width
  - **Table header**: background
  - **Table cells**: alignment, padding, borders, background, font, colour, etc.
- How can we control all of this with a single style?
My objectives for table formatting

- Simplify my table mark-up!

```html
<table x-use-null-cells style="x-cell-content-align: top; width: 100%; border-spacing: 0px;"
cellspacing=0 width=100%>
  <col style="width: 20%;"/>
  <col style="width: 20%;"/>
  <col style="width: 20%;"/>
  <col style="width: 20%;"/>

  <tr valign=top><td style="border-left-width: 1px; border-top-style: Solid; border-top-color: #008000; border-top-width: 3px; border-right-width: 1px; border-bottom-style: Solid; border-bottom-width: 1px; border-bottom-color: #008000; padding-right: 10px; padding-left: 10px;" width=20%>
    <p>Code-Page<br />ID</p>
  </td>
  <td style="border-top-style: Solid; border-top-color: #008000; border-top-width: 3px; border-right-width: 1px; border-bottom-style: Solid; border-bottom-width: 1px; border-bottom-color: #008000; padding-right: 10px; padding-left: 10px;" width=20%>
    <p>Name</p>
  </td>
</tr>
</table>
```
How do we apply the formatting?

- Using CSS of course!

- My objectives for table formatting:
  - No formatting information within HTML code
  - Formatting all within linked CSS file
  - Should be as easy as possible to apply a specific look-and-feel to an entire table
Strategy

- Use CSS properties instead of HTML attributes

Examples:

- Instead of:
  
  ```html
  <table width="100%">
  </table>
  ```

  Use:
  
  ```css
  table { width: 100%; }
  ```

- Instead of:
  
  ```html
  <tr valign="top">
  </tr>
  ```

  Use:
  
  ```css
  tr { vertical-align: top }
  ```
Strategy

- Avoid inline CSS
- Example:
  - Instead of:
    ```html
    <td style="border-left: 1px Solid #008000;">
    </td>
    ```
  - Use:
    ```html
    <td>
    td
    {border-left:1px Solid #008000;}
    ```
The basic table formatting options

- Width (table and columns)
- Height (table and rows)
- Background colour (table, header, rows, cells)
- Borders (table frame and individual cells)
- Border-collapse (table property)
- Padding (within cells)
- Horizontal alignment (of data within cells)
- Vertical alignment (of data within cells)

Demonstration
Border collapse property

<table>
<thead>
<tr>
<th>Real Name</th>
<th>Spice Name</th>
<th>Year of Birth</th>
<th>Likes</th>
</tr>
</thead>
</table>
| Melanie Jane Chisholm | Sporty     | 1975          | • Getting up early  
|                     |            |               | • Breakfast                        |
| Emma Lee Bunton      | Baby       | 1976          | • Eating doughnuts  
|                     |            |               | • Pop/garage music                |
| Geraldine Estelle Halliwell | Ginger | 1972          | • Talking                        |
| Victoria Caroline Beckham | Posh   | 1975          | • Designer clothes  
|                     |            |               | • Posing for the cameras            |
| Melanie Janine Brown | Scary     | 1975          | • Rap  
|                     |            |               | • Hip hop  
|                     |            |               | • Jungle                          |

Table

{border-collapse: separate}

Table

{border-collapse: collapse}

Matthew Ellison Consulting
Which elements do we format?

- Make use of inheritance

<table>
<thead>
<tr>
<th>table</th>
</tr>
</thead>
<tbody>
<tr>
<td>thead</td>
</tr>
<tr>
<td>tr</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>tbody</th>
</tr>
</thead>
<tbody>
<tr>
<td>tr</td>
</tr>
</tbody>
</table>
Which elements do we format?

- Apply formatting properties to the highest possible element
  
  ```
  table { text-align: left; }
  tr { vertical-align: middle; }
  tbody { background-color: #FFC0FF; }
  ```

- Golden rules about inheritance:
  - Font properties, colour and alignment are inherited
  - "Box" properties (margin, border, padding), width and height are not inherited
Where to apply some key properties

- Table border/frame:
  - `table`
- Cell borders and padding:
  - `td`
  - `th`
- Row borders (not supported by IE):
  - `tr`
- Header background colour:
  - `thead` or `th`
- Cell background colour:
  - `tbody`, `tr` or `td`
- Horizontal alignment (`text-align`):
  - `table` or `tbody`
- Vertical alignment (`vertical-align`):
  - `tbody` or `tr`
Problem

- We can define properties for standard table elements:
  
  ```
  table { width 400px, ... }
  tbody { ... }
  tr { ... }
  td { ... }
  etc.
  ```

- BUT all our tables will look the same
Possible solution

- Use CSS classes:

```
<table>
<thead>
<tr>
<th>
...
</th></tr>
</thead>
<tbody>
<tr>
<td>
...
</td>
</tr>
</tbody>
```

```
table { ... }
thead { ... }
tbody { ... }
tr { ... }
th { ... }
td { ... }
```
Possible solution

- Use CSS classes:

```html
<table class="formal">
<thead class="formal">
<tr class="formal">
<th class="formal">
...
</th>
</tr>
</thead>
<tbody class="formal">
<tr class="formal">
<td class="formal">
...
</td>
</tr>
</tbody>
</table>
```

```css
.table.formal { ... }
.thead.formal { ... }
.tbody.formal { ... }
.tr.formal { ... }
.th.formal { ... }
.td.formal { ... }
```
New problem

- Our table mark-up is rather complicated
- To change the look-and-feel of the table, we need to replace all the class attributes in the HTML
Introducing descendant selectors

- Sometimes known as contextual selectors
- Simple example:

  ```
  h1 { font-weight: bold }
  em { font-weight: bold }
  h1 em { font-style: italic }
  ```

  those em elements that are contained by an h1 element

Descendant selectors are very useful

- See www.w3.org/TR/REC-CSS2/selector.html#descendant selectors
How this helps with tables

- Apply class attribute to table element only
- Use descendant selectors to specify formatting for all nested elements

**HTML**

```html
<table class="formal">
  <thead class="formal">
    <tr class="formal">
      <th class="formal">
        …
      </th>
    </tr>
  </thead>
  <tbody class="formal">
    <tr class="formal">
      <td class="formal">
        …
      </td>
    </tr>
  </tbody>
</table>
```

**CSS**

```css
table.formal { … }  
thead.formal { … }  
tbody.formal { … }  
tr.formal { … }  
th.formal { … }  
td.formal { … }  
```
How this helps with tables

- Apply class attribute to table element only
- Use descendant selectors to specify formatting for all nested elements

```html
<table class="formal">

<thead>
  <tr>
    <th>…</th>
  </tr>
</thead>

<tbody>
  <tr>
    <td>…</td>
  </tr>
</tbody>
```

```css
.table.formal { ... }  
.table.formal thead { ... }  
.table.formal tbody { ... }  
.table.formal tr { ... }  
.table.formal th { ... }  
.table.formal td { ... }  
```
Help Authoring Tool issues

- RoboHelp 7:
  - Does not support `thead` and `tbody` elements
  - HTML Editor does not represent table formatting accurately (OK when published)

- Flare 3:
  - Has its own proprietary "table styles" as an alternative strategy
  - Provides easy way to alternate row colours
  - Each table style represented by a separate CSS file
Summary

- Keep table coding as simple as possible
- Avoid using element attributes for table formatting
- Use external CSS
- Apply class attribute to top-level element only
- Use CSS descendant selectors for nested elements