

Tables in HTML: A tutorial



John W. Shipman

2006-05-18 14:41

Table of Contents

1. Basic table building	1
1.1. A simple table	1
1.2. Captioning a table	2
1.3. Specifying dimensions	2
1.4. Specifying horizontal alignment of cells	2
1.5. Specifying vertical alignment for a whole row	3
2. Attributes of the <table> tag	3
3. Column and row headings	4
4. Spanned cells	4
5. A look ahead: HTML 4.01 tables	5
5.1. Specifying the alignment for an entire column	5
5.2. Column grouping	6
5.3. Table headers and footers	6

1. Basic table building

We recommend that you use the older 3.0 table standard described here, at least until more browsers support the current HTML 4.01 standard¹.

At this writing, the current versions of most Web browsers do not support many of the advanced features described below in the section below; see Section 5, "A look ahead: HTML 4.01 tables" (p. 5). Test features in several browsers before relying on them.

This publication is available in Web form² and also as a PDF document³. Please forward any comments to tcc-doc@nmt.edu.

1.1. A simple table

Here is a trivial table of two rows and two columns:

```
<table>
  <tr><td>New Mexico <td>Santa Fe
  <tr><td>Utah <td>Salt Lake City
</table>
```

- The entire table is contained between <table> and </table> tags.

¹ <http://www.w3.org/TR/html401/>

² <http://www.nmt.edu/tcc/help/pubs/htmltables/>

³ <http://www.nmt.edu/tcc/help/pubs/htmltables.pdf>

- Each table *row* starts with a `<tr>` tag.
- Each *cell* starts with a `<td>` tag. A cell is one column of one row.

1.2. Captioning a table

To add a title to a table, place the caption text between `<caption>` and `</caption>` tags just after the `<table>` tag:

```
<table>
  <caption>Southwestern Capitals</caption>
  <tr><td>New Mexico <td>Santa Fe
  <tr><td>Utah <td>Salt Lake City
</table>
```

You can use an `align="bottom"` attribute in the `<caption>` tag to position the caption below the table. The default placement is at the top. For example:

```
<caption align="bottom">Southwestern Capitals</caption>
```

1.3. Specifying dimensions

In the tags that make up a table, there are a number of attributes that allow you to specify the relative or absolute sizes of things. For example:

```
<table width="75%">&hellip;</table>
```

The `width` attribute of a `<table>` tag specifies how wide the table is, relative to the size of the browser window. The above example says to make the table 3/4 the width of the window.

The next example says to make the table five and a half inches wide:

```
<table width="5.5in">...</table>
```

Other dimensions supported include:

- **cm** for centimeters
- **mm** for millimeters
- **pi** for the printer's pica (about 1/6")
- **pt** for the printer's point (about 1/72")
- **em** for a printer's em, equal to the default font size (e.g., 11 points for an 11-point font)
- **px** for screen pixels

1.4. Specifying horizontal alignment of cells

Usually, each item is left-justified in its cell, although this depends on the browser. If the item in a cell is a long text string, the browser may wrap it over multiple lines to fit.

Any cell can include an `align="align-type"` attribute to tell the browser how to position the item in that cell. The `align-type` can be any of:

- `<td align="left">` to left-justify the contents of the cell.
- `<td align="center">` to center the contents of the cell.
- `<td align="right">` to right-justify the contents of the cell.

- `<td align="justify">` to justify the text; that is, if the item is a text string too long to fit on one line, all the lines but the last will be stretched to form straight left and right margins.
- You can use `<td align="char">` for columns of numbers with decimals so that they are aligned on the decimal point.

Warning

Hardly any browsers support this feature at the present.

For example, in this table, all the decimal points should line up:

```
<table>
  <tr><td>kilo  <td align="char">1000.
  <tr><td>deci  <td align="char">0.1
  <tr><td>centi <td align="char">0.01
  <tr><td>milli <td align="char">0.001
```

You can include a `char="c"` attribute to specify an alignment character `c` to use instead of other than a period.

For example, this would line up the colons (:) in that column:

```
<td align="char" char=":">...
```

You can also include a `charoff="dimension"` attribute that says where to align things relative to the cell size. For example, to specify that the point of alignment 1/3 of the width from the left column:

```
<td align="char" charoff="33%">...
```

1.5. Specifying vertical alignment for a whole row

You can control where the items are placed vertically in their cells by using a `valign` attribute in the `<tr>` tag for that row. The `valign` attribute can take these values:

- `<tr valign="top">` says to put each item at the top of its cell.
- `<tr valign="middle">` says to center each item vertically within its cell.
- `<tr valign="bottom">` says to put each item at the bottom of its cell.

Again, a `<td>` tag can use a different `valign` attribute to override the row's vertical alignment in that one cell.

2. Attributes of the `<table>` tag

The `<table>` tag allows a number of optional attributes that specify the properties of the entire table.

- `<table align="left">` positions the table at the left side of the viewer window; `align="center"` centers it; and `align="right"` places it on the right side.
- The `width="dimension"` attribute sets the width of the entire table; it is discussed above in the section on dimensions.
- `border="dimension"` tells the browser to draw a border around the entire table. The `dimension` is optional and specifies how thick to make the border. For example: `<table border="5px">` specifies a five-pixel border around the table.

- **frame="frame-type"** specifies which sides of the frame to show. The **frame-type** can be any of:
 - **void**: No frame.
 - **above**: Only the top.
 - **below**: Only the bottom.
 - **hsides**: The top and bottom (horizontal sides).
 - **lhs**: For the left side.
 - **rhs**: For the right side.
 - **vsides**: For left and right sides.
 - **box**: For all four sides.
- **rules="rule-type"** controls the placement of ruled lines within the table. The **rule-type** can be any of:
 - **rules="basic"** for rules between the header, body, and footer (see below);
 - **rules="rows"** for rules between rows;
 - **rules="cols"** for rules between columns; and
 - **rules="all"** for rules both between rows and between columns.
- **cellspacing="dimension"** says how much additional space to put between the cells of a table.
- **cellpadding="dimension"** says how much additional space to put between the walls of a cell and its contents. For example, this tag would insure that there is always 1/4" between the text in a cell and its borders:

```
<table cellpadding="0.25in">
```

3. Column and row headings

To put one or more column headings on your table, just create rows for them using the `<tr>` tag, but instead of using `<td>` tags for the cells of that row, use `<th>` tags.

Here is an example of column headings:

```
<table>
  <tr><th>Element name<th>Atomic number
  <tr><td>Hydrogen<td>1
</table>
```

You can also provide row headings by starting each row with a `<th>` tags and then using `<td>` tags for the remainder of the row.

4. Spanned cells

The items in your table are not necessarily restricted to the size of one cell.

Any cell (`<th>` or `<td>` tag) can have a **rowspan="integer"** or **colspan="integer"** attribute to "span" more than one cell of the table, that is, place that element's content into an area that would normally be occupied by two or more cells.

For example, an item preceded by the tag `<td colspan="2">` will be placed into two adjacent cells in the same row. The tag `<th rowspan="5">` might be used to place a heading that covers five rows of the table.

Refer to the URL given at the beginning of this document for full details of how cell spanning works.

Here is a brief example showing spanned column headings.

```
<table>
  <tr><th colspan="2" align="center">Lifespan
  <tr><th>Name <th>Born <th>Died
  <tr><td>Louis ``Satchmo'' Armstrong <td>1900<td>1971
  ...
</table>
```

In this table, the word "Lifespan" will appear on the first heading row, centered over the second and third columns. The second heading row will contain the column headings "Name", "Born", and "Died".

5. A look ahead: HTML 4.01 tables

Warning

Most browsers don't support the HTML 4.01 table specification yet. Here are some features you will be able to use once they do.

5.1. Specifying the alignment for an entire column

It's pretty tedious to add an `align="..."` attribute in each cell of a column. A better way is use a group of `<col>` tags before the first row of the table; each of these tags specifies the alignment for that whole column. Here is an example:

```
<table>
  <caption>Atomic numbers</caption>
  <col align="left">
  <col align="right">
  <tr><td>Hydrogen<td>1
  <tr><td>Helium<td>2
</table>
```

The first `<col>` tag says that the first column's cells will be left-justified. The second `<col>` tag says to right-justify the cells of the second column.

You can make a `<col>` tag apply to more than one adjacent column by using the `span=n` attribute to span `n` columns. For example, suppose you have a table with three left-justified columns followed by two centered columns. That table might start like this:

```
<table>
  <col align="left" span="3">
  <col align="center" span="2">
  <tr>...
</table>
```

You can still override the alignment in a particular cell by using an `align` attribute in that cell's `<td>` tag. That attribute will affect only that one cell.

The `<col>` tag also has a **`width="dimension"`** attribute that you can use to define the width of the column. For example:

```
<table>
  <col width="4cm">
  <col width="2cm" align="right" span="3">
  <tr>...
</table>
```

5.2. Column grouping

If you have groups of columns with the same attribute values, you can use a `<colgroup>` tag before each group of `<col>` tags to specify attributes for that group. For example:

```
<table>
  <colgroup align="center">
    <col width="3pi">
    <col width="5pi" span="4">
  <colgroup align="right">
    <col width="4pi">
    <col width="8pi">
    ...
</table>
```

The example above defines seven columns. The first five are all centered and the next two are right-justified.

5.3. Table headers and footers

If you wish, you can designate a “header” section that appears at the top of the table. You may also create a “footer” section that appears at the bottom. For very long tables, some browsers may render a table into separate pages, and they will place the header and footer on each page. This allows you to have running column heads on a multi-page table.

To create a header, use a `<thead>` tag after any `<col>` tags, followed by any table rows that are part of the header.

If there is a footer, use a `<tfoot>` tag after the header section (if any), followed by any table rows that are part of the footer. *The footer must precede the body section.*

If either a header or footer is used, place a `<tbody>` tag before the actual table body.

Here is an example of a table with a header and footer:

```
<table>
  <col align="left">
  <col align="right">
  <thead>
    <tr><th>Element name<th>Atomic number
  <tfoot>
    <tr><th span="7">Does not include the new elements made
      in Germany last week.
  <tbody>
    <tr><td>Hydrogen<td>1
    ...
</table>
```