

CS120

Introduction to Web Site Development

Lecture 4 - Favicons, Tables, Frames

Gabor Kondas

California State University, Los Angeles

(Based on slides by John Hurley)

Favicons

- Site addresses often appear in browser URL windows with small icons
- These are called favicons
- They are traditionally in the .ico file format, although they can now be in .gif or .png with some browsers
- The image must be perfectly square and is typically 16x16 or 32x32
- Put the favicon file in the root directory of the site

Favicons

html syntax to link favicon to document:

- `<link rel="icon" href="[path]/favicon.ico" />`
 - Where [path] is the relative path to the favicon file.
- Favicons may not appear correctly when not placed in the root directory of the site. On cs1, this will be your `public_html` directory. It's not clear what it would be on your hard drive, so don't bother to try to get favicons to work locally. Just make them work on the server.
- On a server in which you control the domain name (eg. `www.mysite.com`), the path is normally just `"/"` but on our cs1 accounts, think out the relative path to `public_html`
- Make sure the file name is `favicon.ico`

TABLE Element

- **Description:** Use the table element to place data in a two-dimensional table
- **Placement:** Flow Content (i.e. don't put a table inside a paragraph or in-between text)
- **Content:** caption, colgroup, thead, tfoot, tbody, tr, td elements

TABLE Element

- Obvious uses: some information lends itself to presentation in tables:
 - <http://www.myguideireland.com/distance-chart-in-miles>
- Tables can be used as the top-level layout of a page
 - <http://craigslist.org>
- Tables can be a little confusing to set up, but the formatting is very rigid and thus predictable
- Page layout using tables is regarded as a little old fashioned. Layout using CSS (which we'll learn about soon) is more powerful

The TABLE Element (Syntax)

```
<table>  
  <tr>  
    <td>1<sub>1</sub></td>  
    <td>1<sub>2</sub></td>  
  </tr>  
  <tr>  
    <td>2<sub>1</sub></td>  
    <td>2<sub>2</sub></td>  
  </tr>  
</table>
```

THEAD Element

- **Description:** Use the thead element to mark one or more rows of a table, starting from the first row, as heading rows
- **Placement:** thead must be a child of table
- **Content:** 0 or more tr elements.

TBODY Element

- **Description:** Use the tbody element to mark one or more rows of a table, starting after the last thead or tfoot row, as body rows.
- **Placement:** tbody must be a child of table and come after thead (if there is a thead)
- **Content:** Zero or more tr elements.

TFOOT Element

- **Description:** Use the tfoot element to mark one or more rows of a table, starting from the bottom row, as footing rows. Useful for summarizing data
- **Placement:** tfoot must be a child of table, and must be placed *before* the tbody
- **Content:** Zero or more tr elements
- tfoot may be rendered identically to tbody by default; we will see how to change this with CSS

Tr

- Use the tr element to begin a new row in a table
- **Placement:** tr must be a child of table, thead, tfoot or tbody
- **Content:** When parent is thead, zero or more th elements; otherwise, zero or more elements.

TD/TH Elements

- **Description:** Use td/th to add table data to the current row of the table
- **Placement:** td / th must be child of tr
- **Th** for headings, usually in thead
- **Td** for data, usually in tbody

TD/TH Elements

Optional Attributes

- **colspan**: The number of columns this cell will span.
- **rowspan**: The number of rows this cell will span.
- **headers**: A string containing a sequence of html identifiers that are associated with table headers. Used by speech readers to help the blind make sense of table data. Use in td with reference to id attributes in the <th> elements.

TD/TH Elements

```
<table border = "1">
  <thead>
    <tr>
      <th>1</th><th>2</th><th>3</th>
    </tr>
  </thead>
  <tbody>
    <tr>
      <td>1<sub>1</sub></td>
      <td>1<sub>2</sub></td>
      <td>1<sub>3</sub></td>
    </tr>
    <tr>
      <td>2<sub>1</sub></td>
      <td>2<sub>2</sub></td>
      <td>2<sub>3</sub></td>
    </tr>
  </tbody>
</table>
```

TABLE Element (Attributes)

Common Optional Attributes

- **Title:** Specify a tool tip when the mouse hovers over the table. Note that this is an attribute; if you add a title element, it will appear as the page title.
- **Cellpadding**
- **Border**
- **Summary** (used by screen readers)

Caption Element

- **Description:** Use the caption element to define caption text for the table (optional)
- **Placement:** caption must be a child of table
- `<caption>This is a table</caption>`

Caption Element

```
<table>
  <caption>Monthly savings</caption>
  <tr>
    <th>Month</th>
    <th>Savings</th>
  </tr>
  <tr>
    <td>January</td>
    <td>$100</td>
  </tr>
</table>
```


Frames

- Display more than one document in a page
- Very popular for links, ads, and navigation in the late 90's, now regarded as outdated
- No body tags

```
<html>
  <head>
    <title>Example Page</title>
  </head>
  <frameset cols="50%, 50%">
    <frame src="http://www.cheaprx.com">
    <frame src="http://www.scamcentral.com">
    <noframes>
      Text to be displayed in browsers that do not support frames
    </noframes>
  </frameset>
</html>
```

Frames

Frames are out of fashion for many reasons:

- confusing
- interfere with bookmarking
- interfere with back button
- interfere with printing

Frames

- HTML5 standard drops the frame and frameset elements (although they may still display OK)
- One type of frame however, will still exist in HTML5, the inline frame element, iframe

IFRAME Element

- **Description:** Use the iframe to load the contents of another HTML page in a child window within the current document.
- No frameset
- **Content:** Fallback text, in case the web browser does not support inline frames.

IFRAME Element (Attributes)

Optional Attributes

- **src**: The URL of the document to load within the frame. Can be absolute or relative.
- **id**: a name of the frame window (useful when we do JavaScript later on).
- **width**: The desired width of the frame window.
- **height**: The desired height of the frame window.
- **scrolling**: whether to include a scrollbar.
Default is auto

IFRAME Element (Syntax)

```
<head>
<title>Paintings by John Hurley</title>
<link rel = "icon" href = "favicon.ico" type = "image/x-icon" />
<link rel = "shortcut icon" href = "favicon.ico" type =
"image/x-icon"/>
</head>
<body>
<h2>Paintings by John Hurley</h2>
<iframe height = "800px" width = "700px" src="http://www.
artchive.com/artchive/1/leonardo/monalisa.jpg">
    Your web browser does not support inline frames!
</iframe>
</body>
```

Time Element

Time

Tax Day: <time datetime = "2013-04-15">4-15-2013</time>

HTML5 Semantics

- The trend in HTML is towards separating out semantics (meaning and structure) from presentation (formatting)
- The div element has traditionally been used to identify different parts of a document for different formatting
- Over time conventions have developed that provide some information about semantics
 - For example, a div used for navigation links is often called “nav”)
- These are not standardized enough to be very useful

HTML5 Semantics

- HTML5 contains new elements that replace some of the typical divs.
- These can be used to apply formatting with css (see next week's material)
- Since they are standardized, they can be useful to programs that parse your HTML for various purposes
 - Search engines
 - Screen readers
 - Machine Translators
 - Future applications

Example HTML5 Document Structure

`<header>`

`<nav>`

`<article>`

`<section>`

`<aside>`

`<footer>`

<header>

<nav>

<section>

<header>

<article>

<footer>

<aside>

<footer>

HTML5 Semantics

```
<body>
```

```
  <header>...</header>
```

```
  <nav>...</nav>
```

```
  <article>
```

```
    <section> ... </section>
```

```
  </article>
```

```
  <aside>...</aside>
```

```
  <footer>...</footer>
```

```
</body>
```

Metadata

`<meta charset = “utf-8”>`

- Specifies character encoding (code for mapping characters to binary representations)
- Required in HTML5