Document Structure

```html
<!DOCTYPE html>
<!-- This is a comment -->
<html>
  <head>
    <title>Some Title</title>
  </head>
  <body>
    Some Body
  </body>
</html>
```

Heading Tags

```html
<h1>...</h1>
<h2>...</h2>
<h3>...</h3>
<h4>...</h4>
<h5>...</h5>
<h6>...</h6>
```

Paragraph-Level Tags

```html
<p>...</p>  // Paragraph
<pre>...</pre> // Preformatted; white space is significant
<blockquote>...</blockquote> // Long quote; usually indented on left and right
```

```html
<p> attributes:
  align="align"  // align is left, center, or right
  style="margin-left:npx">  // n is the number of pixels to indent
```

Empty Tags

```html
<br> or <br />  // Line break
<hr> or <hr />  // Horizontal rule
```

Physical Character Formatting Tags

```html
<b>...</b>  // Boldface
<i>...</i>  // Italics
<u>...</u>  // Underlined
<strike>...</strike>  // Strikethrough
<big>...</big>  // Large font
<small>...</small>  // Small font
```
Logical Character Formatting Tags

- `<strong>...</strong>`: Strong emphasis; usually boldface; replaces `<b>`
- `<em>...</em>`: Emphasized; usually italics; replaces `<i>`
- `<cite>...</cite>`: Citation; usually italics
- `<code>...</code>`: Computer code; usually monospaced
- `<samp>...</samp>`: Literal characters; replaces `<tt>`

Entity References

- `&lt;`: `<`
- `&gt;`: `>`
- `&quot;`: `"`
- `&amp;`: `&`
- `&nbsp;`: Non-breaking space

Character References

- `&#34;`: `"`
- `&#38;`: `&`
- `&#60;`: `<`
- `&#62;`: `>`
- `&#64;`: The "at" character

Etc., for each ASCII character code

- `&#128;`: The "Euro" character
- `&#149;`: The "small bullet" character
- `&#153;`: The "trade mark" symbol
- `&#160;`: Non-breaking space
- `&#163;`: The "pound" symbol
- `&#169;`: The "copyright" symbol

Lists

Pattern:
`<ul>
  <li>...</li>
  ...
</ul>`

Pattern:
`<ol>
  <li>...</li>
  ...
</ol>`

Tags:
`<ul>...</ul>`: Unordered list; items usually displayed with bullets
`<ol>...</ol>`: Ordered list; items usually displayed with numbers
`<li>...</li>`: List item
Tables

Pattern:
<table>
  <tr>
    <th>...</th>
  </tr>
  <tr>
    <td>...</td>
  </tr>
  ...
</table>

Tags:
<table>...</table>  Table
<tr>...</tr>  Table row
<td>...</td>  Table data
<th>...</th>  Table header; like <td> except bold and centered

<table> attributes:
  border="n"  n is the border width
  cellpadding="n"  n is the amount of space between cell edges and content
  cellspacing="n"  n is the amount of space between cells
  align="align"  align is left, center, or right
  width="n"  n is the width of the table in pixels
  width="n%"  n is the width of the table relative to the page width

<td> attribute:
  colspan="n"  n is the number of columns this cell should span

Uniform Resource Locators (URLs)

protocol://hostname:port/filename?name=value;name=value;...

protocol:  http, https, ftp, mailto, file, ...
hostname:  an IP address
port:  a port number
filename:  A path to a file relative to the application directory on the server

Links and Anchors

<a href="url">...</a>  Page link
<a href="mailto:address">...</a>  Email link
<a name="name">...</a>  Anchor
<a href="#name">...</a>  Link to anchor

Forms

Pattern:
<form>
  <input>
  ...
  <textarea>
  ...
</form>
<select>
  <option>...</option>
  ...
</select>
</form>

<form> attributes:
href="url"
method="method"  method is get or post

<textarea> attributes:
rows="rows"
cols="cols"

<input> attributes:
type="type"  type is radio, checkbox, text, password,
select, hidden, submit, or reset
name="name"  name is the name of the input
value="value"  value is the name of the input

get => browser appends name/value pairs to url; CGI program fetches from QUERY_STRING env var
post => browser places name/value pairs in body of request; CGI program fetches from stdin

Copyright © 2017 by Robert M. Dondero, Jr.