Unless otherwise noted, the content of this course material is licensed under a Creative Commons Attribution 3.0 License. http://creativecommons.org/licenses/by/3.0/.

Copyright © 2009, Charles Severance.

You assume all responsibility for use and potential liability associated with any use of the material. Material contains copyrighted content, used in accordance with U.S. law. Copyright holders of content included in this material should contact open.michigan@umich.edu with any questions, corrections, or clarifications regarding the use of content. The Regents of the University of Michigan do not license the use of third party content posted to this site unless such a license is specifically granted in connection with particular content. Users of content are responsible for their compliance with applicable law. Mention of specific products in this material solely represents the opinion of the speaker and does not represent an endorsement by the University of Michigan. For more information about how to cite these materials visit http://michigan.educommons.net/about/terms-of-use.

Any medical information in this material is intended to inform and educate and is not a tool for self-diagnosis or a replacement for medical evaluation, advice, diagnosis or treatment by a healthcare professional. You should speak to your physician or make an appointment to be seen if you have questions or concerns about this information or your medical condition. Viewer discretion is advised: Material may contain medical images that may be disturbing to some viewers.
HTML
Hypertext Markup Language
Charles Severance
www.dr-chuck.com
HTML

• A way of marking up text to indicate that some text is different than other text

• We “tag” portions of the text to communicate meaning

<p>This is a paragraph with a <strong>loud</strong> word in it.</p>
<p>And this is yet another paragraph.</p>
<li>
<a href="mailto:si539@ctools.umich.edu" title="Send mail to si539@ctools.umich.edu">si539@ctools.umich.edu</a>
</li>
Evolution of HTML
The Web is a Young Technology

• Invented in early 1990’s
• Popular in 1994
• Robert Cailliau - coFounder of the World-Wide-Web

http://www.dr-chuck.com/media.php?id=70
The big picture...

A web server produces HTML which is handed to a browser which needs to lay it out in a blink of an eye and have it pixel perfect as good as a print brochure.

(Screen) Source: http://www.umich.edu
(Server) CC: BY Ketmonkey (flickr) http://creativecommons.org/licenses/by-nd/2.0/deed.en
HTML has evolved a *lot* over the years - as computers and networks have gotten faster.

Source: www.yahoo.com
History of HTML / CSS

• HTML 1.0 - 1993 - The Good Old Days - life was simple
• HTML 2.0 - 1995 - Some interesting layout features - abused
• CSS 1 - 1996
• HTML 3.2 - 1997
• HTML 4.0 - 1997 - Layout moving toward CSS
• CSS Level 2 - 1998
• HTML 4.01 - 1999 - What we use today

HTML has evolved a *lot* over the years - as computers and networks have gotten faster.

http://en.wikipedia.org/wiki/HTML
The Good Old Days

In the good old days you wrote HTML and browsers displayed it - since we wrote it by hand - and modems were slow - it was never too long and never too complex. The browser was never the rate limiting factor. Writing HTML was like using a simple, weak word processor. The tags acted as formatting commands to the browser.
The Ugly Middle Ages

- Web Designers designed to browser capabilities - down to particular minor versions of browsers.

- Extensive testing was needed on lots of browsers

- Designers used tables, nested tables, and chopped up graphics to gain control of the look and feel of web pages to produce a “print-like” experience.

- HTML was UGLY, Hard to develop, and brittle - what looked superb on one browser - often was broken on another browser - even a later release of the same browser.
The Modern Era

- HTML is clean and simple
- There is no presentation in HTML - font, color, spacing, etc etc
- No use of tables except for tabular data
- CSS controls all layout, and look and feel
- Still a bit challenging - but converging
<table>
<thead>
<tr>
<th>Time Period</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1990-1994</td>
<td>HTML was simple and pages looked pretty ugly.</td>
</tr>
<tr>
<td>1995-1999</td>
<td>HTML became more complex and each browser was different.</td>
</tr>
<tr>
<td>2000-2005</td>
<td>Browsers slowly supported CSS to varying levels. HTML was still ugly to support multiple browsers.</td>
</tr>
<tr>
<td>2005-2008</td>
<td>New browsers supported CSS. Old browsers were still pretty pervasive but diminishing.</td>
</tr>
<tr>
<td>2009+</td>
<td>The last “pre-CSS” browser (IE5) is &lt; 0.1%</td>
</tr>
</tbody>
</table>

http://www.w3schools.com/browsers/browsers_stats.asp
What does this mean for us?

• Don’t bother with the intermediate steps - don’t make all the mistakes :)

• Either keep it simple - or do it well - simple does work

• If you want a professional site use all of the best practices

• Presentation in CSS + semantic markup in HTML
HTML
HTML Tag Basics

Tags “mark up” the HTML document. The tags are read and interpreted by the browser - but not shown.

A self closing tag does not need a corresponding end tag.
The Basic Outline

<!DOCTYPE ... >
<html>
<head>
  <!-- Describes and sets up the document -->
</head>
<body>
  <!-- The document to be displayed -->
</body>
</html>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Learning the Google App Engine</title>
</head>
<body>
<h1>Google App Engine: About</h1>
<p>
Welcome to the site dedicated to learning the Google Application Engine. We hope you find www.appenginelearn.com useful.
</p>
</body>
</html>
Welcome to the site dedicated to learning the Google Application Engine.
We hope you find www.appenginelearn.com useful.
Validating HTML

- To validate a web page, you can use the online validator
- Paste, or upload the HTML or use a URL
- http://validator.w3.org

Validation Output: 4 Errors

```
Line 13, Column 7: end tag for "p" omitted, but OMITTAG NO was specified.

You may have neglected to close an element, or perhaps you meant to "self-close" an element, that is, ending it with "</>" instead of ">".
```

Source: W3C http://
<HTML>
<HEAD>
<META NAME="GENERATOR" CONTENT="Adobe PageMill 3.0 Win">
<TITLE>dr-chuck.com</TITLE>
</HEAD>
<BODY BGCOLOR="#000000" LINK="#AAAAAA" VLINK="#AAAAAA" ALINK="#AAAAAA">
<table Border=0>
<tr>...<tr>
</table>
</BODY>
</HTML>

Source: W3C http://validator.w3.org/check
Source: www.dr-chuck.com
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN" "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en">
<head>
<title>University of Michigan</title>
</head>

<title>University of Michigan</title>

Source: W3C http://validator.w3.org/check
Source: www.umich.edu
<h1>Google App Engine: About</h1>
<p>Welcome to the site dedicated to learning the Google Application Engine. We hope you find www.appenginelearn.com useful.</p>

White space and line ends in HTML are ignored - the browser re-flows text based on width and font.
Browser Text Wrapping / Resize

The browser wraps lines based on its width - resizing the browser dynamically re-wraps lines.
Symbols

<body>
<h1>HTML: Special Characters</h1>
<p>
Special characters are indicated by the &amp; character. We can use this to display &lt; and &gt;.
</p>
</body>

<table>
<thead>
<tr>
<th>Entity</th>
<th>Symbol</th>
</tr>
</thead>
<tbody>
<tr>
<td>&gt;</td>
<td>&gt;</td>
</tr>
<tr>
<td>&lt;</td>
<td>&lt;</td>
</tr>
<tr>
<td>&amp;</td>
<td>&amp;</td>
</tr>
<tr>
<td>£</td>
<td>£</td>
</tr>
<tr>
<td>©</td>
<td>©</td>
</tr>
<tr>
<td>™</td>
<td>™</td>
</tr>
</tbody>
</table>
Headers were very ugly in default rendering - most folks started with `<h3>`. 

- First Major Heading
  - First Subheading
  - Second Subheading
    - A Sub-subheading
- Another Major Heading
  - Another Subheading
<body>
  <h1>App Engine: Topics</h1>
  <ul>
    <li>Python Basics</li>
    <li>Python Functions</li>
    <li>Python Objects</li>
    <li>Hello World</li>
    <li>The WebApp Framework</li>
    <li>Using Templates</li>
  </ul>
</body>
<body>
<!-- Make sure to style the h1 -->
<h1>App Engine: Topics</h1>
<ul>
<li>Python Basics</li>
<li>Python Functions</li>
<li>Python Objects</li>
<!-- Leave these two out for a while
<li>Hello World</li>
<li>The WebApp Framework</li>-->
<li>Using Templates</li>
</ul>
</body>
Link Anatomy

Start Tag  
Clickable Text  
End Tag

```
<a href="sites.htm"> Sites </a>
```

Where to go when link is clicked (an attribute).

We will make this pretty later with CSS.
Welcome to the site dedicated to learning the Google Application Engine. We hope you find www.appenginelearn.com useful.
Multiple Files

csev$ ls -l
-rw-r--r-- 1 csev staff 618 Dec 18 22:56 index.htm
-rw-r--r-- 1 csev staff 883 Dec 18 22:57 sites.htm
-rw-r--r-- 1 csev staff 679 Dec 18 22:57 topics.htm
csev$
Special File Names

• When a URL points to a directory in your web server, it looks for a file with a special name:
  
  • index.html, index.htm, index.php, default.htm, etc..

• While there is a convention, the “default file” is configurable - so nothing is “sure”

• Usually index.htm or index.html is a safe bet

• This only works when viewing through a web server - when viewing from disk, you must view the file.
Navigating

<h1><a href="index.htm">AppEngineLearn</a></h1>
<ul>
<li><a href="sites.htm">Sites</a></li>
<li><a href="topics.htm">Topics</a></li>
</ul>
Images

Google App Engine: About

Welcome to the site dedicated to learning the Google Application Engine. We hope you find www.appenginelearn.com useful.
Images

Which image to display

Optional - makes display quicker. Will resize to fit.

Put the image on the right and wrap text around it.

Show this when hovering, images are off, or for screen readers.

All information is communicated through the attributes of the img tag.

```
<img src="appengine.jpg"
    width="142" height="109"
    alt="Google App Engine Logo"
    style="float:right" />
```
Images

<h1>
<img src="appengine.jpg" width="142" height="109"
    alt="Google App Engine Logo"
    style="float:right"/>

Google App Engine: About</h1>

<p>
Welcome to the site dedicated to
learning the Google Application Engine.
We hope you find www.appenginelearn.com useful.
</p>

In this case, the file appengine.jpg needs to be in the same directory as the file index.html.
Google App Engine: About

Welcome to the site dedicated to learning the Google Application Engine. We hope you find www.appenginelearn.com useful.
Welcome to the site dedicated to learning the Google Application Engine. We hope you find www.appenginelearn.com useful.
Summary

• HTML has gone through many changes and evolutions

• It started clean and simple - then got ugly and nasty - now we are back to a clean and simple approach

• HTML Markup needs to focus on meaning - not formatting

• Formatting is handled using CSS - Cascading Style Sheets