

Basic Web Pages with XHTML (and a bit of CSS)

CSE 190 M (Web Programming), Spring 2008
University of Washington

Reading: Chapter 1, sections 1.1 - 1.3

Except where otherwise noted, the contents of this presentation are © Copyright 2008 Marty Stepp and Jessica Miller and are licensed under the Creative Commons Attribution 2.5 License.



Hypertext Markup Language (HTML)

- describes the *content* and structure of information on a web page
 - not the same as the *presentation* (appearance on screen)
- surrounds text content with opening and closing **tags**
- each tag's name is called an **element**
 - syntax: `<element> content </element>`
 - example: `<p>This is a paragraph</p>`
- most whitespace is insignificant in HTML
(it gets ignored or collapsed into a single space)

XHTML

- a newer version of HTML, standardized in 2000
- uses a markup format called **XML** (XML + HTML = XHTML)
- though the browser will accept some malformed HTML, we'll write "strict" XHTML that complies to the official web standards
- why use XHTML and web standards?
 - more rigid and structured language
 - more interoperable across different web browsers
 - more likely that our pages will display correctly in the future
 - can be interchanged with other XML data: SVG (graphics), MathML, MusicML, etc.
- a strict XHTML page uses some different syntax and tags

Structure of an XHTML page

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.1//EN"
"http://www.w3.org/TR/xhtml11/DTD/xhtml11.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
  <head>
    information about the page
  </head>

  <body>
    page contents
  </body>
</html>
```

HTML

- a **header** describes the page and a **body** contains the page's contents
- an HTML page is saved into a file ending with extension `.html`

Page title: <title>

describes the title of the web page

```
<title>CSE 190 M: HTML</title>
```

HTML

- placed within the head of the page
- displayed in the web browser's title bar and when bookmarking the page

Paragraph: <p>

paragraphs of text (block)

```
<p>You're not your job.
You're not how much money you have in the bank.
You're not the car you drive. You're not the contents
of your wallet. You're not your khakis. You're
the all-singing, all-dancing crap of the world.</p>
```

HTML

You're not your job. You're not how much money you have in the bank. You're not the car you drive. You're not the contents of your wallet. You're not your khakis. You're the all-singing, all-dancing crap of the world.

-
- placed within the body of the page
 - more paragraph examples

Headings: <h1>, <h2>, ..., <h6>

headings to separate major areas of the page (block)

```
<h1>University of Whoville</h1>  
<h2>Department of Computer Science</h2>  
<h3>Sponsored by Micro$oft</h3>
```

HTML

University of Whoville

Department of Computer Science

Sponsored by Microsoft

- [More heading examples](#)

Block and inline elements (explanation)

- **block** elements contain an entire large region of text
 - examples: paragraphs, lists, table cells
 - the browser places a margin of whitespace between block elements for separation
- **inline** elements affect a small amount of text and must be nested inside of a block element
 - examples: bold text, code fragments, images
 - the browser allows many inline elements to appear on the same line or within the same block element

Horizontal rule: <hr>

a horizontal line to visually separate sections of a page (block)

```
<p>First paragraph</p>
<hr />
<p>Second paragraph</p>
<hr />
```

HTML

First paragraph

Second paragraph

- should be immediately closed with />

More about HTML tags

- some tags can contain additional information called **attributes**
 - syntax: `<element attribute="value" attribute="value"> content </element>`
 - example: `Next page`
- some tags don't contain content; can be opened and closed in one tag
 - syntax: `<element attribute="value" attribute="value" />`
 - example: ``
 - example: `<hr />`

Links: <a>

links, or "anchors", to other pages (inline)

```
<p>
Search
<a href="http://www.google.com/">Google</a>
now!
</p>
```

HTML

Search Google now!

- uses the href attribute to specify the destination URL
- anchors are inline elements, so they must be placed inside a block element such as a p or h1

More about anchors

```
<p><a href="1-internet.html">Lecture Notes 1</a></p>
<p><a href="http://www.google.com/"
title="Search">Google</a></p>
```

HTML

Lecture Notes 1

Google

- types of URLs that can appear in anchors:
 - **absolute** (to another web site)
 - **relative** (to another page on this web site)
- specify a tooltip with the `title` attribute
- more anchor examples

- to make links that open in new windows, we'll need to learn Javascript (later)

Nesting tags

Bad:

```
<p>
  <a href="1-internet.html">Lecture Notes 1
</p>
<p>
  This text also links to Lecture Notes 1</a>
</p>
```

HTML

-
- tags must be correctly nested
 - (a closing tag must match the most recently opened tag)
 - the browser may render it correctly anyway, but it is invalid XHTML

W3C XHTML Validator

```
<p>
  <a href="http://validator.w3.org/check/referer">
    
  </a>
</p>
```

HTML



- validator.w3.org
- checks your XHTML code to make sure it meets the official strict XHTML specifications
- more picky than the web browser, which may render malformed XHTML correctly

Line break:

forces a line break in the middle of a block element (inline)

```
<p>Teddy said it was a hat,<br />So I put it on.</p>
<p>Now Daddy's sayin',<br />
Where the heck's the toilet plunger gone?</p>
```

HTML

Teddy said it was a hat,
So I put it on.

Now Daddy's sayin',
Where the heck's the toilet plunger gone?

-
- br should be immediately closed with />
 - br should not be used to separate paragraphs or used multiple times in a row to create spacing

Images: ``

inserts a graphical image into the page (inline)

```
<p>  
    
</p>
```

HTML



-
- the `src` attribute specifies the image URL
 - XHTML also requires an `alt` attribute describing the image

More about images

```
<p>  
  <a href="http://theonering.net/">  
      
  </a>  
</p>
```

HTML



-
- if placed inside an `a` anchor, the image will become a link
 - the `title` attribute specifies an optional tooltip

Comments: `<!-- ... -->`

comments to document your HTML file or "comment out" text

```
<!-- My web page, by Suzy Student  
CSE 190 D, Spring 2008 -->  
<p>CSE courses are <!-- NOT --> a lot of fun!</p>
```

HTML

CSE courses are a lot of fun!

Phrase elements : ``, ``

em: emphasized text (usually rendered in italic)

strong: strongly emphasized text (usually rendered in bold)

```
<p>  
HTML is <em>really</em>,  
<strong>REALLY</strong> fun!  
</p>
```

HTML

HTML is *really*, **REALLY** fun!

Introduction to CSS

producing styles with Cascading Style Sheets

The bad way to produce styles

```
<p>  
<font face="Arial">Welcome to Greasy Joe's. You will  
<b>never, <i>ever, <u>EVER</u></i></b> beat  
<font size="+1" color="red">OUR</font> prices!</font>  
</p>
```

HTML

Welcome to Greasy Joe's. You will **never, ever, EVER** beat **OUR** prices!

- tags such as `b`, `i`, `u`, and `font` are discouraged in strict XHTML
 - do not use the above tags on your homework assignments!
 - why??

Cascading Style Sheets (CSS)

- describe the appearance, layout, and presentation of information on a web page (as opposed to HTML, which describes the content of the page)
- describe *how* information is to be displayed, not *what* is being displayed
- can be embedded in HTML document or placed into separate `.css` file
 - advantage of `.css` file: one style sheet can be shared across many pages

Basic CSS rule syntax

```
selector {  
  property: value;  
  property: value;  
  ...  
  property: value;  
}
```

CSS

```
p {  
  font-family: sans-serif;  
  color: red;  
}
```

-
- a CSS file consists of one or more **rules**
 - each rule starts with a **selector** that specifies an HTML element(s) and then applies style **properties** to them

Attaching a CSS file: <link>

```
<head>  
  ...  
  <link rel="stylesheet" type="text/css" href="filename" />  
  ...  
</head>
```

HTML

```
<link rel="stylesheet" type="text/css" href="style.css" />
```

```
<link rel="stylesheet" type="text/css"  
href="http://www.google.com/uds/css/gsearch.css" />
```

-
- a page can link to multiple style sheet files
 - in case of a conflict (two sheets define a style for the same HTML element), the latter sheet's properties will be used

CSS properties for colors

```
p {  
  color: red;  
  background-color: yellow;  
}
```

CSS

This paragraph uses the style above.

- color: color of the element's text
- background-color: color that will appear behind the element

Specifying colors

```
p { color: red; }  
h2 { color: rgb(128, 0, 196); }  
h4 { color: #FF8800; }
```

CSS

This paragraph uses the first style above.

This heading uses the second style above.

This heading uses the third style above.

- color names: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, (white), yellow
- RGB codes: red, green, and blue values from 0 (none) to 255 (full)
- hex codes: RGB values in base-16 from 00 (0, none) to FF (255, full)

Grouping styles

```
p, h1, h2 {  
  color: blue;  
}  
h2 {  
  background-color: yellow;  
}
```

CSS

This paragraph uses the above style.

This heading uses the above style.

- a style can select multiple elements separated by commas
- the individual elements can also have their own styles (like h2 above)