

---

## HTML Introduction

DB0340

(like Fangbum)

---

## Why HTML?

- It is platform independent:
    - Pages can be viewed using a variety of different computers and browsers.
  - It allows convenient linking from one page to another.
  - Html files are small since they store the structure of the document, not its precise appearance.
  - Html is open to everybody. It is not owned by a company.
- 

## HTML tags

- A tag identifies a page element's type, format, and appearance.
  - Unknown tags are ignored.
    - This rule allows new tags to be introduced without causing problems for older browsers. It also means you need to be careful to spell tag names correctly!
  - Tags are enclosed in angle brackets
 

```
<tag-name> Content affected by the tag </tag-name>
```
  - Tags usually come in pairs: an opening tag and a closing tag. Some exceptions:
    - <img> tag has optional </>
    - <br/> tag has no corresponding </br/>
  - Tag names are case-insensitive.
    - Style recommendation: Be consistent.
- 

## What is HTML?

- HTML: Hyper Text Markup Language
    - The predominant markup language for web-page design.
    - A markup language is a set of markup tags.
  - File extensions: .html, .htm
  - Information stored in an html file can be viewed in a browser.
- 

## An HTML document

- HTML document is a plain text (ASCII) file that contain code that defines a web page.
  - We can peek at the code for any page using the "View → Page Source" option in a browser.
  - The contents can be divided into two categories:
    - Content: material which the user sees when visiting the page.
    - Meta-information: Information about the document: its structure, formatting, etc.
  - Meta-information is distinguished from content by tags.
- 

## Nesting of tags

- Opening and closing tags define regions affected by the tags. These regions must nest, not overlap.

Yes:

```
<tag1>Some text <tag2>more text</tag2> and more.</tag1>
```

No:

```
<tag1>Some text <tag2>more text</tag1> and more.</tag2>
```

---

## Structure tags

- There are important one-per-document tags:

**<html> </html>**

Tells the browser that this is an HTML document.  
All other tags are placed in html tag.

**<head> </head>**

Contains information about your page. Everything in the head portion is meta-information, not content.

**<body> </body>**

Contains the content of your page.

## Format: bold, italic, and underline

- Bold <b></b>**
- Italic <i></i>**
- Underline <u></u>**
- You can apply more than one

e.g., `<b><i>Just Do It!</i></b>` produces:  
***Just Do It!***

-tags can be in any order, but have to be nested correctly

## Special tags within HEAD

**<title> </title>** defines the title of the document.

Example: `<title> Title of my web-page </title>`

This tag is optional but should always be included to assist the user in navigating the browser's history list.

**<style> </style>** defines style information for an HTML document. (Will spend more time on this later!)

**<meta>** provides metadata about the HTML document.

Example: `<meta name="author" content="Mike Pangburn" />`

Example: `<meta name="keywords" content="word, phrase, etc" />`

Specifies keywords to assist indexing of the page by search engines

## Heading tags

**<h1>Main heading</h1>** - Formats the enclosed text as a prominent large, bold heading.

**<h2> </h2>** heading style a bit less large than h1

**<h3> </h3>** heading style a bit less large than h2

**<h4> </h4>** heading style a bit less large than h3

**<h5> </h5>** heading style a bit less large than h4

**<h6> </h6>** heading style a bit less large than h5

Examples:

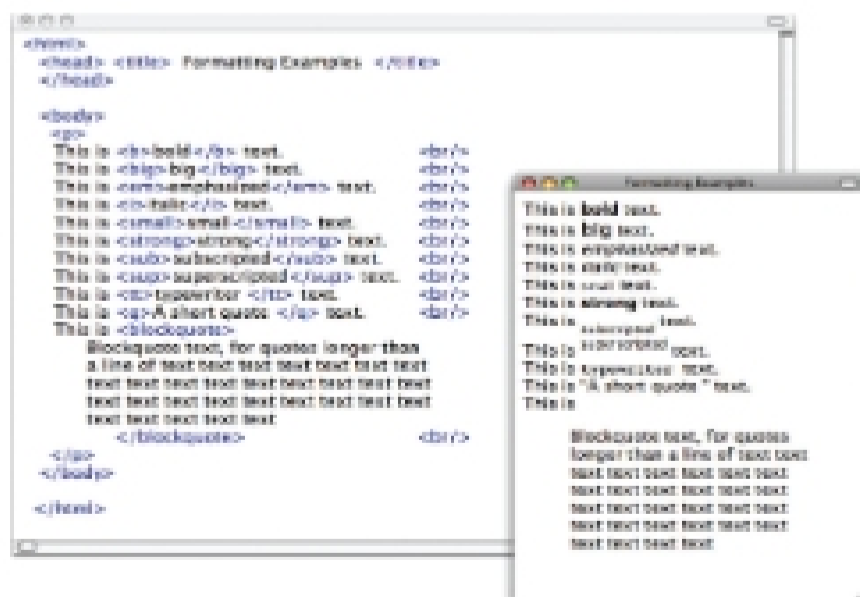
`<h1>This is header 1 </h1>`

**This is header 1**

`<h6>This is header 6 </h6>`

This is header 6

## Text formatting tags



## Other HTML tags

- <hr>** Horizontal rule. Draws a thin solid horizontal line on the web page. No closing tag.
- <br>** Line break. Creates a line break (effectively hitting a "return key"). Use this where the extra line spacing of a paragraph tag is not desirable. No closing tag.
- Comments:** these consist of any text enclosed within `<!-- ... -->`. Their purpose is to enlighten the web programmer reading the HTML. They do not appear in the rendered page.

## The "font" tag

- The `<font>` tag.
  - In the early days of web design, this tag was introduced to allow the web programmer to control the font family, typeface, color, etc.
  - This tag is now considered obsolete. Cascading style sheets (CSS) provide much better control over style and compatibility. We will cover CSS later.
  - However, many web pages still use this tag for simple effects such as text size and color.

```
<font size="+1" color="red">Large, red text</font>
```

## Links with anchor text

Two sides of a hyperlink

- Anchor text

The highlighted text in the current document

- Hyperlink reference

The URL address

- Usage

```
<a href="url">anchor text</a>
```

e.g., Click `<a href=http://www.microsoft.com> here</a>`  
 displays as: Click [here](http://www.microsoft.com)

## URL address

- The Internet **address** portion of a URL can be either a name, e.g. [www.uoregon.edu](http://www.uoregon.edu), or an IP address, e.g. 128.223.142.125
- If omitted, the address of a URL reference within an HTML page is assumed to be the same as the address for the document containing the link.
  - Thus if the address is omitted from a link in a web page, the link refers to a document on the same server that served that page.
- A URL without an address portion can be either absolute or relative, as explained next.

## Special symbols

- What if we want to show a math relationship as:

```
0 <p> r
```

- So the XHTML would be

```
<math>0 &lt; p &gt; r </math>
```

- Special characters

```
&lt; displays as <
&gt; displays as >
&amp; displays as &
&nbsp; non-breaking space
&mdash; displays as -
```

## Links and URLs

- The link target, or href (hypertext reference) is in the form of a URL: Uniform Resource Locator.
- A URL has 3 components, not all of which need to be supplied in every reference:
  - A **protocol**
  - An Internet address (either name or IP number)
  - A file **path**

Example:

```
http://infographics.uoregon.edu/campusMaps/nightmap.pdf
```

## URL file path

- The file **path** portion of a URL optionally specifies the chain of directories (folders) in which the document is located, and the name of the file itself. The directory names in the chain are separated by slash characters.
- If the file name portion of the path is omitted, then it defaults to a value that is defined by the server, typically `index.html`.

Example: the URL

```
http://www.myplace.com/shopping/fruit/
```

lacks a file name, and so it is often equivalent to

```
http://www.myplace.com/shopping/fruit/index.html
```