



HTML BASICS

MARK-UP LANGUAGES

- Traditionally used to provide typesetting information to printers where text should be indented, margin sizes, bold text, special font sizes and styles, etc.
- Word processors like MS Word, and typesetting systems like LaTeX are also forms of mark-up languages.
- Rich Text Format (rtf) files are written in a mark-up text format.

HYPERTEXT MARKUP LANGUAGE (HTML)

- describes the structure of the document
- provides text and other formatting instructions to tell the browser how to render the material.
- has evolved to version 4.01.

www.w3.org

WEB BROWSER

- Software when connected to the Internet is able to access documents at remote locations and display them locally in accordance with its interpretation of markup instructions in the document.
- Examples: Internet Explorer, Chrome, FireFox, Opera, etc.

HTML DOCUMENTS

- Plain text files, with extension `htm` or `html`.
- The extension tells the browser interpret the file according to HTML standards.
- Browsers may display other types of files as well, according to the file's extension.
 - `TXT` - `.txt` - plain text file, no formatting.
 - `PDF` - `.pdf` - portable document format (Adobe corporation)
 - `JPG` - `.jpg` - JPEG image file.

HTML DOCUMENTS

- No special software is needed; you may use any plain text editor to create html files manually.
- Just start by copy-pasting the template codes provided into a new document in the editor program of your choice* and edit/revise the content with your own desired elements.
- Although HTML ignores white spaces, for convenient code viewing and editing, always be sure to use line breaks and indentations logically and sensibly.

WHICH TEXT EDITOR?

○ Windows Users:

- Notepad is all you need but html files must always be saved as "All files" format.
- Notepad++ is one example of a nice alternative with handy features.

○ Mac Users:

- TextEdit is all you need but since it is by default a RTF editor, it must be configured to the plain text mode in Preferences.
- Textwrangler is one example of a nice alternative for novice users who is not so comfortable with file format.

A WORD OF CAUTION:

- No matter which editor you use, if it is not strictly plain text, some characters may cause display errors.
 - One common problem involves illegal quotation mark characters (double and single) such as those generated as **smart quotes** or **inverted** quotation marks (“ ... ”) that are improperly rendered by browsers.
 - Such characters should be replaced by plain **straight quote** (" ... ") characters.

MORE ON HTML EXTENSIONS

- For special applications there are proprietary extensions for the html language although they are NOT official parts of html.
 - Active Server Pages (ASP) files are server-side script engine by Microsoft.
 - Personal Home Page: Hypertext Preprocessor (PHP) which, like asp, generally require the server to have a special handling package installed, for the extension.

CASE RESTRICTIONS

- Technically, HTML is **not case-sensitive** (case-insensitive), meaning that either upper or lower case may be used in tags and the browser will not care.

Example: `<hr>` = `<HR>` = `<hR>`.

- **However, it is strongly recommended that you use lower case only in this course to avoid confusion.**

DOCTYPE

- A pre-processor directive at the top of the document tells the browser (and the validator) what kind of standards apply to the document.

Important: The pre-processor directive at the top, together with the <meta tag specifying the content type in the head are both required in your course assignment document to enable proper code validation.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
...
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

**See their proper application in the "testpage.html" example in the next slide.*

TEMPLATE

**Use the codes below to begin constructing your assignment document.*

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
```

```
<html>
```

```
<head>
```

```
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

```
<title>***replace this text with course & section number and your full name  
</title>
```

```
</head>
```

```
<body>
```

```
<!--Begin your content specification by adding your elements here. -->
```

```
</body>
```

```
</html>
```

BASIC HTML DOCUMENT STRUCTURE

Html document consists of the **head** and the **body**.

- The **head** contents are used by the browser/system.
- The **body** elements are rendered (displayed) by the browser.

```
<html> { Markup Language Type Declaration }
```

```
<head>
```

```
<title> { Descriptive Text Here }</title>
```

```
</head>
```

```
<body>
```

```
    { Main content here }
```

```
</body>
```

```
</html>
```

THE HEAD (HEADER) SECTION

- `<head>` (begin the head section)
- `<title>` (a descriptive title of document)
- `</title>`
- `<! --- >`*Optional items such as script here*`</>`
- `</head>` (end of the head section)

THE BODY SECTION

`<body>`

{ elements of the document to be displayed in the document area go here }

`</body>`

BODY ELEMENTS

- html documents consist of a number of **elements** where text is placed in *containers* with a open/begin tag and an close/end tag (**tandem** tags)

<open tag>some text goes here.**</close tag>**

- Some containers use only one tag – aka **empty** or **stand-alone** tags

<hr> : draws a line across the document

**
** : a line feed, or break

BASIC CONSTRUCTS

- All tags must be **nested** correctly to pass validation.
 - *Rule of thumb: last-opened-first-closed*

Consider these element combination:

`<p> ... </p>` contains a normal paragraph text and

` ... ` contains emphasized/bolder text

Invalid:

`<p>`This overlap is not `` nested. `</p>` So it is invalid. ``

Valid:

`<p>`This is correct ``nesting.`` It is valid.`</p>`

TEXT ELEMENT TAGS

`<p> ... </p>` (Paragraph)

` ... ` (Bold)

`<i> ... </i>` (Italics)

`<u> ... </u>` (Underline)

` ... ` (Emphasis - depends on browser)

`
` (Line feed or break)

`<hr>` (Horizontal Rule/line)

PRESET FORMAT

`<h1> ... </h1>` (Heading level 1: most prominent)

`<h2> ... </h2>`

`<h3> ... </h3>`

...

`<h6> ... </h6>` (Heading level 6: least prominent)

`<blockquote>` (Indents content one tab)

...

`</blockquote>`

LISTS: NUMBERED OR BULLETED

{Begin an Ordered List – numbered}

 ... {1st list element}

 ... {2nd list element}

 {End of Ordered List}

{Begin an Unordered List – bulleted}

...

SIMPLE HTML DOCUMENT EXAMPLE

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN">
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">

<!-- This is a comment line: you may place it anywhere in your document; the browser will ignore it. -->
<title> HTML document example</title>
</head>

<body>
<h3> This be a heading level 3</h3>
<p>This is a sentence in this document.</p>
<p>This is another sentence.</p>
<blockquote> <!-- The list below will be indented. -->
<ol>
<li> First list item</li>
<li> Second list item</li>
<li> Third list item</li>
</ol>
</blockquote>
</body>
</html>
```

THE ANCHOR ELEMENT

- The **anchor** element, the tag name, single letter **a**, is used to connect or link different documents or parts of one document.
- Any text associated with the anchor element, called **anchor text** is displayed on the screen.
- When an anchor text is clicked, a '**jump**' is made to the destination or target location.

HYPertext LINKS

- Anchor tag is used to create a link that jumps from one point in a document to another destination (target location).

Destination may be:

- a. **external** (another web site), or
- b. **local** (a document in the local server), or
- c. **internal** (another section in the current document).

GENERAL FORM:

** anchor text **

- Be Sure To Read: Uniform Resource Locators (URLs) - what they are and how they're structured is explained in [What is a URL?](#)
- As our examples demonstrate, there are many ways to specify the URL attributes in various situations.

EXAMPLE A (EXTERNAL):

```
<a href = "http://www.yorku.ca">York University</a>
```

- Searches for the web address "www.yorku.ca" on the World Wide Web.
- The York site's default page, e.g., index.html, will be displayed when the site is located.

EXAMPLE B (LOCAL):

```
<a href = "faves.html">My Favourite Things </a>
```

Note: a particular file path is not specified here; the browser assumes the file is in the same directory as the calling document (default) and, if found, opens "faves.html" replacing the current page in the browser's document area.

- If the target file, faves.html, is not found, an error message will be displayed.

EXAMPLE C: (WARNING: NOT TRANSPORTABLE)

```
<a href = "file://c:/myweb/resume.html">My Resume</a>
```

- The "file" protocol, tells the browser to search in the **myweb** subdirectory of the C: drive for the resume page - as long as the page is accessed from the same machine, this is not problematic.
- However in the context of the Web access, unless an identical file path exists in the web server (which is virtually never the case), the browser will fail to find the file.
 - Therefore, a web page with such URL specification is not **transportable**, i.e., not useful for publication on the Web.

INTERNAL (NAMED) LINKS:

- Used to create a link that jumps from an origin point in a document to another location within the **same** document.
- The *href* attribute of the anchor tag at the origin point uses a hashtag (#) pre-pended to the destination name.

`Go To Target`

- Another anchor placed at the destination (target) location in the document is marked with a **name** attribute.

` `



Note the absence of the anchor text.

EXAMPLE D: INTERNAL (NAMED) LINKS

`Go To Reference List`

- Displays "Go To Reference List" and causes a jump to the point of the **named location** when clicked.

` `

- Creates a named **label** "target" at the point in the document where it is located.

Note: names are case-sensitive; must be identical in both tags.

IMAGES - EMBEDDED/INLINE

- The **img** (element) is used for displaying image and graphics files.

Attributes:

- **src** = "filename.... ": Identifies the file to be loaded into a document.
- **align** = "top", (or "middle", "bottom", "left", or "right") positions the image in the document

E.G. 1

```

```

- Loads a jpeg* file named "selfie01" into the document at the location of the element.
- JPEG - Joint Photographic Experts Group compression format – file extension, .jpg
- Note: the file must be found in the specified folder one level above current.

E.G. 2

```
<img src = "animals/elephant.gif" align = "right">
```

- Loads a gif* file into the document at the location of the element, and moves it to the right hand side of the page.
- GIF - Graphics Interchange Format compression format - file extension, .gif
- Note: the file must be found in a subfolder of the current folder.

E.G. 3

```
<img src = "/graphics/redballoon.bmp" align =  
"middle" height = "100" width = "200">
```

- Loads a *.bmp file into the document, centers it vertically, relative to the line, and gives it a height of 100 pixels and a width of 200 pixels.
 - BMP- BitMap: simpler non-compressed graphics image format. - file extension, .bmp
 - Changing the height and width values may distort the original proportions of the image.
- Note: the file must be found in the subfolder of the rood directory.

E.G. 4

```
<img src = "winner.jpg" alt = "Sorry, no picture available!">
```

- Displays the image if found in the default folder.
- If the image can not be loaded, the browser should display the text "**Sorry, no picture available!**" in its place.
- This facility was originally intended to accommodate text-only browsers but should always be included in your assignment document.

ANIMATED GIF FILE

- A *.gif image file which is composed of a series of frames or images, simulates motion by displaying the images in the file consecutively.
- It is an electronic equivalent of an older style celluloid motion picture.
- Only use with care.

TABLES

A table is a matrix formed by the intersection of horizontal rows and vertical columns.

	Column 1	Column 2	Column 3
Row 1			
Row 2			
Row 3			

TABLES

- The intersection of a column and row is called a cell.
- Cells in the same row or column are usually logically related in some way.

	Column 1	Column 2	Column 3
Row 1	cell	cell	cell
Row 2	cell	cell	cell
Row 3	cell	cell	cell

TABLES

- A table is defined by the tandem tag container `<table> ... </table>` .
- **HTML expects everything between the start and end elements to be part of the table.**
- Different browsers may render table element attributes differently, but, in general, things like **cell spacing**, type of **border**, **size of cells**, **number of cells** in a row, and **background** and **text colors** may be defined.

`< tr > ... < /tr>`

- Within each row container, each cell is defined by either table header `< th > ... < /th >`, or table data `< td > ... < /td >`

`<th> ... < /th >` is for Table Head

`<td> ... < /td >` is for ordinary Table Data

The Table Header container `< th > ... < /th >`
has exactly the same output as the combination:

`<td><center> ... </center></td>`

Table Format [3 columns x 2 Rows]

<table>

<tr>

<th> Row Header </th> <td> Data </td> <td> Data </td> </tr>

<tr> <td> Data </td> <td> Data </td> <td> Data </td> </tr>

</table>

<th> and <td> Attributes:

colspan = "x" : Forces the cell to cover "x" number of columns. Default = 1.

E.G. `<td colspan = "3" > . . . </td>`

rowspan = "y" : Forces the cell to cover "y" number of rows. Default = 1.

font *whatever* = (as desired) : font attribute such as color, size, family, etc.

valign = "top", "middle", or "bottom" : Sets vertical alignment in the cell.

E.G. `<td valign = "top" > . . . </td>`

align = "left", "center", or "right" : Sets horizontal alignment in the cell.

Other Table Options:

<thead> ... </thead>

- Defines a header section. Contains <tr> elements.
- If used, it MUST precede a

<tbody> ... </tbody>

- which contains the usual table rows, and this should be followed by a footer.

<tfoot> ... </tfoot>

- These three are optional unless the first is used.

More Table Format [3 columns x 2 Rows]

```
<table>
```

```
<thead>
```

```
<tr> <th colspan = "3" align = "center"> Header </th> </tr>
```

```
</thead>
```

```
<tbody>
```

```
<tr> <th> Row Header </th> <td> Data </td> <td> Data </td> </tr>
```

```
<tr> <td> Data </td> <td> Data </td> <td> Data </td> </tr>
```

```
</tbody>
```

```
<tfoot>
```

```
<tr> <th colspan = "3" align = "center"> Footer </th> </tr>
```

```
</tfoot>
```

```
</table>
```

More Example Specifications

```
<table cellpadding = "3" cellspacing = "2"  
width = "85%" border = "0">
```

Define a table with:

- Three pixels between the inside edge of a cell and its contents,
- Two pixels between the outside edges of cells,
- A width of 85% of the browser display,
- No visible border.

Table cells may contain many different ordinary HTML containers, making them quite versatile as aids in layout and design.

Including:

- . Images
- . Hypertext Links
- . Lengthy Text
- . Other tables
- . Also, Applets & Objects

Table-in-a-table example:

```
<table>
```

```
<tr> <td>
```

```
<table>
```

```
<tr> <td> <img src = "somepicture.jpg"></td></tr>
```

```
<tr> <td><a href = "targetURL">Click ME!</td></tr>
```

```
</table>
```

```
</td></tr>
```

```
</table>
```