

# HyperText Markup Language (HTML)

---

Introduction

# What is HyperText Markup Language (HTML)?

- HTML is a format that tells a computer (web browser) how to display a web page.
- The HTML documents are plain text files (ASCII) with special "tags" or codes that a web browser knows how to interpret and display on your screen.
- There are software "tools" that allow you to create web pages without touching any HTML, e.g. dream waver, etc
  - But grounding in the basics will greatly accelerate what you can do.

# Requirements

- Web browser
  - Google chrome, Mozilla Firefox, or others
- Text editor
  - Notepad ++, sublime text or other text editors based on your preference.

# What are HTML tags?

- When a web browser displays a page, it reads from a plain text file
  - and looks for special codes or "tags" that are marked by the **<** and **>** signs.
- The general format for a HTML tag is:
  - `<tag_name>string of text</tag_name>`

# HTML Tags Cont...

- HTML tags may tell a web browser to:
  - bold the text, italicize it, make it into a header, or make it be a hypertext link to another web page.
- It is important to note that the ending tag, `</tag_name>`
  - contains the "/" slash character, tells a web browser to stop tagging the text.
- Many HTML tags are paired this way.
  - If you forget the slash, a web browser will continue the tag for the rest of the text in your document, producing undesirable results
- A web browser does not care if you use upper or lower case.
  - For example, `<h3>...</h3>` is **not different** from `<H3>...</H3>`

# Creating HTML Document

- To create your HTML document, use the simplest text editor available – *NotePad* in windows.
- An HTML document contains two distinct parts, the head and the body.
- The **head** contains information about the document that is not displayed on the screen.
- The **body** then contains everything else that is displayed as part of the web page

- Example:

```
<html>
```

```
<head>
```

```
<!-- header info used to contain extra information  
      about this document, not displayed on the page -->
```

```
<title>Title of the page</title>
```

```
</head>
```

```
<body>
```

```
<!-- all the HTML for display -->
```

```
::
```

```
::
```

```
::
```

```
</body>
```

```
</html>
```

- Enclose all HTML content within **<html>...</html>** tags.
- Inside is first your **<head>...</head>**
  - and then the **<body>...</body>** sections
- Also note the **comment** tags enclosed by **<!-- blah blah blah -->**.

- Here are the steps for creating your first HTML file.
  1. Launch your text editor program.
  2. Go to the text editor window.
  3. Enter the HTML text
    - You do not have to press RETURN at the end of each line; the web browser will word wrap all text
  4. Save the document as “**fileName.html**”
    - *fileName* stands for any legal file name.
    - For old windows versions, use **.htm** extension
      - .htm works in later windows versions too

# Text Formatting

- As a professional who deals with web files, you will consider an HTML file on two fronts:
  - what the browser "sees"
  - what the viewers see
- What the browser "sees" or reads is made of html tags.
- The browser interprets your instructions and displays the result to the best of its interpretation
- Whenever the browser finds text in your file, it tries to know how you want your text to display.
- You are allowed to change a font's style, its color, or its name.

## Some formatting styles, with examples

- **Bold** `<b> </b>` `<b>Project</b>` **Project**
- **Italic** `<i> </i>` `<i>Painting</i>` *Painting*
- **Underline** `<u> </u>` `<u>Planet</u>` Planet
- **Strikeout** `<strike></strike>` `<strike>Peach</strike>`  
Peach
- **Superscript** `<sup></sup>`  $3x^{<sup>2</sup>}$   
 $-x^{<sup>2n+1</sup>}$   $3x^2 - x^{2n+1}$
- **Subscript** `<sub></sub>`  
 $K^{<sub>2</sub>} = a^{<sub>2</sub>} x^{<sub>n-1</sub>}$   
 $K_2 = a_2 x_{n-1}$

- You can use combined styles as in the example:
  - `<i><b>Web Design</b></i>` ***Web Design***
  - `<u><b>Internet Explorer</b></u>` **Internet Explorer**
  - `<b><u><i>Best Seller</i></u></b>` ***Best Seller***
  - $I = I_0 e^{-kd}$
- Use the following general template, e.g.
 

```

<html>
<head>
<title>Web Page Title</title>
</head>
<body>
Your web content runs here
</body>
</html>
```

# Fonts

- A font is a pattern of characters designed to display a character, a word, or a group of words following a particular concept or design.
- To use a font, you include the desired section between the `<font>` and the `</font>` tags.
- The font tag needs to know the name of the font you want to use.
- A font name here is recognized as its "face".
  - Therefore, the `<font face="Font Name">` will specify what font you want to use.
- To close the font tag, which might include many values, use `</font>`

- Examples
  - `<font face="Arial">tourist</font>`
- `<font face="Verdana, Tahoma, Helvetica">entertaining and supporting</font>`
- In this case, the browser will first check whether the visiting computer has the Verdana font, if it doesn't, then the browser will check whether Tahoma is available; if not, "what about Helvetica?"; if none of these fonts is installed, then the browser will use its default.

- You can also control a font's size using one of 7 sizes.
- The font size is controlled using the `<font>` tag.
- Once again, inside of the `<font>` tag, you specify the size using the size attribute:
  - `<font size="Size">`.
- The sizes available in HTML are 1, 2, 3, 4, 5, 6, & 7.
  - 3 is normal size
- For an example `<font size="5">activities</font>`
- The other method for using the **font tag is to provide a relative size change:**
  - `<font size="+1">blah blah blah</font>`
  - `<font size="-2">blah blah blah</font>`
  - i.e. where the **+N or -N value provides the offset from the current font size (default size 3).**

- HTML also allows you to control the color display of a character, a word, or a group of words.
- You use the color attribute of the `<font>` tag to set a font's color:
  - `<font color="Font Color">`
- The other option is to use a combination of letters from a to f and digits from 0 to 9 to specify a color.
  - To use this code, you precede it with #.
  - E.g.
    - `<font color="red">attractive</font>`
    - `<font color="#0000ff">setting</font>`
- All together, e.g.  
`<font size="3" color="blue" face="tahoma">Coolfront  
Island</font>`

- If the whole document is using a particular font color, you can include its attributes in the **<body text>** tag, and this color will impact everything.
- Fortunately, any section that has a particular font color will not be impacted, it will keep its set color.
- E.g. `<body text="#008000">`

# Paragraph Formatting

- Text in your document is usually made of various sections.
- Your job as a Webmaster is to set where a section starts and ends.
- There are various tags used to control the beginning and end the overall display of a section.
- HTML considers as paragraph any group of words that starts somewhere until you as the Webmaster decides where it ends, by breaking it.
- This means a character, a word, a line or a whole file can be considered as one paragraph.

- Everything between the **<p>** and the **</p>** tags is considered as belonging to the same paragraph.
- Wherever a browser sees the paragraph tag, it inserts a blank line and starts a new paragraph.
- To control a section as centered, you use the **<center>** and **</center>** tags
  - Usually used for titles and some section headers
- The **<br>** tag sets the end of a section.
  - This tag doesn't use the beginning and end combinations as the other tags.
  - Simply type it somewhere, then the subsequent words would start on the following line.
- The **<pre>** and **</pre>** tag (preformat tag) instructs your web browser to display the text exactly as typed in the HTML document, including spaces, tabs, and carriage returns

- By default, any paragraph you type is left aligned.
- The `<p>` features an align attribute that is used to change that.
- A paragraph can be aligned to the left, to the center, or to the right or justified.
- The formula to use the tag and its attribute is
- **`<p align="Alignment">`**
  - The Alignment can be: left, center, right, or justify.
- E.g.
  - `<p align="center">Program Presentation</p>`
  - `<p align="right">We feature everything.</p>`
  - `<p align="justify">Player of the season </p>`

- HTML is equipped with **header tags** that are preset formatting features that primarily control a section's text size and alignment.
- There are 6 of these and they are identified as h1, h2, h3, h4, h5, and h6.
- They range from the tall sized h1 to the small h6.
- They are used like the other tags, with an opening and a closing.
- E.g. `<h1>Volcano Web</h1>`
- They can also be used as, e.g.:
  - `<h2 align="center">Program Presentation</h2>`
- The `<h>` tags have a built in break so it is unnecessary to put `<p>` tag before a header tag

# Graphics and Pictures

- Colours, images, pictures, and graphics bring a lot of flavours to a web page.
- Unlike in a word processor, a picture cannot be included on a web file.
  - Instead, you tell the browser where the image is located, the browser will try to find it and then display it.

- If your web pages include graphics, consider the following:
  - Large and numerous images may look great on a high-end computer, but they will consume much bandwidth over the network.
    - As a suggestion, keep the total file size of all images on a web less than 100k
  - Keep graphic images no wider than 480 pixels and no higher than 300 pixels to avoid forcing users to scroll or resize their web browser window.
  - Most importantly, make sure that the images are ones that **add meaning to your HTML** documents.

# Adding Picture on a Page

- There are two primary sources you can use to direct the browser as to where the picture is:
  - the graphics could be on your server or
  - on another server on the Internet.
- Either way you have to clearly provide the address.
- If you will use a picture on your server, you should store it first.
- Make sure the picture is in a format (it has a file extension) that the browser can read (supports).
- Once you have the picture, or to use a picture on your web page, you use the **** combination tags.

# The Image Source

- The first and most important information the browser needs to know about the image is its (physical) location.
- To provide it, you give the source with
  - ``
  - URL means the name of the image you provide, including its extension
- E.g. ``

## The ALT Attribute

- The image tag uses a lot of attributes to give it particular specifications.
- Since the graphics you use usually download relatively slowly depending on the machine used by your visitors, it is recommended to provide **an alternative text** to let the user know what is coming up.
- This alternative text also helps in case your source is not correct or the image is not available.
- This same alternative text pops up when a user places the mouse pointer on the image.
- To provide the alternative text, you use the ALT attribute.
  - E.g. ``

# The ALIGN Attribute

- When you type a text close to the graphic you use, sometimes you want **the text to have a particular relative position** with respect to the image.
- That's when you use the ALIGN attribute.
- You can relatively **align your text** to the **top, middle, bottom, right, or left** of the picture
- Experiment with all of them.
- E.g. ``

## The Image Dimensions

- You can **specify the height and the width of the image**, in pixels.
- Here is an example:

```

```

## The BORDER Attributes

- You can surround your image with a border specified in pixels.
- Here is an example:
- ```

```

➤ The order of the attributes inside the `<img>` tag does not matter

# HTML Lists

- There are two primary styles of lists in HTML: ordered and unordered

## Ordered Lists

- provides a list of items as 1, 2, 3, etc; A, B, C, etc; I,II, III, etc
- start the list with the `<ol>` tag
- Use `<ol type="list_type">`, list\_type being A, 1, a, I, etc
- then provide each item of the list on its own line or paragraph,
  - Each item starts with the `<li>` and ends with the `</li>`
  - It is suggested that each item start on its own line or paragraph for easy readability.  
Properly indented
- At the end of the list, you close it with the `</ol>` tag
- You can create a new list inside of an item of the parent list,
  - start the new one with its own `<ol>` tag.
  - Then list each one of its items inside of its own `<li>` and `</li>` tags.
  - Close any child list with its own `</ol>` tag

## Unordered list

- A list that doesn't present any preset policy
- Initialized with the `<ul>` tag.
  - Then, each line of the list starts with the `<li>` tag
    - closing this tag is optional.
- Close the list with the `</ul>` tag.
- You can also create a new list inside of an item of the parent list
- Use proper indentation for clarity

### E.g. Unordered list

```
<html> <head> <title>HTML List Styles</title>  
  </head>
```

```
<body>
```

Here are a few countries

```
<ul>
```

```
  <li>Cameroon</li>
```

```
  <li>U.S.A.</li>
```

```
    <ul>
```

```
      <li>Maryland</li>
```

```
      <li>Ohio</li>
```

```
      <li>Florida</li>
```

```
    </ul>
```

```
  <li>Bangla Desh</li>
```

```
  <li>Germany</li>
```

```
  <li>El Salvador</li>
```

```
</ul>
```

```
</body>
```

### E.g. Ordered list

```
<html>
```

```
<head>
```

```
<title>Music Collection</title>
```

```
</head>
```

```
<body>
```

Some music albums and their genres:

```
<ol type="1">
```

```
<li>Tizita</li>
```

```
<li>Bati</li>
```

```
<li>Rock</li>
```

```
<li>Salsa</li>
```

```
<li>New Age
```

```
<ol type="A">
```

```
<li>Jean Michel Jarre</li>
```

```
<li>Yanni</li>
```

```
</ol>
```

```
</li>
```

```
<li>indian Jazz</li>
```

```
<li>Soukous</li>
```

```
</ol>
```

```
</body>
```

```
</html>
```

# HTML Links

- The real power of the web is the ability to create hypertext links to related information:
  - Like other web pages, graphics, sounds, digital movies, animations, software programs, etc
- The World Wide Web uses an addressing scheme known as **URLs** (Uniform or Universal Resource Locator) to indicate the location of such items.
- These hypertext links, the ones usually underlined in blue, are known as **anchors**

# Linking to Local Files

- The simplest anchor link is one that opens another HTML file **in the same directory as the presently displayed web page.**
- The HTML format for doing this is:
- `<a href="filename.html">text that responds to link</a>`
  - Think of it as "a" for anchor link and "href" for "hypertext reference"
  - The filename must be another HTML file
- To link to a graphic:  
`<a href="filename.gif">text that responds to link</a>`  
where filename.gif is the name of a GIF image file.

## Links to other directories (Same Computer)

- The anchor tags can also link to an HTML document or graphic file in another directory/folder in relation to the document that contains the anchor
- **To link to lower directory:**
  - Use the "/" character to indicate the change to a subdirectory
- Example:  
<a href="pictures/msh.gif">some text</a>
  - Links to a picture called *msh.gif* in a sub-directory or folder called *pictures* within the directory/folder that contains the working HTML file

- **Anchor Links to a Higher Level Directory**

- You can also construct a link that connects to a higher level directory as well by using:

`<a href="../../page.html">return to home</a>`

- Each instance of "../" the URL of an anchor link tells the web browser **to go to a higher level directory/folder relative to the current page**

e.g.

`<a href="../../D:/academic/cpp/mine/test.html">  
C++</a>`

## Links to the World: Internet Sites

- Uses the tag:

`<a href="URL">Text to Activate Link</a>`

- URL can be a link to another World Wide Web (WWW) server, FTP site, or any text, graphic, sound, video file on these servers.

- E.g.

`<a href="http://solarviews.com/eng/mars.htm">`  
Mars`</a>` has its fair share of volcanic  
landforms and `<a href=`  
“`http://solarviews.com/r/mars/olympus.jpg` “`>`  
Olympus Mons`</a>` also the same.

## Links to Sections of a Page

- To connect to a specific section **within a document**

### The Named Anchor

- A hidden reference marker for a particular section of your HTML file
  - This might be used to link to a different section of the same page if it is long
  - Or to a marked section of another page
- For creating the named anchor:

`<a name="NAME">Text to Link To</a>`

- e.g.

`<h2>Introduction</h2>`

`<h2><a name="intro">Introduction</a> </h2>`

## ***Adding Links to a Named Anchor in the Same Document***

- To create a link to the named anchors in the document
  - Make an item on a page act as a hypertext link to another section of the same document
- E.g. A hyperlink to a section named **"intro"**:  
`<a href="#intro">Introduction</a>`

## ***Adding Links to a Named Anchor in Other Document***

- To create a link that jumps to a section of another HTML document that is marked by a named anchor
- A link to a named anchor in **another local HTML** document is:  
`<a href="file.html#NAME">Text to activate link</a>`
- If the document exists **on another web site**, e.g:  
`<a href="http://www.cheese.org/pub/recipe.html#colby">  
Colby Cheese</a>`

# HTML Tables

- The HTML tags for the basic table structure

## HTML

```
<table border=1>
  <tr>
    <td>Row 1 col 1</td>
    <td>Row 1 col 2</td>
    <td>Row 1 col 3</td>
  </tr>

  <tr>
    <td>Row 2 col 1</td>
    <td>Row 2 col 2</td>
    <td>Row 2 col 3</td>
  </tr>

  <tr>
    <td>Row 3 col 1</td>
    <td>Row 3 col 2</td>
    <td>Row 3 col 3</td>
  </tr>
</table>
```

## Result

|             |             |             |
|-------------|-------------|-------------|
| Row 1 col 1 | Row 1 col 2 | Row 1 col 3 |
| Row 2 col 1 | Row 2 col 2 | Row 2 col 3 |
| Row 3 col 1 | Row 3 col 2 | Row 3 col 3 |

- The `border=1` attribute instructs the browser to draw a line around the table with a thickness of 1 pixel
- Each **row** is defined by **Table Row** tags `<tr>...</tr>`
- And then **cells** in each row are defined by **Table Data** `<td>...</td>` tags.
- Each `<td>...</td>` tag can contain any type of HTML tag we have used so far

## HTML

## Result

```
<table border>
  <tr>
    <td>Row 1 col 1</td>
    <td align=center colspan=2>
      Row 1 col 2-3</td>
  </tr>

  <tr>
    <td>Row 2 col 1</td>
    <td>Row 2 col 2</td>
    <td>Row 2 col 3</td>
  </tr>

  <tr>
    <td>Row 3 col 1</td>
    <td>Row 3 col 2</td>
    <td>Row 3 col 3</td>
  </tr>
</table>
```

\*\* Note the attribute for the second cell of the first row -- it **spans** 2 columns. We have also aligned the text in the center of this cell.

|             |               |             |
|-------------|---------------|-------------|
| Row 1 col 1 | Row 1 col 2-3 |             |
| Row 2 col 1 | Row 2 col 2   | Row 2 col 3 |
| Row 3 col 1 | Row 3 col 2   | Row 3 col 3 |

## HTML

```
<table border=1>
  <tr>
    <td>Row 1 col 1</td>
    <td align=center colspan=2>
      Row 1 col 2-3</td>
    </tr>

    <tr>
      <td>Row 2 col 1</td>
      <td valign=top rowspan=2>
        Row 2 col 2</td>
      <td>Row 2 col 3</td>
    </tr>

    <tr>
      <td>Row 3 col 1</td>
      <td>Row 3 col 3</td>
    </tr>
</table>
```

## Result

|             |               |             |
|-------------|---------------|-------------|
| Row 1 col 1 | Row 1 col 2-3 |             |
| Row 2 col 1 | Row 2 col 2   | Row 2 col 3 |
| Row 3 col 1 |               | Row 3 col 3 |

# Invisible Tables

- A table with borders is useful for charts and data but other times we want to create a grid-like layout that does not have the borders
- Invisible tables appear to readers that they are not looking at a table
- Invisible table is built in the same manner as the table we built above except the `<table>` tag looks like:

`<table border=0>`

# Colouring Tables

- It is as simple as inserting bgcolor=#FF0000 (for e.g.) or the color name as an attribute to the different <table> tags:

| Table Part                                             | HTML                                       |
|--------------------------------------------------------|--------------------------------------------|
| <b>table</b><br>color the area behind the entire table | <code>&lt;table bgcolor=#880000&gt;</code> |
| <b>row</b><br>color the area behind a single row       | <code>&lt;tr bgcolor=#880000&gt;</code>    |
| <b>cell</b><br>color the area behind a single cell     | <code>&lt;td bgcolor=#880000&gt;</code>    |

E.g.

```
<html>
<head>
<title>A Table</title>
</head>
<body>
<table bgcolor=purple>
  <tr bgcolor=yellow>
    <td>Row 1 col 1</td> <td align=center colspan=2>
    Row 1 col 2-3</td>
  </tr>
  <tr>
    <td>Row 2 col 1</td> <td>Row 2 col 2</td> <td>Row 2
    col 3</td>
  </tr>
  <tr bgcolor=orange>
    <td>Row 3 col 1</td> <td>Row 3 col 2</td> <td>Row 3
    col 3</td>
  </tr>
</table>
```

# Frames

- Frames are used to divide your web screen in different parts that can each behave independently
  - For example, one part of the screen could display and change data regardless of what the other screen(s) is (are) displaying.
  - You can also make different frames depend on each other.

Self exercise:

*Create pages that use frames.*