Publishing on the web

1

INTRODUCTION TO HTML

Outline

- In this lecture we will see:
  - What HTML is
  - What are Tags
  - Standard HTML form
  - How you can write a simple HTML document
  - Headers, paragraphs, styles
  - Lists
  - Pre-formated text

What is HTML?

- Hyper Text Mark up Language
- It is not a programming language.
- Mark up Language: we use marking up of regular text for emphasis and organisation.
- That's why people who work with HTML are called "designers" or "developers" and not programmers.
- The HTML standard is overseen by a few different organisations.
- It is also influenced by what commands and features the Web designers use, and what they choose to ignore.

HTML framework

- It is important to realize that an HTML document must be built on a very specific framework.
- This framework, in its simplest form, consists of three sets of matched container tags.

- A tag is a simple mark-up element, and always takes the form <tag>.

- A container is a pair of HTML tags of the form <tag> </tag>.
  - You can think of the <tag> element as turning something on, while the </tag> turns that same thing off.
  - Surrounding text with these tags is like giving a specific command to the reader of HTML documents, the Web browser.

A brief history of HTML

- Version 1, developed in 1991
- HTML+ developed in 1993
- HTML 2.0 developed in 1995
- HTML 3.2 developed in 1997
- HTML 4.01 developed in 1999
- XHTML developed in 2000
- HTML 5 developed in 2012

- HTML 5 is the latest, cleaner form of HTML. To have full effect however, it needs to co-operate with other technologies like CSS (which we will also see at later lectures).

What are HTML Tags Anyway?

- When a Web browser displays a page on the Web, it reads from a text file, and looks for special codes or "tags" that are marked by the < and > signs. There are start tags <> and end tags </>.
- For example:

  `<h1>This is a header tag</h1>`
- Tells the browser to display the enclosed text using the style header level

More on Tags

- There are two types of HTML tags:
  - an empty tag, and non-empty tag.

- Empty tags such as <br>, can occur at any location in the document. Empty tags do not require an ending tag.

- Non-empty tags such as <a> and , surround some text. Non-empty tags require both a starting <a> and ending tag .

- The closing order of tags must be in opposite order of the starting tags. So, <a>... is correct coding, but <a>... would cause a syntax error.

- HTML tags can tell a browser to bold the text, italicize it, make it into a header, make it a hypertext link to another Web page, and more. We will explore these areas later on in the lessons.

Some Important Points to Note

- The ending tag </tag name>, contains the backslash (/) character. This"/" character tells a Web browser to stop tagging the text. All HTML ending tags use this character.
  - If you forget the "/", the Web browser will continue to tag for the rest of the remaining text in your document. For example, if you entered the starting tag for bolding but forgot to use the ending tag , the text you wanted bold as well as all text that follows, will appear bold.

- HTML is case-insensitive, that is: <BR> means the same as <br> as <Br> as <bR>. It is up to you to decide whether to use uppercase or lowercase tags in your Web documents. Lowercase is suggested.

- Empty spaces and return between tags do not matter.

Let's Start!

The standardized document structure used for all HTML files has three tags:

html: Defines the start of the html tags. All html content must be contained within this element

head: information about the document that the browser will not show, but needs to know (for a search engine for example.

body: what the browser will show
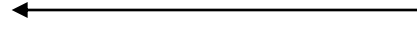
Ignore the doctype for the moment

```
<!DOCTYPE html>
<html>
    <head>

    </head>
    <body>

    </body>
</html>
```

Content related information goes here

Actual text to appear in the web-page goes here

Structure

- 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. An example being <title>text</title> which displays specified text in the title bar of your browser.
- The body contains everything else that is displayed.
- Enclose all HTML work within the <html> and </html> tags. You may notice that your Web pages will display fine without these tags.
- However by using them, your Web pages will be fully compliant with International HTML standards and ensure compatibility with future Web browsers.

A word of caution

- You may find out that even if you do not follow the rules about the tags to the letter, IE displays your document correctly.
- CAUTION: This does not mean that your page is correct.
- The same page may not be correctly displayed on other browsers.
- The only way to insure that your web-page will be displayed correctly on all browsers, is to follow the rules.

HTML Example

You can try these examples on your browser!

Open brackets,

Copy and paste the example at a new document,

Select "save as" and give for file name "example.html"

Before you press the "save" button select "save as type" the value "HTML (*.htm*, *.stm*)"

Press the "save" button

From the toolbar press the icon with the thunder

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Our first web page</title>
    </head>
    <body>
    In this lesson we are going to start our journey
    of becoming wonderful web authors of wonderful
    home pages!
    </body>
</html>
```

Things to keep in mind when writing HTML

- Use a simple text editor (TextPad, notepad, write).

- If you are using a word processing program like Word to create your HTML documents, be sure to save the documents using the ASCII text format.
    - Using Word, and applying bold to some text, will not work.

- Always name your files with the extension .html. By using this file name extension, a Web browser will know to read these text files as HTML and properly display the Web page.

Modifying Our First HTML Document

Try adding the following text in the body element, of the previous example.

Save the file and reload it on the browser what will be displayed?

```
<h1>What is a Markup Language?</h1>
A Markup Language is nothing more than a
collection of styles (indicated by markup tags)
that define the various components of a
hypertext document. The Web browser reads and
interprets the <i>tags</i> and displays the
document accordingly.

<h2>The Hypertext Markup Language-HTML</h2>
HTML is the coding language used to create
Hypertext documents for use on the World Wide
Web. HTML looks a lot like old-fashioned
typesetting code, where you surround a block of
text with codes that indicate how it should
appear. In HTML you can specify that a block of
text, or a word, be <i>linked</i> to another
file on the Internet.
```

Headers

HTML provides six levels of headers for Web documents.

Different Web browsers display these headers differently. Headers will begin and end on a new line.

Headers appear in six levels, <h1> *being the largest*, <h6> *being the smallest*:

```html
<!DOCTYPE html>
<html>
    <head>
        <title>Our first web page</title>
    </head>
    <body>
     <h1> A top level header is displayed in a large size </h1>
     <h2> A second level header is not as large </h2>
     <h3> A third level header uses a smaller font </h3>
     <h4> A fourth level header uses even a smaller font </h4>
     <h5> A fifth level header is smaller yet </h5>
     <h6> A sixth level header uses the smallest font </h6>
    </body>
</html>
```

Use of headers

- Use headings regularly and consistently to help add structure and provide guideposts to your documents.
- Generally, <h1>....</h1> and <h2>....</h2> are used for titles.
- <h3>....</h3> can be used for boldfacing but remember this will add a blank line after the contents of your level 3 heading.

Paragraphs

Whenever a browser encounters the paragraph tag, "<p>", it *inserts a blank line then begins a new line.*

Note that "<p>" *is a non empty tag* and does require an ending tag (</p>).

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
      <p>We are learning how to insert a new
      paragraph with the paragraph tag.</p>
      <p>This is a second paragraph with the
      appropriate tag.</p>
   </body>
</html>
```

Text breaks

Whenever a Web browser encounters the text break tag, "<br>", it begins a new line but does not insert a blank line like the "<p>" tag.

Note that "<br>" is an empty tag and does not require an ending tag (</br>).

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
      <h2>Entering Text Breaks</h2>
      This is how you create <br> text
      breaks which immediately start on the
      <br> next line
   </body>
</html>
```

Horizontal lines

Whenever a Web browser encounters the horizontal rule tag, "<hr>", it inserts a horizontal rule in the document.

You may have noticed that you have entered this HTML tag in previous cases.

Note that "<hr>" is an empty tag and does not require an ending tag (</hr>).

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
      <hr>
      <h2>Entering Horizontal Lines</h2>
      <hr>
      This code enters three horizontal
      rules on your document:<hr><hr><hr>
   </body>
</html>
```

Lists: bulleted and numbered

- HTML supports both *bulleted* and *numbered* lists. You can create the traditional unordered (bulleted) list or an ordered (numbered) list by simply using HTML List tags. List tags can also be used to *create nested lists.*
- The following list tags are non-empty tags:
- The <ul> tag indicates the beginning of an *unordered, bulleted list.* The tag indicates the end of an unordered, bulleted list. The <ol> tag indicates the beginning of an *ordered, numbered list.* The tag indicates the end of an ordered, numbered list.
- The <li>.. tag is a non empty tag and *specifies a list item.*

List example

Using the ul and li tags

Changing the ul to ol will create a numbered list

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
      <hr>
      <h2>Unordered Bulleted Lists</h2>
      <hr>
      <ul>
         <li>This code specifies the</li>
         <li>beginning of a</li>
         <li>list item in an</li>
         <li>indented, bulleted list</li>
      </ul>
   </body>
</html>
```

Nested Bulleted List

A list may contain another list!

Notice there is no difference between the li tags of the two different lists

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
      <p>
      <hr>
      <h2>Nested Lists</h2>
      <hr>
      <ul>
         <li>You can nest bulleted lists </li>
            <ul>
               <li>to create </li>
               <li>multilevel </li>
               <li>bulleted </li>
            </ul>
         <li>lists </li>
      </ul>
      </p>
   </body>
</html>
```

The Comment Tag

- There's one other tag, called the comment tag. This tag is fairly unique, in that it's *actually used to make the Web browser ignore anything the tag contains.*
- That can be text, hypertext links, image links, even small scripts and programs.
- Although it's best to delete obsolete links and tags from your documents, rather than just using the comment tag.
- Some browsers will display certain tags even if they are "commented out"!
- For now, you'll use the comment tag to hide text.
- The point in hiding the text is that it allows you to create a private message that is intended to remind you of something or to help those who view the raw HTML document to understand what you're doing (for example your instructor).

```
<!--This is a comment that won't display in a browser-->
```

Using the comment tag

- To mark a point in a particular HTML document where you need to remember to update some text.
- To explain a particularly confusing part of your page.
- Since it's fairly easy for anyone to view your raw HTML document, you might also use the comment tag to create a copyright message or give information about yourself. For instance:

```
<!--Contents of this document Copyright 2006. Please do not copy or otherwise reproduce the source HTML code of this document without permission.-->
```

- It's not really a container tag, since it doesn't have two similar tags that are differentiated only by / in the second tag. At the same time, it's difficult to describe as an empty tag, since it does do something to text in the document.

Pre-formatted Text

- The <pre> (preformatted text) tag is designed to allow you to keep the exact spacing and returns that you've put between the start and end tags.
- The basic reasoning behind this tag is the notion that every once in a while you'd like your text to stay exactly as you put it -for instance, in a mathematical formula, or if you create a table.
- While there are other ways to do both tables and math, they don't fall under the HTML standard. On top of that, you can use <pre> for a number of other reasons: code lists, lining up decimals for dollar figures, and even poetry.

Example:

lets try this first

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
   <p>Oh, pilot of the storm who leaves no trace
   Like thoughts inside a dream
   Heed the path that led me to that place
   Yellow desert stream
   My Shangri-La beneath the summer moon
   Will return again
   Sure as the dust that floats b'hind you
   When movin' through Kashmir</p>
   </body>
</html>
```

Example:

And now this

```
<!DOCTYPE html>
<html>
   <head>
      <title>Our first web page</title>
   </head>
   <body>
   <pre>Oh, pilot of the storm who leaves no trace
   Like thoughts inside a dream
   Heed the path that led me to that place
   Yellow desert stream
   My Shangri-La beneath the summer moon
   Will return again
   Sure as the dust that floats b'hind you
   When movin' through Kashmir</pre>
   </body>
</html>
```

- Notice that the preformatted text is in a monospaced font-it will always be that way. Otherwise, the <pre> tag works pretty much like the paragraph font, except that it lets you decide where the line breaks and spaces will appear. Look at the following example:

```
<pre>I simply want to make this
<b>really</b> clear to you.

</pre>
```

Potential drawbacks

- There is one potential drawback to the <pre> tag. It doesn't allow the browser screen to wrap text automatically.
- Instead, users need to expand their browser window if you use particular long lines within a <pre> container.
- Just keep this in mind, and make sure your lines of text are reasonably short so that all browsers can view them without scrolling.