

# Unit 2, cont.

September 12

More HTML

# Attributes

- Some tags are modifiable with *attributes*
- This changes the way a tag behaves
- Modifying a tag requires two parts:
  - The *attributes* you want to modify
  - The *value* of the attributes
- `<tag attribute = “value”>`
- Always goes into the opening tag
- Quotes around the value

# Horizontal Rule Example

- `<hr>` is the horizontal rule tag
- Normally goes all the way across the screen
- Say, we want it to go halfway across:
  - `<hr width = "50%" />`
- `<hr>` is an empty tag so we can immediately close it

# Multiple Attributes

- Tags can have more than one attribute
- Format for those tags are:  
`<tag attribute1 = "value1" attribute2 = "value2">`
- The order of the attributes does not matter
- Some attributes are listed in a certain order by matter of convention
  - Follow convention when possible
  - Inserting images is an example

# Multiple Horizontal Rule Attributes

- What if we want to change the thickness and the width of the horizontal rule?

```
<hr width = "75%" size = "20" />
```

- Now lets also make it solid:

```
<hr width = "75%" size = "20" noshade = "noshade"/>
```

# Attribute Example with Text

- To change the font, use the *style* attribute
- If we wanted a paragraph with red text:
  - `<p style = "color:red">`
- What about changing the font size?
  - `<p style = "font-size:150%">`
- And if we want to change both?
  - `<p style = "font-size:150%; color:red">`
- Don't forget the semi-colon!

# Entities

- Special characters can't just be typed into an HTML document
- For instance the “<”, or less than sign is the same as the opening brace for a tag  
<>
- Browser will assume that what comes after the “<” is the tag!
- To print special characters like this, use *entities*

# Entities

- All entities start with an ampersand and end with a semi-colon
- Entities look something like:
  - &lt; which produces the < or less than symbol
- Some examples:
  - &gt; “>”, greater than
  - &amp; “&”, or ampersand
  - &quot; “ “ ” quote marks
- Not all browsers can display all entities....use sparingly



# Comments

- Comments are a way to include text in the document that you don't want displayed
- Why?
  - To place notes for yourself, or others
  - To “comment out” HTML so that it is not displayed
- Comments begin with `<!--` and end with `-->`
- `<!--` Comments can be useful when reading HTML `-->`

# Hyperlinks

- Hyperlinks are what connect one web page to another
- Shortened to the term “links”
- A webpage with no link to it cannot be found by a web crawler
- Make sure all your pages are linked together (properly) or we won't find it either

# Links, cont.

- Links are usually displayed differently than normal text in a web page
- Often it is a different color and underlined
  - But not always
  - Often change color once the user has visited that link as well
  - Should be able to tell it's a link

# How to Create a link

- The `<a>` tag is used to create a link
- The `href` attribute is the URL of the destination of the link
- The contents of the `<a>` tag are the text of the link
  - Or could even be a picture
- `<a href = "URL">content</a>`

# Simple Link Example

This is what a `<a href="http://www.sfu.ca">link</a>` looks like

This is what a [link](http://www.sfu.ca) looks like

# More Complicated

This is a `<a href =“http://www.cs.sfu.ca”><em>more interesting</em></a>` link

- This is a [more interesting](http://www.cs.sfu.ca) link

# Absolute URLs

- Absolute URLs are what the previous examples are
- They specify the full address of a web page
- Includes the scheme, server, and path
- Used mostly for off-site links
  - Links that go away from the current web site

# Why Use Relative URLs?

- Imagine for every page on a website having to put in the entire web address
  - <http://www.cs.sfu.ca/CC/165/sbrown1/lectures.html>
  - [http://www.cs.sfu.ca/CC/165/sbrown1/courseSoftware.](http://www.cs.sfu.ca/CC/165/sbrown1/courseSoftware)
  - <http://www.cs.sfu.ca/CC/165/sbrown1/references.html>
- First, its tedious to keep having to type the full address
- If the page moves to a new server, every single URL would have to be fixed



# What are Relative URLs

- Allow us to link to pages *relative* to the current page
- Much shorter to type!
- Allows websites to be moved from one location to another
  - Can build entire site on home computer then upload to the server

# How Relative URLs Work

- Everything with relative URLs is based on directories
- If the destination page is in the same directory as the current page  
URL is just the “destinationpage.html”
- Example:  
`<a href = “somepage.html”> some text</a>`

# How Relative URLs Work

- If the destination page is down a directory
  - URL:  
“destinationDirectory/destinationPage.html”
  - For each sub directory add its directory/  
before the page
- Example:  
<a href = “lectures/sept12.html”>September 12</a>

# Relative URLs, cont.

- If the destination page is in a directory above the current directory
  - For each directory “up” add a ../
  - URL: “ ../destinationPage.html”
- “up” and “down” can be combined:
  - URL: “../../directory1/directory2/mypage.html”

# Relative URL Example

- In class example

# Images

- Insert images using the `<img>` tag
- `<img>` has 2 required attributes
  - `src` : indicates the URL of the image
  - `alt` : specifies *alternative* text for the image
- `<img>` is an empty tag
- Why text for an image?
  - Image cannot be loaded
  - Image hasn't been downloaded yet
  - Browser doesn't support images

# alt Attribute

- Truly important for the visually impaired
- Should attempt to describe the image as much as possible, without going overboard
  - i.e. no paragraphs
- <http://ppewww.ph.gla.ac.uk/~flavell/alt/alt-text.html>  
(available from useful links page)
  - Talks about using alt properly
- Should try to point out the feature of the image that is relevant
- Try to make it readable inline with the rest of the text on page
- The alternative text in your assignments will be part of your marks

# Simple Image Example

- Lets say we want to insert the image sfuAQ.jpg and its in the same directory as the page we want to put it on

```
<img src = "sfuAQ.jpg" alt = "Simon Fraser  
University's Academic Quadrant at sunset" />
```



# Image in Another Directory

- Images are specified by their URL
- Just like a webpage, you link them relative to the page (when possible)
- Lets say that the sfuAQ.jpg is stored in a folder called “pics”

```
<img src = “pics/sfuAQ.jpg” alt = “Simon  
Fraser University’s Academic Quadrant at  
sunset” />
```

# Specifying Image Size

- Good idea to specify image size
  - Allows the page to be loaded more correctly before images are downloaded
  - Browser already knows how big of a space to leave for the image
- Specify the size in pixels
  - Use *height* and *width* attributes
- Browser inserts the image like a character
- You should know how to specify the image size

# Specifying Image Size Example

- Lets insert sfuAQ.jpg again with a size of 240 x 120 pixels

```
<img src = "sfuAQ.jpg" alt = "Simon Fraser  
University's Academic Quadrant at sunset"  
width = "240" height = "120" />
```

# So many tags, rules, etc.

- There are quite a few tags, attributes, entities, etc.
- Where can you find information on them?
- Several links from the course webpage
  - 1. XHTML 1.0 Reference (found in reference section)
  - 2. Useful links has other HTML sites, including tutorials

# XHTML 1.0 Reference

- **Syntax:** general usage of the tag
  - Empty tags are shown with the short-form closing tag, e.g. `< br />`
- **Attribute Specification:** attributes for the tag and their possible values
- **Contents:** what can be put inside the tag
  - `<li>` tags can be put inside `<ul>` tags
- **Contained in:** what can the tags go into
  - `<ul>` tag can go into a `<p>` tag, `<blockquote>`, etc.
- Remaining text describes what the tag is for, how it should be used, and what attributes it has

# Deprecated Tags and Attributes

- In the XHTML 1.0 reference, some text is lighter colored
- This indicates that the tag or attribute is deprecated
- Deprecated is used to describe features which are obsolete and are being phased out
- In the future, you will not necessarily need to use these deprecated tags and attributes, but for assignment 1, it is okay.

Questions?