CS7026: Authoring for Digital Media

HTML Authoring
Before you can link to another page (or resource), you have to know its URL (uniform resource locator).

A URL is used to specify addresses on the World Wide Web. It is the fundamental network identification for any resource connected to the web (e.g., hypertext pages, images, video and sound files).

You can find the URL for any resource in the Address box (or Location box) of your browser.
Linking Text and Documents

- URLs have the following format:

  protocol://hostname/other_information

- For example, the URL for TCD's home page is:

  http://www.tcd.ie/

- The protocol specifies how information from the link is transferred. The protocol used for web resources is HyperText Transfer Protocol (HTTP).
Linking Text and Documents

- The protocol is followed by a colon, two slashes, and then the domain name.

- The domain name is the computer on which the resource is located.

- Links to particular files or subdirectories may be further specified after the domain name. Directory names are separated by single forward slashes.
Hyperlinks

The `<a>` tag (called an anchor) is used to define hyperlinks.

The `<a>` tag requires an attribute. You must specify whether you want the enclosed text to:

- **link to** someplace (with the `<a href="the URL you want to link to">` tag).
- **be linked to from** someplace (with the `<a name="the place in a document you want to link to">` tag).
Hyperlink to another location

The `<a>` tag with the `href` attribute and attribute value (the URL you are linking to), and its closing tag, `</a>`, surround any text that you want to highlight.

For example, here you would click on the words ‘click here’ to jump (hyperlink) to the document found at the URL inside the quotes:

Please `<a href = “http://www.google.com”> click here </a> to search.`
You can use the same `href` attribute to link to email addresses, to other pages of your own Web site, or even to other files on your own computer.

This hyperlink points to another file (second.htm) on your own computer. The second.html file is stored in the same directory as the page linking to it:

Please `<a href="second.html">click here</a>` to open my second web page.
Linking to Other Files

- If the second.html file was stored in another directory (e.g. The links directory) **below** the directory that your first page is stored in the hyperlink would need to include the directory name too:

Please `<a href="links/second.htm">click here</a>` to open my second Web page.
If we now want to link from the second.html back to the original file (which is stored in the directory \textit{above} it) we would need to include the directory name \textit{and} an indication that it is \textit{above} it:

Please \texttt{<a href="../links.html">click here</a>} to open my second Web page.

Similarly if it is two directories above it:

Please \texttt{<a href="../../second.htm">click here</a>} to open my second Web page.

What happens if the directories are on the same level??
Linking to Email

- You can link to an email address by using the `mailto` prefix.

- When you click on the words click here, an email window appears that enables you to type in your message to Mickey Mouse will appear:

  `<a href="mailto:mickey.mouse@disney.com"> click here to send mail to Mickey.</a>`
Hyperlinks can be used to point to a specific spot (anchor) in a HTML document, instead of just pointing to an entire document.

Remember anchor tags come with two attributes: \texttt{href} and \texttt{name}.

The \texttt{name} attribute allows you to name an anchor in your document.

You can then link to that anchor using the \texttt{href} attribute.
Linking Text and Documents

- Linking Within the Same Page (cont.)

```html
<html>
<head>
<title>Linking</title>
</head>

<body>
<h1>Click on a headline to view the entire article</h1>
<a href="#article1">The First Article</a>
<a href="#article2">The Second Article</a>
<a href="#article3">The Third Article</a>

<a name='article1'><h2>The First Article</h2></a>
<p>Content of article one...</p>

<a name='article2'><h2>The Second Article</h2></a>
<p>Content of article two...</p>

</body>
</html>
```
**Linking Within the Same Page**

- Notice how the `href` attribute points to the location named by the name attribute.

- The `<a href>` tag includes the same format you’ve seen before, but also includes the `#` symbol to separate the filename from the named anchor.

- The `<a>` tag with the `href` attribute is highlighted, but the `<a>` tag with the name attribute is not.

- When naming anchors, keep the names short and don’t use spaces (these are not HTML requirements they just make your life easier...).
Linking to an Anchor in Another Page

- One more element needed - the addition of the URL.
- You can link to
  - an anchor on a file in your own Web site.
  - a *known* anchor in a file on another Web site.

```html
<html>
<head>
<title>internal linking</title>
<style type = "text/css">
</style>
</head>

<body>
<h1>Click a Headline to Read the Full Text</h1>
<a href="articles/firststory.htm#headline1">(i) The First Headline</a>
<p>
<a href="articles/secondstory.htm#headline2">(ii) The Second Headline</a>
</p>
</body>
</html>
```
HTML Comments

- By the way...<!-- this is how you comment your code in HTML -->

- Comments are added with the purpose of making the source code easier for humans to understand, and are generally ignored by compilers and interpreters.

- Comments are used to describe and explain complicated code to someone who may have to work on it in the future (including themselves!)
HTML Comments

- They can also provide details about when the code was edited or reviewed. This can be helpful information and act somewhat like a development log of when a site was last worked on.

- Use them to chat with other developers working on the code. This would likely be done during development as a way to share details with others working on the code at the same time. (If these comments are purely development conversations, it is a best practice to remove them from the code before a site is launched.)
Comments for Debugging

- You can use comments to remove sections of the code, but leave them available for later use.

- This is often done during site testing and debugging. If you have a problem on a site and are not sure what is causing it, you can comment out sections one at a time to quarantine and identify the issue, and then work on a fix for whatever problems you are having.
Creating Lists

- Bulleted (Unordered) Lists
  - A bullet (usually a solid circle) appears in front of each item.
  - HTML automatically creates the bullet when you use the unordered list tag, `<ul>`, together with the list item tag, `<li>`.

```html
<ul>
  <li>first item in the list</li>
  <li>second item in the list</li>
  <li>third item in the list</li>
</ul>
```
Creating Lists

- Numbered (Ordered) Lists
- HTML automatically numbers or letters your items for you.
- Exactly the same as bulleted lists except your use `<ol>` instead of `<ul>`:

```html
<ol>
  <li>first item in the list</li>
  <li>second item in the list</li>
  <li>third item in the list</li>
</ol>
```
Creating Lists

- Numbered (Ordered) Lists

  Like the `<ul>` tag, HTML has a default style for the list items within the `<ol>` tag - it will number the items: 1, 2, 3, 4...

  It will automatically renumber the list items if you add/delete items later.
Creating Lists

- Ordered Lists - Setting a Start Value
  - If you wish to start your list with a value other than 1
  - You may need to interrupt a numbered list with some other material
    - Close the first list.
    - Add the additional materials.
    - Start a new list using the list items `value` attribute to set the beginning number for the new list.
Creating Lists

- Ordered Lists - Setting a Start Value

```html
<html>
<head>
<title>Formatting Ordered Lists</title>
</head>
<body>
It’s assessment time!
<ol>
<li>Turn in your assignment</li>
<li_Tick your name off the list</li>
</ol>
If your name is not on the list contact your tutor directly
<ol>
<li value="4">Now you can relax</li>
</ol>
</body>
</html>
```
Creating Lists

- Definition Lists
  - The Definition list uses the <dl> tag
  - The <dt> tag is used for definition terms
  - The <dd> tag is used for their definitions

```html
<dl>
  <dt>The definition term</dt>
  <dd>Is defined below the term</dd>
  <dt>The second definition term</dt>
  <dd>Is defined below the second term</dd>
</dl>
```
Creating Lists

Nested Lists

- Simply place one list inside another:
  
  ```html
  <ul>
      <li>First item</li>
      <li>Second item</li>
      <ul>
          <li>first sub-item</li>
          <li>second sub-item</li>
      </ul>
      <li>Third item</li>
  </ul>
  ```