

HTML Review

*<http://www.htmldog.com/> is used as a reference for most of the notes in this powerpoint

Getting Started

- **HTML** files are nothing more than simple text files, so to start writing in **HTML**, you need nothing more than a simple text editor
- HTML stands for hyper text markup language

Reminder

- Be careful. It is important that the extension “.html” is specified - some text editors, such as Notepad, will automatically save it as “.txt” otherwise.
- You also need to ensure that your file is being saved as **plain text**. TextEdit, for example, will start new files as “Rich text”, containing lots of formatting extras, by default. In such cases, go into the preferences and make sure you check the “Plain text” format option **before creating** a new file.

Tags, Attributes, and Elements

- Although the basics of HTML is plain text, we need a bit more to make it a nice and shiny HTML document.
- **Tags**
- The basic structure of an HTML document includes **tags**, which surround content and apply meaning to it.

```
<!DOCTYPE html>
```

```
<html>
```

```
<body> This is my first web page </body>
```

```
</html>
```

- The first line on the top, `<!DOCTYPE html>`, is a **document type declaration** and it lets the browser know which flavor of HTML you're using (HTML5, in this case).

- `<html>` is the **opening tag** that kicks things off and tells the browser that everything between that and the `</html>` **closing tag** is an HTML document. The stuff between `<body>` and `</body>` is the main content of the document that will appear in the browser window.

Closing tags

- The `</body>` and `</html>` put a close to their respective elements (more on elements in a moment).
- Not all tags have closing tags like this (`<html></html>`) some tags, which do not wrap around content will close themselves. The line-break tag for example, looks like this : `
` - a line break doesn't hold any content so the tag merrily sits by its lonely self. We will come across these examples later. All you need to remember is that all tags with content between them should be closed, in the format of opening tag → content → closing tag. It isn't, strictly speaking, always a requirement, but it's a convention we're using in these tutorials because it's good practice that results in cleaner, easier to understand code.

- You might come across “self-closing” tags, whereby a br tag, for example, will look like “
” instead of simply “
”. This is a remnant of XHTML, a form of HTML based on another markup language called XML.

Attributes

- Tags can also have **attributes**, which are extra bits of information. Attributes appear inside the opening tag and their values sit inside quotation marks. They look something like `<tag attribute="value">Margarine</tag>`.

Elements

- Tags tend not to do much more than mark the beginning and end of an element. Elements are the bits that make up web pages. You would say, for example, that everything that is in between (and includes) the `<body>` and `</body>` tags is the body element. As another example, whereas “`<title>`” and “`</title>`” are **tags**, “`<title>Rumple Stiltskin</title>`” is a title **element**.

Page Titles

- All HTML pages should have a page **title**

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>My first web page</title>
```

```
  </head>
```

```
    <body> This is my first web page </body>
```

```
</html>.
```

- We have added two new elements here, that start with the [head](#) tag and the [title](#) tag (and see how both of these close).
- The head element (that which starts with the <head>opening tag and ends with the </head> closing tag) appears before the body element (starting with <body>and ending with </body>) and contains information **about** the page. The information in the head element does not appear in the browser window.
- We will see later on that other elements can appear inside the head element, but the most important of them is the **title** element.

Paragraphs

```
<!DOCTYPE html>
```

```
<html>
```

```
  <head>
```

```
    <title>My first web page</title>
```

```
  </head>
```

```
  <body>
```

```
    This is my first web page.
```

```
    How exciting
```

```
  </body>
```

```
</html>
```

Emphasis

- You can emphasize text in a paragraph using em (emphasis) and strong (strong importance).

<p>Yes, that really is exciting.

Warning: level of excitement may cause head to explode.</p>

Line breaks

- The line-break tag can also be used to separate lines like this:
- This is my first web page
 How exciting
- There's no content involved in breaking lines so there is no closing tag.

Headings

- They are [h1](#), [h2](#), [h3](#), [h4](#), [h5](#) and [h6](#), [h1](#) being the almighty emperor of headings and [h6](#) being the lowest pleb.
- `<h1>My first web page</h1>`
- Note that the [h1](#) tag is only used once, as the main heading of the page. [h2](#) to [h6](#), however, can be used as often as desired, but they should always be used in order, as they were intended. For example, an [h4](#) should be a sub-heading of an [h3](#), which should be a sub-heading of an [h2](#).

Lists

- There are three types of list; **unordered lists**, **ordered lists** and definition lists.
- Unordered lists and ordered lists work the same way, except that the former is used for non-sequential lists with list items usually preceded by bullets and the latter is for sequential lists, which are normally represented by incremental numbers.
- The ul tag is used to define unordered lists and the ol tag is used to define ordered lists. Inside the lists, the li tag is used to define each list item.

```
<ul>
```

```
  <li>To learn HTML</li>
```

```
  <li>To show off</li>
```

```
  <li>Because I've fallen in love with my  
computer and want to give her some HTML  
loving.</li>
```

```
</ul>
```

Exercises

- Using the tags you have learned in this lesson create a website that will represent the work you do in this class. Use the directory tree map provided to organize your work.
- Start by making a homepage. Make sure to save it as **index.html** (make sure everything is lowercase-web page servers are case sensitive and don't add unusual characters to your file or folder names). Save this in your root folder. Include the course code as the title and the course name as the heading. Provide a paragraph describing yourself and why you choose this course.
- Next create a separate page for each of your 5 units. Place these in each of your 5 unit folders. In each page add an appropriate title and an appropriate heading (for example the unit number as a title and the name of the unit as a heading).
- Under each unit's heading create a paragraph outlining the general description of the unit (see your course outline).
- Create a subsection in each page. Title it "**My Work**". Underneath it create a list of 5 items. Each item name "**Item 1**", "**Item 2**", "**Item 3**" and so on. These will be replaced with links to your completed class handouts and work.
- Use the emphasis tags where possible to highlight special terms.
- When you are done, upload the contents of your websites root folder to your server space at **sdsscomputers.com**.
- Visit **sdsscomputers.com** to make sure your site is working.
- Don't worry about images and links yet. We will get to that soon!