Web and Mobile Application Development

Image from https://www.teaching-materials.org/htmlcss-1day/html-basics

Material created by:
David Augenblick, Bill Mongan, Dan Ziegler, Samantha Bewley, and Matt Burlick

Web and Mobile Application Development

Excerpts Taken from “How to Create Your Own Personal Website” - A DUCSTeach Presentation

Download Samples & Examples
http://examples.ducsteach.org

Intro

- A website is made up of three major parts:
  1. “Code” that specifies how/where to render the images and objects on the page
     - The browser does the actual rendering. We can interpret theses specifications as instructions for the browser like “render a box of width 5px here”
  2. Code that stylizes the objects and text on the page
  3. Code that dynamically alters the page
- These are better known as
  - **HTML**
  - **CSS**
  - **JavaScript**

About HTML

- HTML (**H**yper**t**ext **M**arkup **L**anguage) is a standardized system for tagging text files to achieve font, color, graphic, and hyperlink effects on World Wide Web pages.
- Browsers interpret HTML (as well as CSS and JavaScript) in order to display the page that you coded.
- This means that each browser can interpret your code differently based upon its DOM (**D**ocument **O**bject **M**odel, discussed in the next few slide-sets) interpreter.

About HTML

- Note:
  - Chrome and Firefox are built off of the same Netscape renderer.
  - Internet Explorer used its own renderer (and this is why many webpages aren't compatible with it!)
  - Safari and Opera also use proprietary renderers.
- As browsers update, in addition to security vulnerabilities, they also add compatibility for new HTML standards (i.e. new tags in HTML5).

Notes About HTML

- HTML is a case sensitive language. You can't mix and match tags with different capitalization.
  - `<p> </P>` would not close, for example
  - However, some browsers do allow this (but it's not good practice)
- Not every tag or every attribute of a tag is supported by every browser
  - Consult resources such as WW3 schools for version numbers and browser platform compatibility
- HTML doesn't need to be compiled
  - What you write is what will be used! Your browser "interprets" it on the fly.

Notes About HTML

- HTML can be edited with any text editor
  - Sublime text
  - Notepad++
  - Etc..
- If you're working with a compiled language, or if you're using Git, you may want to use a full IDE for web application development since there's some nice integration tools available
  - WebStorms (or any JetBrains IDE)
  - But for this course it's probably better just to stick to a basic text editor

Tag Types

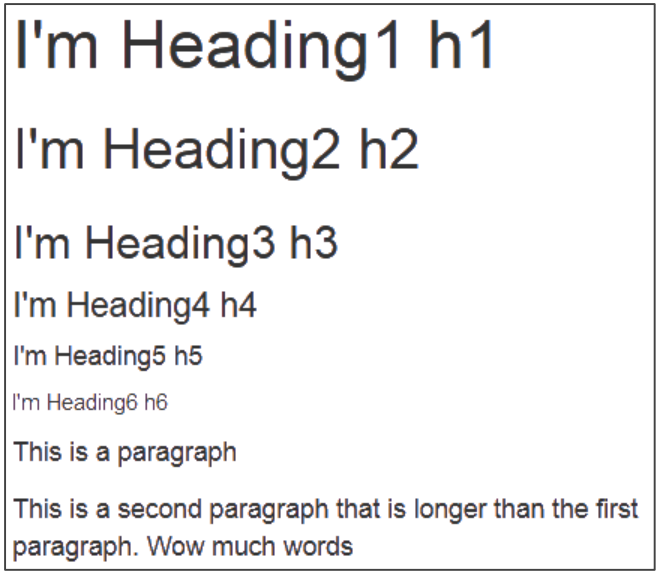
Basic Tags and Layout

- Every HTML Document has a basic structure.
  - `<html>` - Starts and ends every webpage
  - `<head>` - Used to section off for styles, page set ups, and “includes”
  - `<title>` - Shows in the tab at the top of your internet browser
  - `<body>` - the main area of the HTML document

```
<html>
<head>
    <!-- This is a comment -->
    <title>Drexel University</title>
    <!-- Javascript and CSS References -->
</head>
<body>
This is where the text would go!
More about formatting this on the next
couple of slides.
</body>
</html>
```

Headers and Paragraphs

- Header Tags `<h1>...<h6>` are used to represent main idea points or major sections of your web page.
- They come in 6 sizes with lower numbers corresponding to larger headings. Think ranking, the higher the rank the larger it is.
- Paragraph Tags `<p>` are used to insert paragraphs or chunks of text. They can be used for any amount of text.
- Remember to close tags after the content with a `</>` ex. `</p>` or `</h1>`
- We can also insert line breaks `<br>`
  - Note that since a line break doesn't encapsulate an area, there is not closing tags




```
<h1>I'm Heading1 h1</h1>
<h2>I'm Heading2 h2</h2>
<h3>I'm Heading3 h3</h3>
<h4>I'm Heading4 h4</h4>
<h5>I'm Heading5 h5</h5>
<h6>I'm Heading6 h6</h6>
<p>This is a paragraph</p>
<p>This is a second paragraph that is longer than the first paragraph. Wow much words</p>
```

Links (Anchor Tags)

- Anchor tags are used to navigate to different pages of your website and can also be used to link to other websites.
- Inside the `<a>` tag, there is an attribute named "`href`". You set "`href`" to the location of where you want the link to go to.

A link to drexel.edu

```
<a href="http://drexel.edu/">A link to drexel.edu</a>
```

Relative Link

```
<a href="page1.html">Relative Link</a>
```

Note: page1.html must be in the same directory as your website page for the link to work.

Image Tags

- Image tags, `<img src="link to image">`, are used to embed images in your web page.
- Inside the `<img>` tag, there is an attribute named "`src`". You set "`src`" to the location of your image.
- There is another useful attribute named "`alt`". This is used to specify the description of the image.
- To specify the width and height of an image we can add the `height` and `width` attribute.
    - We set the `height` and `width` of the image to an integer which maps to how many pixels should be used for the height and width.

I Can Has img

```
<h3>I Can Has img</h3>
<img src="" alt="Cat in Space" width="250px" height="200px">
```

Lists

- A list is a set of items.
- We can create two type of lists in html.
  - `ul`: Unordered List - a list of items in which the order does not explicitly matter. Items in unordered list are marked with bullets.
  - `ol`:Ordered List - a list of items where order in which they appear matters. List items in ordered list are marked with numbers.
- We can define ordered list using `<ol>` and unordered list using `<ul>` tag in html.
- We can define individual list items using tag `<li>`.

Sample html code:

Unordered List

```
<ul>
  <li>Home</li>
  <li>Products</li>
  <li>About us</li>
  <li>Contact</li>
</ul>
```

Ordered List

```
<ol>
  <li>Home</li>
  <li>Products</.li>
  <li>About us</li>
  <li>Contact</li>
</ol>
```

Output:

- Home
- Products
- About us
- Contact

1. Home
2. Products
3. About us
4. Contact

Tables

- Organize data that requires grid layout
- Code Specifics
  - **`<table>`** : creates a table
  - **`<th>`** : header of table
  - **`<tr>`** : table row; creates row to add table data
  - **`<td>`** : content to be placed in cell
- Several of these tags have common/useful attributes:
  - **`colspan`**: attribute for content that covers many columns (horizontal)
  - **`rowspan`**: attribute for content that covers many rows (vertical)
  - **`border`**: attribute specifies border width, type of line, and color of line
    - width: written in px, pt, cm, em, etc.
    - type of line: solid, dotted, dashed, double, etc.
    - color: can be written in hex or name
  - **`padding`**: adds space between border and table data; specify in pixels

Tables

```
<body>
<table>
 <tr>
   <th> Club/Organization </th>
   <th> About </th>
   <th> Website </th>
 </tr>
 <tr>
   <td> DUCSTeach </td>
   <td> A student led organization of volunteers from Drexel University providing technical assistance to local schools and community groups in need. </td>
   <td> <a href="https://www.cs.drexel.edu/~dmz38/ducsteach/">www.ducsteach.org </a> </td>
 </tr>
 <tr>
 <td colspan="3"> <img src="DUCSTeach.png" width="80%"> </td>
 </tr>
</table>
</body>
</html>
```

Club/Organization	About	Website
DUCSTeach	A student led organization of volunteers from Drexel University providing technical assistance to local schools and community groups in need.	www.ducsteach.org

Form Elements

- Items that are meant to be interacted with fall into the category of *form elements*
- We'll look more formally at forms later, but for now let's just make sure we know some of the critical form element types
- The most common type is the `<input>` tag. This can come in various different "flavors" based on the type attribute:
  - Text Box: `<input type=text>`
  - Password: `<input type=password>`
  - Check box: `<input type=checkbox>`
    - This has an optional attribute `checked` to pre-check it
  - Button: `<input type=button>`
  - Radio: `<input type=radio>`
- And each has additional attributes you might set

Form Elements

- Some additional common form elements are:
  - Textarea (for multi-line input)

```
<textarea>…</textarea>
```

  - Dropdown select

```
<select>
      <option>…</option>
</select>
```

Example

```
<input type=text name=name>
<input type=password name=pass>
<select name=notify>
    <option value=0>No</option>
    <option value=1>Yes</option>
</select>
<input type=button value=Register>
```

HTML5

- HTML5 builds upon HTML4 (what we've done thusfar) to include new features and remove outdated ones.
- New HTML5 *Elements*
  - Control elements
    - `<datalist>` - Provides a drop down list of pre-defined options for an <input> element.
    - Additional input types (limited support)
      - `Range` (slider), `date`, `color`
  - Semantic elements
    - `<header>`, `<footer>`, `<progress>`, `<figure>`
  - Graphics elements like `canvas`
  - Multimedia elements like `audio` and `video`
  - Additional tags `<sup>` (superscript) `<sub>` (subscript), `<label>`

HTML5

- Figures:

```
<figure>
      <img src=….>
      <figcaption>…</figcaption>
</figure>
```

- Video Controls

```
<video controls>
      <source src=…. type=video/mp3>
</video>
```

HTML5

- Additional attributes
  - `placeholder` – Allows us to put temporary stuff in a typeable area until the user types.
  - `min, max`
- Additional APIs
  - Geolocation
  - Drag and Drop
  - Local Storage
- A lot of these are must useful when combined with CSS and Javascript, so we'll look at HTML5 again a little later.