Full Stack Web development

Lecture 3: CSS Introduction & Page Layout

Today:

- Revisit the HTML element:
  - ˜ <div>
  - ˜ <section>
  - ˜ <header>
  - ˜ <footer>

- Learn
  - ˜ Cascading Style Sheets (CSS)
  - ˜ Browser Compatibility
  - ˜ Responsive Web Design
  - ˜ CSS3

The <DIV> HTML Element

- The most powerful layout element in current web development.

- So powerful that special <DIV> elements, <HEADER>, <FOOTER>, and <SECTION> were created.

- Review of how it works.

What are Cascading Style Sheets?
Support for advanced styling and building complex layouts in HTML.

- Advantages
  - Enables complex layouts
  - Adds better support for mobile and web layouts
  - Better support for disabled web browsers
  - Easier to read code.

- Disadvantages
  - Very complex
  - Browser compatibility issues
  - Can be difficult to debug.

CSS attributes

Id="x"

• Should apply to a single element

• Accessed by "#"
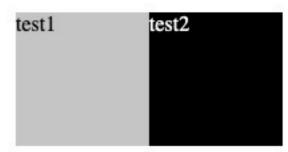
Class="y"

• Should apply to a group of elements

• Accessed by "."

A Basic Example

```
<style>
.testing { float:left; height:100px; width:100px }
#test1 { background-color:#cccccc }
#test2 { color:#ffffff; background-color:#000000 }
</style>

<div id=test1 class=testing>test1</div>
<div id=test2 class=testing>test2</div>
```

Output:

Anatomy of a CSS declaration

```
<STYLE TYPE="TEXT/CSS">
SELECTOR {
    PROPERTY: VALUE;
    PROPERTY: VALUE;
}
<STYLE>
```

- **_Selector_**: *The HTML Element, Class, ID, or nested combination to identify a specific element or groups of elements.*
- **_Property_**: *A list of allowed properties for a given element.* https://www.w3schools.com/cssref/default.asp
- **_Value_**: *A list of allowed values for a specific property.*

CSS Selectors

- Select a single **ID** of any element type `#id1`
- Select a single **ID** of a **<u>div</u>** element `div#id1` notice the div?
- Select a single **Class** `.class1` Notice the .
- Select **all** elements `*`
- Select all <p> tags *inside* a specific div class `div.class1 p`
- Select all div *and* p elements `div, p`
- Select all links with a href to "https://disney.com"
- `[href="https://disney.com"]`

Common CSS Properties

- align
- background
- border
- color
- display
- box-sizing
- visibility
- float
- font
- padding
- margin

- color
- position
- z-index
- width
- height
- list-style
- opacity
- text-decoration
- text-align
- transform

Embed CSS Three Ways

- Directly write CSS into an HTML element. **INLINE**

   `<p style="text-align:center">`

- Write CSS in the HTML page using a <style> tag.
  INTERNAL

   `<style> p { text-align:center} </style>`

- Embed a separate CSS file into your webpage.
  EXTERNAL

   `<link rel="stylesheet" href="style.css">`
  - The advantage of this approach is that you can have one style.css file that is shared across your entire website.

CSS Specificity

- In general (higher priority first)
  - `!important` css declaration
  - Direct `style=""` attributes on an HTML element
  - targeting an element directly `p { color: red; }`
  - targeting an id
  - targeting a class
  - Universal selector *

Further Reading:

https://developer.mozilla.org/en-US/docs/Web/CSS/Specificity

https://specifishity.com/

https://www.w3schools.com/css/css_specificity.asp

CSS Specificity

- OMG!! There are so many different ways to write CSS!!! What takes precedence?!?

Consider:

```
<style>
  p { font-size:12px; }
  p.test { font-size:14px; }
  p.test2 { font-size:16px; }
</style>

<p class="test" id="test2" style="font-size:10px">
```

What font size is the <p> tag?

CSS Media Queries

Media queries allow you to tailor your CSS to different media types, like printing and mobile devices, in addition to web browser screens.

Example:

```
@media screen and (max-width: 350px) {
   body { background-color: orange }
}
@media print {
   body { background-color:white }
}
```

We will revisit in the advanced CSS and UI/UX areas of this course.

Let's Build another Page

- Open your about me page
- Add some identifiers (id, name, class, etc)
- Build out some styles
- Don't forget to use W3Schools CSS reference!!

Project 1 – Due 2/15

- Come up with a concept company (or a real one) that you want to build a website for. Build a 5-page (minimum) website for the company, homepage, 2 content pages, offer/sale page (with a list of said offers), and a sign-up or contact form.
- The design should be of professional quality and should mimic some other website you have found online (cite source(s)).
- As you are developing the website, create a list of things you would like to do, but cannot with the things you have learned so far.