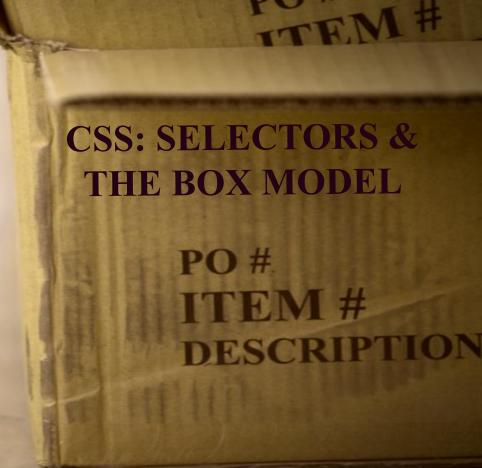
Module 3 Day 2



Module 3 Day 2

Can You ... / Do You ... ?

- Utilize the following selectors to apply CSS styles:
 - element
 - \circ id
 - class
 - universal
 - attribute
 - pseudo (hover, focus)
 - descendant combinator
- Understand and utilize selector specificity and cascading to apply styles
- Describe the box model (padding, margin, border)
- Understand and demonstrate the use of block, inline and inline-block
- Understand the effect of "normal flow" on the positioning of elements on a page
- Describe and use the different positioning styles: static, fixed, absolute, and relative
- Understand the difference between common units of measure (px, %, rem) and be able to choose an appropriate unit for CSS styles

CSS: Cascading Style Sheets

- CSS allows us to define how an element appears on a website through the use of style rules.
- These rules can control everything from colors and fonts, to positioning and behavior.
- For the purposes of layout and design, every element is thought of as a box, and a style rule is meant to format that box (and its contents).

CSS: Using Stylesheets

Internally:

CSS can be used by declaring a <style>...rules...</style> element within the <head></head>

Externally:

CSS files are linked, related, the a webpage (HTML) using the k)> element within the <head></head>. Link elements should have three attributes:

```
href = "path_to_the_stylesheet.css"
rel = "stylesheet"
type = "text/css"
```

CSS: The Box Model

When you look at a website, learn to "see" every element as a box.

This box surrounds both **BLOCK** level and **IN-LINE** level elements.

The box has four (4) parts:

Margin - Outside Edge

Border - Outline

Padding - Surrounds Content

Content - Inner HTML & Text

Block:

>

(Start on a new line) <div> <h1-6>

In-line:

(Flow with text)

 & <i>
<imq>



CSS: Application of Style Rules

```
Boxes:
    Background - Color and Image
    Border - Width, Style (Solid, Dashes, Dots), Color
    Position
Text:
    Font (Typeface)
    Color
    Size
    Format (Bold, Italic, Upper\Lower Case)
Special Element-Specific Styling Attributes Exist For:
    Forms
    Lists
    Tables
```

CSS: Anatomy of a Style Rule

Style rules have four parts:

```
SelectordivDeclaration{ border-style: solid; border-width: 2px; }Properties{ border-style: solid; border-width: 2px; }Values{ border-style: solid; border-width: 2px; }
```

```
div { border-style: solid; border-width: 2px; }
```

CSS: Selectors - The Basics

•	Universal: * {}		* {}	Targets all elements on a page		
•	Element: h1 {}		h1 {}	Targets all h1 elements on a page		
•	ID:		#userphone {}	Targets the element whose ID="userphone"		
•	• Class:		.info	Targets any element with a class attribute with a value "info"		
			div.info	Targets only div elements with a class attribute value "info"		
•	Descendent:		div ul {}	Targets any <i>ul</i> elements inside a <i>div</i> element <i>regardless</i> of other elements in the <i>div</i>		
•	Child:		li>a {}	Targets any a element that is the direct child of an li element		
•	Siblings:					
	0	Adjacent:	h1+p {}	Targets the next <i>p</i> element after an <i>h1</i> element under the same Parent element		
	0	General:	h1~p {}	Targets all <i>p</i> elements that are siblings of an <i>h1</i> without regard to its postion		

CSS: Selectors - Beyond The Basics

•	Attributes						
	0	Existence:	h1[class]	Targe	ets all h1 tags with a class attribute		
	0	Equality	h1[class="note"]	Targets all h1 tags with a class attribute value equal to "note"			
	0	Space	h1[class~="note"]	Targets all h1 tags with space delimited attributes in which one value			
		is equal to "note"					
	0	Prefix	div[attr^"n"]	Targets all div tags with an attribute value beginning with "n"			
	0	Substring	div[attr*"ot"]	Targets all div tags with attribute values containing the letters "ot"			
	0	Suffix	div[attr\$"e"]	Targe	ts all div tags with an attribute value ending in "e"		
•	Pseu	do					
	0	:hover	input[type="text"]:h	nover	Applies when the cursor or pointer is over an element		
	0	:focus	input[type="text"]:focus		Applies when an element has the focus (is ready for and the		
					focus of user interaction)		
	0	:active	input[type="text"]:a	active	Applies when a user is interacting with the element		