Class 3 notes

# Welcome

# CSS Review – Id and Class Support CSS

CSS – Cascading Style Sheets provide the styling of the web page – color, font, and alignment. If you think of each element of a web page being within a box, the CSS for that element controls the design of that box.

Html attributes that aid CSS – ID and Class

# CSS Review - The CSS Rule

Two parts – Selector (element the rule applies to) and Declaration (how it should look)

# CSS Review – Selectors

Selector can be:

* Universal selector
* Type or HTML element,
* Class
* ID
* Relative

# CSS Review – Multiple Keyword Selectors

The selector can be a single keyword or a group of keywords

Comma separated means multiple selectors

Space separated means specific nested selector i.e. child selector

# CSS Review – Font and Color Declarations

Declarations are property - value pairs.

The property and value can be single keywords or space separated lists.

Typeface is a somewhat esoteric but important feature of web design. Good designers spend hours finding or creating typeface for webpages.

There are default typefaces – font-families - and specialized ones that are downloaded to the browser

* Typeface terminology:
  + Generic Font-families
    - Serif
    - San-serif
    - Monotype
    - Cursive
    - Fantasy
  + Qualities
    - Weight – light, normal, bold
    - Size – pixels, percentage, ems and relational
    - Kerning (stretch)
  + Font – ordered space separated keywords to describe a font:
    - Style
    - Weight
    - Size
    - Family
    - Color
* Three ways to designate the same color:
  + Color name - 147 color names
  + RGB comma separated list – how much Red, Yellow, and Blue used. 0,0,0 is Black 255, 255, 255 is white
  + Hex value - #000000 Black #FFFFFF is white
* Background Color
* Contrast

# CSS Review - 3 Ways to Add CSS to HTML Page

CSS is accessed for a webpage in three ways:

* As an attribute of a element’s tag
* In the HTML file
* External CSS file

# Cascading

Cascading

* Inheritance
  + Auto inheritance usually those related to text e.g. font-family or color
  + Not inherited – background-color, border
    - Use inherit value
* Precedence
* Last Rule
* Specificity
* Important!

~~Just as different browsers may present HTML tags differently, CSS isn’t implemented the same on every browser - although the browser wars have calmed down a lot in the last few years.~~

~~These are called browser quirks or CSS bugs.~~

~~To see how a page looks in various browsers, use:~~

* ~~BrowserCam.com,~~
* ~~Browser.Lab.Adobe.com,~~
* ~~CrossBrowserTesting.org~~

~~For work arounds look at:~~

* ~~PositionIsEverything.net~~
* ~~QuirksMode.org~~

# What We Will Be Developing Today

* Create a page like this using div and span tags to add positioning and style text.
* Use pseudo-classes to style links
* Applying the box model to space and size blocks of code.

# Inline & Blocking Elements

Positioning elements well on a webpage gives it a professional polish. If we think of each element existing in it’s own box, then the styling of the elements makes more intuitive sense.

Think of the elements as being either

* Block-level – each element begins on a new line, like a header <h1> or paragraph <p>
* Inline – each element follows the previous element on the same line (unless a margin makes it wraps to the next line) like anchors, images, emphasis, and important.

# Group Elements Within Divs

The div tag is used everywhere to group elements together into sections.

Use div tags to apply CSS to a group of elements like paragraphs or lists.

# Add CSS to the Grouped Elements

For example, what if we want the first 2 paragraphs of a section to be right-aligned, purple & bold, but we don't want any other paragraphs to be right-aligned?

We would wrap them in a div element to style them differently

# Let’s Develop It

Help class develop the page with header, footer, content area and sidebar.

# Element: Span

Inline element. Each new span is rendered next to each other &amp; only wraps when it reaches the edge of the containing element

Can be used to apply styles to text inline without breaking the flow of content.

# Pseudo-Classes, More Style for Links

Pseudo-Classes act as an additional parameter for an element type selector. We are going to use the anchor (link) element and apply two presentation pseudo-classes on that element. In most browsers, the links are styled as blue and underlined and change color when visited. With :link and :visited, you can add your own styling to override the default the styling for the link.

# Pseudo-classes: Responding to the User

But wait! There’s more. Wouldn’t it be cool if you could have CSS that responded to the user interacting with the link in other ways? Pseudo-classes :hover, :active, and :focus are responsive to user.

List the pseudo-classes in the preferred order in the style sheet for the links to be responsive in the expected way.

# Let’s Develop It

Help students add pseudo-codes

# Box Model

Each element of a web page has a box around it. There is a default box style applied to each element. Since the box model is not inherited by nested, child elements, each element type that does not take the default values for it’s box, needs to be styled individually.

The style of the box includes, margin, padding, border, dimension (width and height), limiting dimension, and overflow.

Margin, padding and border work together. Margin the offset or space between one element and another. It is outside the border.

Padding is the offset or space between the border and the content. It is the space inside the border.

# Padding

Padding (and margin) is usually specified in pixels – or number of dots of light on the display = but it can also be ems or percentage. Percentage is the percent of the containing element – like another box. If there is no parent container, it is percentage of the total display.

# Padding – Individual Rules

To create uniform padding you can give it a single property – value pair using the keyword padding.

Other wise use hyphenated keywords with the direction: top, bottom, right, and left;