Class 3 notes

# Review (and a couple of important slides that got missed)

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

The CSS Rule

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

Selector can be:

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

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

Declarations are property - value pairs.

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

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

* 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 diffently, CSS isn’t implemented the same on every browser., although the browser wars have calmed down a lot in the last fe years.

These are called browser quirks or CSS bugs. Use BrowserCam.com, Browser.Lab.Adobe.com, CrossBrowserTesting.org to see how a page looks in various browsers.net

For work arounds use PositionIsEverything.net or QuirksMode.org

Property: Color

Coloring the type can be used to really give zing to a page.

Three ways to designate the same color:

Color name –

RGB comma separated list – how much Red, Green, and Blue used. This is the color wheel for transmission not reflection as you are used to with Red, Yellow, Blue crayons 0,0,0 is Black 255, 255, 255 is white,

Hex value - #000000 Black #FFFFFF is white

Background Color

Contrast

Property: Font

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

# Layout – Inline vs Block

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, important.
* Nesting...is complicated.

Generally speaking, inline elements nest well inside other inline element or in a block element. Some block elements nest well lists and tables, for example but paragraphs and headers don’t nest inside each other or themselves – Try it: <h1><h2></h2></h1>.

When an element is nested, the outer element is the containing or parent, the inner element is the child. As we said, some CSS rules are inherited by default and some are not, and some can be made to inherit explicitly using the inherit property.

# Element: Div

The div tag is the workhorse block-level tag. It is often given an id or a class attribute. In HTML 5 – some of the most common div id combos become their own tags ex. <div id=”footer”> becomes <footer> and <div id =”header”> becomes <header>. But that is for future classes.

Remember that an id should only appear once on a page!

# Grouping 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.

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.

# Positioning

CSS provides an number of schemes to position elements on the page

Position rules can have a number of values:

* Static – this is the default positioning, so rarely used explicitly. The normal flow is each element is stacked under the preceding element.
* Relative – The element is moved in relation to where it would have been in the normal flow.
* Absolute – The element is taken out of the normal flow and placed specifically on the containing element using top and left or right and bottom properties
* Fixed – The element is specifically placed relative to the window, so when the user scrolls down, the element remains in the same place – used to be popular for menus but not so much anymore.

When add

# 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 so as not to break the flow of content.