**Midterm Notes**

**Global attributes**

-----------------

**HTML Attributes:**

Global

Element\_Specific

data -\*

role, aria -\*

-----

WAI-ARIA, WCAG 2.0

**HTML Elements**

html

head

title, base, link, meta, style

body

article, aside, max, section

header, footer

main (main content)

address (contact information)

div (generic divider for block elements)

h1, h2, h3, h4, h5, h6 (6 levels of headings)

p

hr (horizontal line)

pre (preformatted text)

blockquote (for quotations)

ol, ul (ordered list, unordered list)

li (description list)

dl

dt, dd

figure

figcaption

a

em, strong, small, s

cite, q

dfn, abbr (defining instance, abbreviation)

ruby, rb, rt, rtc, rp (ruby annotations)

data, time

code, var, samp, kbd

sup, sub

i, b, u, mark

bdi, bdo (bidirectional text)

span (generic divider for inline elements)

br, wbr

ins, del (document insertion and deletion)

table

caption

colgroup, col

thead, tbody, tfoot

tr

th, td

**CSS**

* Cascading Style Sheet
* Specify the presentation aspects
* Developed by Hakon Wium Lie (CHSS) and Bert Bos (SSP)

**CSS Versions**

* CSS 1
* CSS 2.1
* CSS 3
* CSS Preprocessors; CSS Frameworks
* Sass, Less, 960 Grid Systems, Bootstrap, Foundation, Materialize, etc.

**HTML/XHTML Stylesheets**

* Author styles – external stylesheets (recommended), embedded style, and inline styles
* User style
* User agent style

**CSS Statements (At-Rules)**

@charset

@import

@media

@font-face (define and download webfonts)

@keyframes (keyframe animation)

@page

* CSS Rule Sets (aka CSS Rules, Style Rules)

consists of a selector, followed by a brace-enclosed declaration block \_\_\_\_\_\_

e.g. \*body is the selector

body {

display: none;

color: blue;

}

**CSS Selectors**

* Selector
* structure used as a condition in a CSS rule to determine which elements in the document tree are matched by the selector and are thus targeted by the formatting specified in the CSS rule declaration block.
* The matched elements are called the subjects of the selector \_\_\_
* Selector syntax
* chain of one or more sequences of simple selectors separated by combinators, with one pseudo-element possibly appended to the last sequence.
* example: div #abc > p.xyz[title] + span:first\_child::after

>, + = contributor

after = pseudo-element

span:first\_child::after = pseudo-class

* Sequence of simple selectors
* chain of simple selectors not separated by combinators
* always starts with a type selector or a universal selector
* cannot contain other type selectors or universal selectors
* Group of selectors
* Comma – separated list of selectors representing the union of all elements selected by each of the selectors in the list
* e.g. h1, h2, h3
* Simple selectors
* type selector (p, div, span)
* universal selector (\*)
* attribute selector

[attr]

[attr=value]

[attr~=value] (class attribute)

[attr|=value] (language attribute)

[attr^=value] (start CSS3)

[attr$=value] (end CSS3)

[attr\*=value] (anywhere in between CSS3)

* class selector (<p class="xyz">)
* ID selector (<div id="abc"> \*for ID attribute)
* pseudo-selector

- dynamic psuedo-class

link pseudo-classes

:link

:visited

user action pseudo-classes

:hover

:active

:focus

- target pseudo-class

:target (CSS3)

- language pseudo-class

:lang()

- UI element states psedo-class

:enabled (CSS3)

:disabled (CSS3)

:checked (CSS3)

:indeterminate (CSS3)

- structural pseudo-class

:root

:first-child

:last-child

:only-child

:nth-child()

:nth-last-child

:first-of-type

:last-of-type

:only-of-type

:nth-of-type()

:nth-last-of-type()

:empty

\*all CSS3 except first-child

- negation

:not (CSS3)

* Combinators
* descendant combinator (whitespace ie space, tab, line feed, carriage return, form feed)
* child combinator (>)
* sibling combinators

adjacent sibling combinator (+)

general sibling combinator (~) CSS3

* Pseudo-elements

::first-letter , :first-letter

::first-line , :first-line

::before , :before

::after , :after

**CSS Rule Precedence**

* by origin and importance
* user agent important declarations
* user important declarations
* author important declarations
* author normal declarations
* user normal declarations
* user agent normal declarations
* by specificity
* inline style
* number of ID selectors
* number of class selectors, attribute selectors and pseudo-classes
* number of type selectors and pseudo-elements
* by order