Midterms

**Cascading Style Sheets**

* specify presentation aspects of structurally marked up documents
* developed by Hakon Wium Lie (CHSS) and Bert Bos (SSP)

Versions:

* + CSS 1, 1996
  + CSS 2.1, 2011
  + CSS 3
* CSS Preprocessors , CSS Frameworks
* Sass, Less, 960, Grid System, Bootstraps, Foundation, Materialize, etc.
* Sass – Syntactically awesome style sheet

**HTML/XHTML Stylesheets**

* Author styles
* External stylesheets
* Embedded styles
* Inline styles
* Use <Link> to attach stylesheets for external
* Use <Style> for embedded style, used for quick prototyping
* <Style> attribute used for inline
* User styles
* Preferences of the user (browser extensions)
* User agent styles
* Example default CSS 2.1 stylesheet for HTML 4

**CSS Statements**

* At-rules
* @charset – including other characters
* @import – including other content
* @media – media queries
* @font-face – include other fonts
* @keyframes – keyframe animations
* @page
* @document
* CSS Rule Sets (a.k.a CSS rules, style rules)
* Consists of a selector, followed by a brace-enclosed

**CSS Selectors**

* Selector
* 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
* Matched elements
* Selector Syntax
* Chain of one or more sequences of simple selectors separated by combinators, with one pseudo-element possibly appended to the last sequence

**div#abc>p.xyz[title] + span::first-child::after**

* Sequence of simple selectors
  + Chain of simple selectors not separated by combinators
  + Always start 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
* Simple Selectors
* Type selector
* P, div, span
* Universal selector
* \*
* Attribute selector - targets an element based on the value
* [attr]
* [attr = value]
* [attr ~ = value]
* [attr |= value]
* [attr ^= value]
* [attr $= value] CSS 3
* [attr \*= value]
* Class selector - makes use of a period
* ID selector
* Pseudo-class
* Dynamic pseudo-class

-link pseudo class

* :link
* :visited

-hover/user action pseudo class

* :hover
* :active
* :focus

-target pseudo class

* :target

-language pseudo class

* :lang()

-UI element pseudo classes

* :enabled
* :disabled
* :checked
* :indeterminate

-structural pseudo classes

* :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
  + Negation pseudo-classes
* :not()
* Combinators
  + Descendant combinator (whitespace)
  + Child-combinator (>)
  + Sibling combinator

-adjacent sibling combinators (+)

-general sibling combinator (~)

* Pseudo- elements
  + ::first-letter
  + ::first-line
  + ::before
  + ::after

**CSS Rule Precedence**

* By origin and importance
* !important

Order:

* 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
* Last one wins

**CSS Declarations**

* Properties
* Shorthand properties
* Vendor specific extensions
* Values
* Keywords
* Numbers

-integers, decimals etc.

* Dimensions

-length, angle, duration, frequency, resolution

* Length units

-font-relative : em, ex ,ch ,rem

-viewport-percentage: vw, vh , vmin, vmax

-absolute lengths: cm, mm, q, in, pt, pc, px

* Angle units: deg, grad, rad, turn
* Duration units: s, ms
* Frequency units: hz, khz
* Resolution units: dpi, dpcm, dppx
* Percentages
* URLs and URIs
* Colors
* Strings
* Functions: calc(), attr(), counter(), linear-gradient(), radial-gradient(),translate(), scale(), rotate(), etc.

**CSS Preprocessors and Frameworks**

* Use advanced syntax, converts syntax into something valid
* Using existing frameworks to create
* Sass - syntactically Awesome Style Sheet, also the most popular
* SCSS
* Bootstrap