**CSS (Cascading Style Sheets)**

* Language used to specify the presentation aspects (i.e. layout and formatting of structurally marked up (e.g. HTML, XML, XHTML, SVG, etc.) document).
* Developed by *Hakon Wium Lie* (HTML Cascading Style Sheets / CHSS) and *Bert Bos* (Stream-based Style Sheet Proposal / SSP)

**Versions:**  CSS 1, CSS 2, CSS 2.1, CSS 3 (structural semantic markup / modularizes CSS 2.1)

**CSS Processors, CSS Frameworks**

-Sass, Less, 960 Grid System, Boot Strap, Foundation, Materialize, etc

**HTML/XHTML Stylesheets**

**1. Author Styles**

-defined by the author

**2. User Styles**

-stylesheet that can be provided by a person visiting the website

-user agent styles (ex. default SSS 2.1 styesheet for HTML 4)

**Reset CSS**

-standard stylesheets that contains all the default formatting

link

media = “print”

title = “normal”

**Multi-line Comment:** /\* \*/

**CSS Statements**

**At-Rules**

@charset – to specify the character set

@import – used to import

@media – used for media queries

@font – face – to define fonts

@keyframes – used for animations

@page – formatting the page for printing

**CSS Rule Set**

* selector - body
* enclosed declaration block - {}
* one property declaration - color: blue
* property name; property value

**Selector Syntax**

-1 or more sequences of simple selectors separated by combinators with 1 pseudo-element possibly appended to the last sequence

div.xyz > p # abc + ul : : after

div.xyz ( equence of simple selectors)

p # abc (other simple selectors)

+ ul (other)

: : after (pseudo-element)

>, + (combinators)

**Type Selector:**

div, p ,ul, hl, body

**Simple Selector**

**Universal selector** (\*) – matches everything

**Attribute selector**

[attr]

[attr = value]

[attr = value]

[attr |= value]

[attr ∧= value]

[attr $= value]

[attr \*= value]

**Class selector**

**ID selector** - <div id = “x”>

**Dynamic**

Link

-link

-visited

User Action (hover, active, focus)

**Target**

**Language**

**UI Element States**

**Target Pseudo-class**

:target

**Language Pseudo-class**

:lang()

**UI Element Pseudo-class**

:enabled

:disabled

:checked

:intermediate

**Structural Pseudo-class**

:root

:first-chid

:last-child

:only-child

:nth-child()

am + b

2m

Odd

Even

:nth-last-child()

:first-of-type

:last-of-type

:only-of-type

:nth-of-type()

:nth-last-of-type()

**Negation Pseudo-class**

:not()

**Combinators**

**-descendant combinators** (whitespace, i, e, space, tab, line feed, carriage return, form feed)

**-child combinators** (>) – direct descendant

**-sibling combinators**

Adjacent sibling combinators (+)

General sibling combinators ()

**Pseudo-Elements**

:first-letter

:first-line

:before

:after

**CSS Rule Precedence**

**1. By origin and importance**

-user agent important declarations

-user important declarations

-author important declarations

-user normal declarations

-author normal declarations

-user agent normal declarations

**2. Specificity**

-inline style

-number of class selectors, attribute selectors, pseudo-classes

-number of type selectors and pseudo-elements

**3. By order**

**CSS Declarations**

**A. Properties**

1. Short-hand Properties

Background; font-family, font-size

2. Vendor-specific Extensions aka Vendor Prefixes

**B. Values**

1. Keywords

2. Numbers (integers & reals in decimal notation)

3. Dimensions

* **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)

4. Value Processing Calculations (First, declared values – cascading yield, cascading value)

\*Percentages

\*URLS and URIS

\*Colors (rgb, rgba, hsl)

\*Strings

\*Functions: calc(), attr(), counter(), counters(), linear-gradient(), radial-gradient(), translate(), scale(), rotate()