**Cascading Style Sheets (CSS)**

* It is a language used to specify the presentation aspects (i.e. layout and formatting of structurally marked up (e.g. HTML, XML, XHTML, SVG, etc.) document.
* It is developed by **HakonWium 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)

* **(X)HTML Style Sheets**
* **Author Styles**– can be specify by the author himself
* **External style sheets (recommended)**

<link rel=“stylesheet” type=“text/css” href=“style.css”>

* **Embedded styles**

<style>

h1{

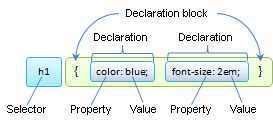
font-size: 1.5em;

}

</style>

* **Inline styles**

<h1 style=”color:fuschia”>Hello, CSS!</h1>

* **User Style** – can be specify by the user
* **User Agent Style** – a browser has default style sheet
* **CSS Statements**
* **At-Rules** (**@**)
* @charset
* @import
* @media
* @font-face – allows the author to download fonts
* @keyframes – it is used for animation
* @page – characteristics of a page
* **CSS Rule Set Structure**
* It is consist of a selector, followed by a declaration block, which contains zero or more semi-colon (;) separated declarations which in turn consists of a property name followed by a colon (:), followed by the value.
* **CSS Selectors**
* It is used to select element of interest
* Matched element is called “subject”

p[title]+#abc:first-child>span::first-letter

adjacent sibling combinatorcombinator

* **Selector Syntax:**
* sequence of simple selectors
* chain of one or more sequences of simple selectors separated by a combinators
* always starts with a type selector or a universal selector
* cannot contain other type of selectors or universal selectors
* group of selectors
* comma separated list of selectors
* **Combinators**
* **Descendant Combinator** - whitespace (i.e. space, tab, line feed, carriage return, form feed)
* **Child Combinator**– (**>**)
* **Sibling Combinators**
* **Adjacent Sibling Combinator**– (**+**) immediately follows
* **General Sibling Combinator**– (**~**) anything that follows
* **Simple Selectors**
* **Type Selector** – it matches elements (p, h1,h2, …)

p{

}

* **Universal Selector** – it matches everything (\*)

\*{

}

* **Grouping Selector**

h1,h2,h3{

}

* **Attribute Selector** – allows you to select based on the given attribute

[attr] – any

[attr=value] – specific

[attr~=value] - the value is one of the variant

[attr|=value] – the value should be the beginning

[attr^=value]css3 – starts

[attr$=value] css3– ends

[attr\*=value] css3– contains

* **Class Selector**

p.x{

}

* **ID Selector**

#nav{

}

* **Descendant**

div.contentem{

}

* **Child**

div.content>em{

}

* **Adjacent sibling**

h1 + p{

}

* **Attribute acronym**

div[class=note] {

}

* **Pseudo classes**
* Structural pseudo class
* root pseudo class (:root)css3
* First child pseudo class (:first-child)css3

li:first-child{

}

* :last-childcss3
* :only-childcss3
* nth-child()css3
* nth-last-child()css3
* first-of-typecss3
* last-of-typecss3
* only-of-typecss3
* nth-of-type()css3
* nth-of-last-type()css3
* :emptycss3– element with no children
* Target pseudo class
* :targetcss3
* Language pseudo class
* :lang()
* UI element pseudo class
* :enabledcss3
* :disabled css3
* :checkedcss3
* :intermediatedcss3
* Link pseudo class (Dynamic pseudo class)
* :link
* :visited
* User action pseudo class (Dynamic pseudo class)
* :hover
* :active
* :focus
* Negation pseudo class
* :not()css3
* **Pseudo elements**

::first-letter

::first-line

::before

::after

* **Declaration Block**
* **Properties**
* Visual, aural, paged, interactive
* Margins, borders, and padding
* Colors and background
* Fonts
* Text Formatting
* Lists and tables
* Positioning and dimensions
* Miscellaneous
* **Values**
* Keywords
* Numbers
* Dimension
* Length units: em ex pxin an mm pt pc

relative absolute

* Percentages
* Colors
* Strings
* URLs
* Counters
* Initial values, inherited values, important rules
* Specified, computed, used, and actual values
* **CSS Rule of Precedence**
* By origin and importance
* User Agent Important Declaration
* User Important Declaration
* Author Important Declaration
* Author Normal Declaration
* User Normal Declaration
* User Agent Normal Declaration
* By specificity
* Inline styles (1) or not (0)
* Number of ID selectors
* Number of class selectors, attribute selectors and pseudo-classes
* Number of type selectors and pseudo-elements
* By order
* **CSS Comments**/\* \*/

**CSS Preprocessors**

* Haml
* Sass - Syntactically Awesome Stylesheet
* LESS
* 960 grid system
* Bootstrap
* foundation
* materialize

**Client-side web scripting**

* Allows programs (i.e. scripts) to be downloaded from a web server and executed in the client environment (e.g. browser)
* Common client-side scripting technologies
  + JavaScript
  + ECMAScript
  + Jscript
  + VBScript
  + ActionScript