HTML Elements:

Html

Head

Title, base, link, meta, style

Body

Article, aside, max, section

Header, footer

Main (main content)

Address (content information)

div (generic divider for block elements)

h1, h2, h3, h4, h5, h6 (six 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 instances, 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

Cascading Style Sheet (CSS)

* Language used to specify the presentation aspects (e.g. layout and formatting) of structurally marked up documents.
* Developed by Hakom Wium Lie (CHSS) and Bert Bos (SSP)
* Versions:
  + CSS 1 (December 1996)
  + CSS 2.1 (June 2011)
  + CSS 3
* CSS Preprocessors, CSS Framework
  + Sass, Less, 960 Grid System, Bootstrap, Foundation, Materialize, etc.

HTML/XHTML Stylesheet

* Author styles
  + External stylesheet (recommended)
    - Uses the link element

(ref, type, href, media, title)

* + - * + Alternate stylesheet supported in firefox
  + Embedded Styles
    - Responsve design
    - For quick prototyping
  + Inline styles
* User agent styles (example default CSS 2.1 stylesheets for HTML 4)

CSS Statements

* At - Rules

@charset

@import

@media

@font-face (define and download webfonts)

@keyframes (keyframe animation)

@page

* CSS Rule Set (aka CSS Rules, Style Rules)
  + Consist of a selector, followed by a brace-enclosed declaration block

e.g. \*body is a selector

body {

display: none;

color: blue

}

CSS Selectors

* Selector
  + Structure used as a condition in a CSS rule to determine which elements in the documentation 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 sector or a universal selector
    - Cannot contain other type selectors or universal selectors
  + Group of selectors
    - Coma- separated list of selectors representing the union of all elements

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 pseudo-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 pseudo-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

//ADDED

Style sheets may have three different origins: author, user, and user agent.

* **Author.** The author specifies style sheets for a source document according to the conventions of the document language. For instance, in HTML, style sheets may be included in the document or linked externally.
* **User:** The user may be able to specify style information for a particular document. For example, the user may specify a file that contains a style sheet or the user agent may provide an interface that generates a user style sheet (or behaves as if it did).
* **User agent:** Conforming user agents must apply a default style sheet (or behave as if they did). A user agent's default style sheet should present the elements of the document language in ways that satisfy general presentation expectations for the document language (e.g., for visual browsers, the EM element in HTML is presented using an italic font). Style sheets from these three origins will overlap in scope, and they interact according to the cascade.
* Sort according to importance (normal or important) and origin (author, user, or user agent). In ascending order of precedence:
  + - 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