**WHAT IS CSS?**

**Cascading Style Sheets (CSS) is a basic instrument for including style (e.g., textual styles, hues, separating) to Web records.**

***HIGHLIGHTS: The “CSS Snapshot” (latest: 2017) lists the parts that are ready for implementers.***

**NEWS: The Future of Style**

***14 Jun 2018:***

*The CSS Days 2018 in Amsterdam are this year on 14 and 15 June (with workshops on the thirteenth). The beginning day's subject is client connection. Speakers incorporate Eric Meyer and Greg Whitworth.*

***15 Mar 2018:***

New Proposed Recommendation:[*CSS Color Level 3*](https://www.w3.org/TR/2018/PR-css-color-3-20180315/)*.*

Updated Candidate Recommendation:[*CSS Fonts Level 3*](https://www.w3.org/TR/2018/CR-css-fonts-3-20180315/)

***13 Mar 2018:***

New Working Draft: [*CSS Text Decoration Level 4*](https://www.w3.org/TR/2018/WD-css-text-decor-4-20180313/).

***4 Mar 2018:***

Updated Working Draft: [*CSS Intrinsic & Extrinsic Sizing Level 3*](https://www.w3.org/TR/2018/WD-css-sizing-3-20180304/).

***6 Feb 2018:***

New Working Draft: [*CSS Grid Layout Level 2*](https://www.w3.org/TR/2018/WD-css-grid-2-20180206/).

Visit syndicator “The Future of Style” for more news!! <https://www.w3.org/Style/CSS/Planet/>

**CSS Structure**

* ***At – Rule***
* Statement which furnishes CSS with guidelines to perform or how to carry on.
* Usually begins with ‘@’ sign followed by an identifier then semi-colon, ‘;’

[**@charset**](https://developer.mozilla.org/en-US/docs/Web/CSS/@charset) - Used by the style sheet to define a set of character.

[**@import**](https://developer.mozilla.org/en-US/docs/Web/CSS/@import) - Ask CSS engine to provide external sheet.

[**@namespace**](https://developer.mozilla.org/en-US/docs/Web/CSS/@namespace) - Tells that the XML namespace are prefixed with the content of the CSS engine.

**Nested at-rules** - settled articulations, utilized as an announcement of a template and in addition within contingent gathering rules:

[**@media**](https://developer.mozilla.org/en-US/docs/Web/CSS/@media) – rule that decide should be applied its substance if the gadget grant the specific requirements characterized utilizing a media question.

[**@supports**](https://developer.mozilla.org/en-US/docs/Web/CSS/@supports) – Apply the rule if the program achieved the required specification/condition.

[**@document**](https://developer.mozilla.org/en-US/docs/Web/CSS/@document) - A rule that applies its substance if the archive that the template is connected achieved the required specification/condition.

[**@page**](https://developer.mozilla.org/en-US/docs/Web/CSS/@page) - defines the part of the changes in design that is connected if printing the archive occurs.

[**@font-face**](https://developer.mozilla.org/en-US/docs/Web/CSS/@font-face) - Describes the part of an outer textual style to be downloaded.

[**@keyframes**](https://developer.mozilla.org/en-US/docs/Web/CSS/@keyframes) - Describes sequence of CSS animation.

[**@font-feature-values**](https://developer.mozilla.org/en-US/docs/Web/CSS/@font-feature-values) - regular names in text style variation exchanges for include enacted distinctively in OpenType.

* **Selector –** speaks to a structure, an example is the CSS rule, which figures out which components a selector coordinates in the document tree/as a level portrayal.
* Elements in the tree are coordinated by the selector called the subjects.

**Selector Syntax –** chain of at least one groupings of straightforward selectors isolated by combinators, with one pseudo-component in the final arrangement.

* ***Sequence of Simple Selectors***
* Not separated by combinators
* Starts with type selector/universal selector
* Cannot contain other type selectors or universal selectors (\* - implicit).
* The subjects are components that match every one of the selectors in the grouping.

**SIMPLE SELECTORS**

* *Universal selectors* - It refers to any single component in the document tree in several namespace including those having without a namespace without automatic/default namespace indicated for selectors.

Example:

* \*[hreflang|=en] and [hreflang|=en] are equivalent,
* *Type selectors* - is the name of a record dialect component compose composed utilizing the punctuation of CSS qualified names [CSS3NAMESPACE].

Example:

@namespace foo url(<http://www.example.com>);

foo|h1 { color: red } /\* 1st rule \*/

foo|\* { color: green } /\* 2nd rule \*/

\*|h1 { color: green }

h1 { color: green }

* ID Selectors
* global attribute (#)
* enables creators to dole out an identifier to one component case in the archive tree.
* speaks to an element occasion, the identifier is equivalent to the ID selector.

Examples:

h1#chapter1

#chapter1

\*#z98y

* Class selectors – can be associated with specific elements.

*Note: In the event elements that has different class properties, the values should linked including spaces in between values previously looking the class.*

Examples:

\*.pastoral { color: red

H1.pastoral { color: red }

* Attribute selectors: enclosed in brackets

(\*) any, (^) beginning, (~)list/exactly, ($)end, (|)starts exactly

: Selectors permit the portrayal of a component's attributes. At the point when a selector is utilized as an articulation to coordinate against a component, attribute selectors must be considered to coordinate an attribute if that attribute has a property that matches the trait spoke to by the attribute selector.

Examples:

h1[title]

span[class="example"]

span[hello="Cleveland"][goodbye="Columbus"]

a[rel~="copyright"] { ... }

a[href="http://www.w3.org/"] { ... }

* Pseudo classes

*Dynamic pseudo classes*

* + Link pseudo-classes

:link :visited

* + user action pseudo-classes

:hover :active :focus

*Target pseudo-classes*

:target (css3)

*Language pseudo-classes*

:enabled, :disabled, :checked

, :indeterminate

*Structural pseudo-classes*

:root pseudo-class, :nth-child() pseudo-class

* **Combinators**
* Descendant combinators (whitespaces, spaces, i.e.) – any level
* The whitespace, which divides two selectors in the form “A B”. This represents that element B is a descendant of element A.

Examples:

**h1 em** - *(em element is a decendant of h1 element)*

**div \* p** - *(element p is a grandchild of div elemet)*

* Sibling combinators – share the same parent
  + *Adjacent-sibling combinatory (+),* also called next-sibling combinator.

Examples:

**math + p -** *(p element following a math element)*

**h1.opener + h2 *-*** *(adds an attribute selector)*

* + *General-sibling combinatory (~),* also called subsequent-sibling combinatory.

Examples:

**h1 ~ pre –** *(pre element following an h1)*

* **COUNTERS –** used to naturally list number of things/items in CSS. They are made and controlled with the counter-increase, counter-set and counter-reset properties, and utilized with the counter() and counters() capacities. Counters have a name, a number esteem, a maker component, and potentially another counter settled inside themselves.

For more information about counters. Just visit this site!!

<https://www.w3.org/TR/css-lists-3/#counter>

* **SELECTOR GROUP** – the union of all selectors separated by comma with more than one target.

Example:

h1,h2,h3 {

color: darkred; /\* color of h1 h2 h3 should be darkred \*/

font-variant: small-caps; /\* h1 h2 h3 should be in small caps \*/

}

* **CSS RULE PRECEDENCE** – selectors of several style rules can be a subject to an HTML element
* Such that rule targets have different properties, their effects cascade (i.e., are combined)
* Such that styles involve the same property, they conflict, and must be resolved such that only the style is applied.

# Additional Readings for CSS Cascading and Inheritance Level 3!

# <https://www.w3.org/TR/css-cascade-3/#cascading>

* **OVERRIDES**

*CSS OVERRIDE AND COMPUTED STYLE SHEET*

* Resolution:
* By origin and importance
* Specificity
* Order

*The specificity of a selector can be computed by:*

* Number count of id selectors (= a)
* Number count of class selectors/attribute/pseudo classes (= b)
* number count of type selectors/pseudo elements (= c)

# For more information about calculating a selector’s specificity:

# https://www.w3.org/TR/selectors-3/#specificity