CSS Selectors

CSS Selectors are used to define a style for an element or a group of elements.

Simple Selectors

1. Type Selector

- the type selector applies styling through utilizing the name of an element. When used, styling will be applied to all instances of the element.

2. Class Selector

- the class selector uses the name specified in a class attribute. Styling using the class selector is done by placing a ‘.’ character before the class name.

3. ID Selector

- the ID selector applies style on an element based on the name specified in an ID attribute. The syntax ‘#’ followed by the ID name is followed when specifying styling using an ID name.

4. Universal Selector

- the universal selector (\*) is applicable for styling elements of any type. All elements can be selected or may be narrowed based on need.

5. Attribute Selector

- the attribute selector works for applying style based on an attribute or a specified value of an attribute.

Combinators – combinators specify relationships between selectors and are be used to be more specific with element selection.

There are three types of combinators

1. [Descendant combinator](https://developer.mozilla.org/en-US/docs/Web/CSS/Descendant_selectors) ( )

- the descendant combinator selects the descendants of the first element.

2. [Child combinator](https://developer.mozilla.org/en-US/docs/Web/CSS/Child_selectors) (>)

- the child combinator selects the direct children of the first element.

3. Sibling Combinators

[a. Adjacent sibling combinator](https://developer.mozilla.org/en-US/docs/Web/CSS/Adjacent_sibling_selectors) (+)

- the adjacent sibling combinator selects adjacent siblings of an element.

b. [General sibling combinator](https://developer.mozilla.org/en-US/docs/Web/CSS/General_sibling_selectors) (~)

- the general sibling combinator selects elements that are siblings of another specified element.

Pseudo-classes and Pseudo-elements

Elements can also be styled according to their state. Pseudo-classes allow elements to be selected based on other information about the element. An example would be styling the appearance of a button when hovered on or clicked. This uses the ‘:’ symbol. If the web designer wants to specify an element based on its position then pseudo-elements can be used. Pseudo-elements use the ‘::’ symbol and can select an element based on its position. An example would be selecting the first line of a paragraph and such.

Selector group

- multiple selectors, separated by a comma or the ‘,’ symbol, can be used to style different instances of elements.

Reference: CSS Selectors(n.d.). Retrieved April 8, 2018 from <https://developer.mozilla.org/en-US/docs/Web/CSS/CSS_Selectors>.

CSS Rule Precedence

For the reason that specified styles might clash because elements could be stylized by different selectors, the CSS rule precedence is followed. Prioritizing a style is done to resolve conflict.

Three ways that resolve conflict

1. Origin and Importance

- Styles can be prioritized based from its origins. The following precedence is followed:

1. Transition declarations
2. Important user agent declarations
3. Important user declarations
4. Important override declarations
5. Important author declarations
6. Animation declarations
7. Normal override declarations
8. Normal author declarations
9. Normal user declarations
10. Normal user agent declarations

See <https://www.w3.org/TR/css3-cascade/#cascade-origin> for the list.

2. Specificity

- The more specific the selector for a style is, the more it is prioritized.

3. Order

- Inline styles are prioritized but if an external CSS is used, the later declaration based on line number will be prioritized.