CSS Rule Precedence

In CSS an element may be subjected to multiple style properties. Style properties that are unique in each element are cascaded and properties that are the same in each element conflict.

The cascade algorithm takes an unordered list of values declared for a style property for an element, then sorts them according to the precedence of their declarations, and a single cascaded value is outputted.

By Origin and Importance

The precedence of various origins is in descending order:

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

By Specificity

When two or more selectors apply to a single element, the one with higher specificity is cascaded. To calculate for a selector’s specificity:

A = number of ID selectors in a selector

B = number of pseudo-classes, attribute, and class selectors in a selector

C = number of pseudo-elements and type selectors in a selector

Universal Selectors(\*) are ignored for calculating specificity.

The one with the most specific selector will be in effect.

Examples:

* \* A=0 B=0 C=0
* LI A=0 B=0 C=1
* OL LI A=0 B=0 C=2
* LI.class A=0 B=1 C=1
* #idselector A=1 B=0 C=0

By Order of Appearance

The last declaration of each of all elements in all documents imported or externally linked to an html document will be the style that will be rendered by a browser. The cascade will output a sorted list of declarations for every style property of every element