Name: Amr Mohamed Hggar

Specificity in CSS.

Q) What does Specificity mean ?

A)

Specificity is the algorithm used by browsers to determine the CSS declaration that is the most relevant to an element, which in turn, determines the property value to apply to the element. The specificity algorithm calculates the weight of a CSS selector to determine which rule from competing CSS declarations gets applied to an element.

Note: Browsers consider specificity after determining cascade origin and importance. In other words, for competing property declarations, specificity is relevant and compared only between selectors from the one cascade origin and layer that has precedence for the property. Order of appearance becomes relevant when the selector specificities of the competing declarations in the cascade layer with precedence are equal.

Q) How is specificity calculated?

A)

Specificity is an algorithm that calculates the weight that is applied to a given CSS declaration. The weight is determined by the number of selectors of each weight category in the selector matching the element (or pseudo-element). If there are two or more declarations

providing different property values for the same element, the declaration value in the style block having the matching selector with the greatest algorithmic weight gets applied.
The specificity algorithm is basically a three-column value of three categories or weights - ID, CLASS, and TYPE - corresponding to the three types of selectors. The value represents the count of selector components in each weight category and is written as *ID - CLASS - TYPE*. The three columns are created by counting the number of selector components for each selector weight category in the selectors that match the element.

Selector weight categories:

The selector weight categories are listed here in the order of decreasing specificity:

ID column
Includes only ID selectors, such as #example. For each ID in a matching selector, add 1-0-0 to the weight value.

CLASS column
Includes class selectors, such as .myClass, attribute selectors like [type="radio"] and [lang|="fr"], and pseudo-classes, such as :hover, :nth-of-type(3n), and :required. For each class, attribute selector, or pseudo-class in a matching selector, add 0-1-0 to the weight value.

TYPE column
Includes type selectors, such as p, h1, and td, and pseudo-elements like ::before, ::placeholder, and all other selectors with double-colon notation. For each type or pseudo-element in a matching selector, add 0-0-1 to the weight value.
No value

The universal selector (*) and the pseudo-class :where() and its parameters aren't counted when calculating the weight, but they do match elements. The value for both the universal selector and the pseudo-class is 0-0-0; these selectors do not impact the specificity weight value.

Combinators, such as +, >, ~, " ", and ||, may make a selector more specific in what is selected but they don't add any value to the specificity weight. The negation pseudo-class, :not(), itself has no weight. Neither does the :is() pseudo-class. The parameters in these selectors, however, do. The values of both come from the parameter in the list of parameters that has the highest specificity. The :not() and :is() exceptions are discussed below.