**learning Objective**

* Master the concept and calculation method of selector specificity in CSS
* Master the method to improve the priority of CSS attributes
* Understand how styles specifically work on the page
* Master the initial values of common CSS attributes

**Content**

* For example, when a different CSS selector has selected an element, which rule will take effect?
* ***<!-- HTML -->***
* <article>
* <h1 class="title">**I am a title**</h1>
* </article>
* ***<!-- CSS -->***
* <style>
* **.title {**
* Color**: blue;**
* **}**
* **Article h1 {**
* Color**: red;**
* **}**
* </style>
* CSS solves style conflicts by specificity. The concept of specificity is that when two or more CSS selectors with the same importance have the declaration code applied to the same element to change the same attribute, then the declaration code of selector with the highest specificity will be applied to the element first. An example of CSS specificity is shown as follows:

| **Selectors** | **Inline** | **ID number** | **Pseudo-classes** | **Label number** | **Specificity** |
| --- | --- | --- | --- | --- | --- |
| #nav .list li a:link | 0 | 1 | 2 | 2 | 0122 |
| .hd ul .links a | 0 | 0 | 2 | 2 | 0022 |

* The specificity level of the simple selector: (the higher the level, the higher the priority)
  + Level 0:\*
  + Level 1: label selector, pseudo-element
  + Level 2: class, pseudo-class, pseudo-element
  + Level 3: id
  + Level 4: Inline
* Style coverage rules:
  + Prioritize according to the introduction method: the sequence of priority from high to low refers to: "inline attribute" -> "embed" -> link"
  + Attributes written at the same level override the prior one: that is, the higher the code line number on the document is, the higher priority will be
  + Add "!important" style with the highest priority: whichever the case is, sp long as the important mode is added to the style, such style has the highest priority.
* The source of CSS styles mainly consists of the following three aspects:
  + Page developer: the CSS code written by the developer in the program
  + User settings: The browser can specify a local CSS document that is automatically loaded when opening all pages
  + Browser presets: each browser has a default style for the tag inside
* So which statement will take effect specifically? The rules are as follows:
  + Find all declarations that this attribute can match
  + According to the source of the rule, the priority is arrayed from low to high:
    - Browser presets
    - User settings
    - Page developer
    - Web style with !important
    - User settings style with !important
  + In the same source, sorted by specificity, the higher the specificity, the higher the priority
  + When the specificity is same, follow the style writing order, and the subsequent one comes with higher priority
* Initial value. In CSS, each attribute has an initial value. The initial value of background-color is transparent, and the initial value of margin-left is 0. Of course, we can also explicitly reset the changed value to Initial value, such as: background-color: initial

**Recommendation**

* Cascading and Inheritance (<https://developer.mozilla.org/en-us/docs/Web/Guide/CSS/Getting_started/Cascading_and_inheritance>)
* CSS Cascading and Inheritance (<http://www.cnblogs.com/yinzixin/archive/2010/03/11/1683728.html>)
* CSS Cascading and Inheritance (<https://zhuanlan.zhihu.com/p/27536190>)
* CSS Inheritance and Cascade (<https://webplatform.github.io/docs/tutorials/inheritance_and_cascade/>)