



# CSS Specificity

[< Previous](#)[Next >](#)

## What is Specificity?

If there are two or more conflicting CSS rules that point to the same element, the browser follows some rules to determine which one is most specific and therefore wins out.

Think of specificity as a score/rank that determines which style declarations are ultimately applied to an element.

The universal selector (\*) has low specificity, while ID selectors are highly specific!

**Note:** Specificity is a common reason why your CSS-rules don't apply to some elements, although you think they should.

## Specificity Hierarchy

Every selector has its place in the specificity hierarchy. There are four categories which define the specificity level of a selector:

**Inline styles** - An inline style is attached directly to the element to be styled. Example: `<h1 style="color: #ffffff;">.`

**IDs** - An ID is a unique identifier for the page elements, such as #navbar.

**Classes, attributes and pseudo-classes** - This category includes .classes, [attributes] and pseudo-classes such as :hover, :focus etc.

**Elements and pseudo-elements** - This category includes element names and pseudo-elements, such as h1, div, :before and :after.

---

## How to Calculate Specificity?

Memorize how to calculate specificity!

Start at 0, add 1000 for style attribute, add 100 for each ID, add 10 for each attribute, class or pseudo-class, add 1 for each element name or pseudo-element.

Consider these three code fragments:

### Example

A: h1

B: #content h1

C: <div id="content"><h1 style="color: #ffffff">Heading</h1></div>

The specificity of A is 1 (one element)

The specificity of B is 101 (one ID reference and one element)

The specificity of C is 1000 (inline styling)

Since  $1 < 101 < 1000$ , the third rule (C) has a greater level of specificity, and therefore will be applied.

---

## Specificity Rules

**Equal specificity: the latest rule counts** - If the same rule is written twice into the external style sheet, then the lower rule in the style sheet is closer to the element to be styled, and therefore will be applied:

## Example

```
h1 {background-color: yellow;}  
h1 {background-color: red;}
```

Try it Yourself »

the latter rule is always applied.

---

**ID selectors have a higher specificity than attribute selectors** - Look at the following three code lines:

## Example

```
div#a {background-color: green;}  
#a {background-color: yellow;}  
div[id=a] {background-color: blue;}
```

Try it Yourself »

the first rule is more specific than the other two, and will be applied.

---

**Contextual selectors are more specific than a single element selector** - The embedded style sheet is closer to the element to be styled. So in the following situation

## Example

```
From external CSS file:  
#content h1 {background-color: red;}
```

In HTML file:

```
<style>
#content h1 {
  background-color: yellow;
}
</style>
```

the latter rule will be applied.

---

**A class selector beats any number of element selectors** - a class selector such as .intro beats h1, p, div, etc:

## Example

```
.intro {background-color: yellow;}
h1 {background-color: red;}
```

Try it Yourself »

---

**The universal selector and inherited values have a specificity of 0** - \*, body \* and similar have a zero specificity. Inherited values also have a specificity of 0.

[< Previous](#)

[Next >](#)

COLOR PICKER



SHOP

HOW TO

Tabs  
Dropdowns

Accordions  
Side Navigation  
Top Navigation  
Modal Boxes  
Progress Bars  
Parallax  
Login Form  
HTML Includes  
Google Maps  
Range Sliders  
Tooltips  
Slideshow  
Filter List  
Sort List

SHARE



CERTIFICATES

HTML  
CSS  
JavaScript  
SQL  
Python  
PHP  
jQuery  
Bootstrap  
XML

[Read More »](#)



---

[REPORT ERROR](#)

[FORUM](#)

[ABOUT](#)

[SHOP](#)

---

## Top Tutorials

[HTML Tutorial](#)  
[CSS Tutorial](#)  
[JavaScript Tutorial](#)  
[How To Tutorial](#)  
[SQL Tutorial](#)  
[Python Tutorial](#)  
[W3.CSS Tutorial](#)  
[Bootstrap Tutorial](#)  
[PHP Tutorial](#)  
[jQuery Tutorial](#)  
[Java Tutorial](#)  
[C++ Tutorial](#)

## Top References

[HTML Reference](#)  
[CSS Reference](#)  
[JavaScript Reference](#)  
[SQL Reference](#)  
[Python Reference](#)  
[W3.CSS Reference](#)  
[Bootstrap Reference](#)  
[PHP Reference](#)  
[HTML Colors](#)



[jQuery Reference](#)[Java Reference](#)[Angular Reference](#)

## Top Examples

[HTML Examples](#)[CSS Examples](#)[JavaScript Examples](#)[How To Examples](#)[SQL Examples](#)[Python Examples](#)[W3.CSS Examples](#)[Bootstrap Examples](#)[PHP Examples](#)[jQuery Examples](#)[Java Examples](#)[XML Examples](#)

## Web Certificates

[HTML Certificate](#)[CSS Certificate](#)[JavaScript Certificate](#)[SQL Certificate](#)[Python Certificate](#)[jQuery Certificate](#)[PHP Certificate](#)[Bootstrap Certificate](#)[XML Certificate](#)[Get Certified »](#)

---

W3Schools is optimized for learning, testing, and training. Examples might be simplified to improve reading and basic understanding. Tutorials, references, and examples are constantly reviewed to avoid errors, but we cannot warrant full correctness of all content. While using this site, you agree to have read and accepted our terms of use, cookie and privacy policy. Copyright 1999-2020 by Refsnes Data. All Rights Reserved.

Powered by W3.CSS.

