



# New CSS selectors: :is and :where

A pair of new pseudo-class selectors have made it easier to write CSS in a way that is easier to read and to reason through.

:is and :where allow you to combine selectors that have the same styles. For example, to identify the headers inside an article you'd normally do this:

```
article, h1,  
article, h2,  
article, h3,  
article, h4,  
article, h5,  
article, h6 {  
  background: black;  
  color: white;  
}
```

Instead, you could use :is and improve legibility while avoiding a long selector and potential typos associated with it:

```
article :is(h1, h2, h3, h4, h5, h6) {  
  background: black;  
  color: white;  
}
```

When writing CSS, :is (and its cousin :where) give you a way to group elements together anywhere in a selector.

Here are a few more examples of :is and :where building selectors for a hypothetical article:

```
/*  
  All the headers in an article
```

```

*/
article :is(h1,h2,h3,h4,h5,h6) {
  background-color: black;
  color: white;
  padding: 1em 1em 1em 0;
}

/*
  all the paragraphs in an article
  header or footer
*/
article :is(header, footer) > p {
  color: darkgrey
}

/*
  All paragraphs inside blockquotes
  and pullquotes
*/
article :where(blockquote, .pull-quote) > p {
  font-size: 1.25em;
}

/*
  h1 or h2 elements
  with class header or boldest
*/
article :is(h1,h2,:is(.header,.boldest)) {
  font-weight: 900;
}

```

## Specificity

While `:is` and `:where` are almost identical they have one key difference: how they handle specificity.

`:is` will take the highest specificity from among their children. The example below will take the specificity of the ID element rather than the class or a

combination of both.

```
/*
  All paragraphs inside .header or #header
  elements.
*/
article :is(#header, .header) > p {
  font-size: 1.5em;
}
```

:where will remove all the specificity from their children. The whole selector will have no specificity.

This example removes the specificity from the selector below, it works as long as there are no other rules targeting the same selector.

```
/*
  All paragraphs inside .header or #header
  elements.
*/
article :where(#header, .header) > p {
  font-size: 1.5em;
}
```

#### TL,DR:

- :is() takes the specificity of the child with the highest specificity in the selector list passed as parameter
- :where() has no specificity. It removes all the specificity in the selector list passed as parameter

## Browser compatibility

:is and :where are supported in Chromium (88 and later), Firefox (78 and later) and Safari (14 and later). Some older browser versions support the :is() selector as :matches() or -webkit-any().