RED space™

TECH SHARE

Modern CSS Selectors

I can :has() parent?



Hello! 👋

I’m Luke DeWitt

Director of Web Services

RED space

TECH SHARE



I love CSS

RED space

TECH SHARE



RED space

TECH SHARE



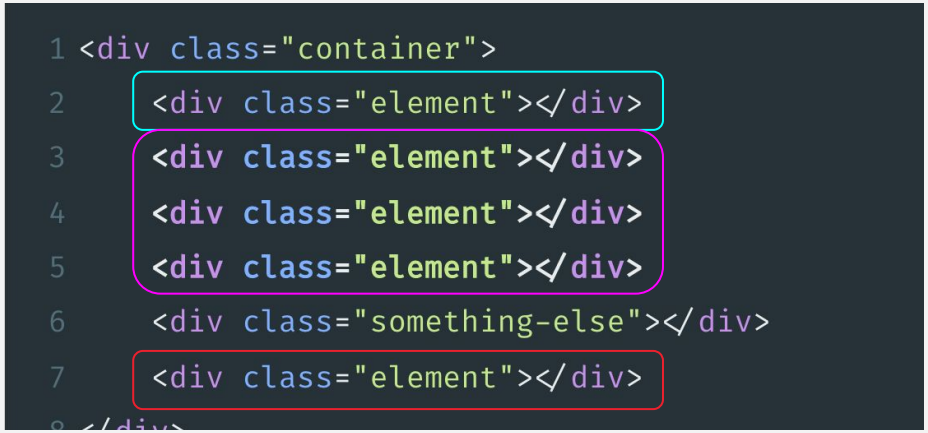
I love CSS

Favorite CSS Selectors

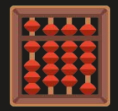
(yes, like children, you can have favorites)

- **Direct Sibling (Adjacent Sibling Combinator)**
  - The adjacent sibling combinator (+) separates two selectors and matches the second element only if it immediately follows the first element, and both are children of the same parent element

.element + .element



MDN Link: https://developer.mozilla.org/en-US/docs/Web/CSS/Adjacent_sibling_combinator



I was told there’d be no math

Favorite CSS Selectors

- **Indirect Sibling (General Sibling Combinator)**
  - The general sibling combinator (~) separates two selectors and matches all iterations of the second element, that are following the first element (though not necessarily immediately), and are children of the same parent element.

```
<div class="container">
    <div class="element"></div>
    <div class="element"></div>
    <div class="element"></div>
    <div class="element"></div>
    <div class="something-else"></div>
    <div class="element"></div>
</div>
```

.element ~ .element

MDN Link: https://developer.mozilla.org/en-US/docs/Web/CSS/General_sibling_combinator



Some new favorites, perhaps?

:where()

“takes a selector list as its argument, and selects any element that can be selected by one of the selectors in that list”

```css
.container:where(h3, .element) {
  color: red;
}
```

```html
<div class="container">
  <h3>Header Text</h3>
  <div class="element"> ... </div>
  <div class="el2"> ... </div>
  <div class="element"> ... </div>
</div>
```

MDN Link: https://developer.mozilla.org/en-US/docs/Web/CSS/:where

:where()

```css
.container:where(h3, .element) {
  color: red;
}
```



```css
.container h3,
.container .element {
  color: red
}
```

```html
<div class="container">
  <h3>Header Text</h3>
  <div class="element"> ... </div>
  <div class="el2"> ... </div>
  <div class="element"> ... </div>
</div>
```

:where()

Let's get SASSy

```
.container:where(h3, .element) {
  color: red;
}
```



```
.container {
  h3, .element {
    color: red;
  }
}
```

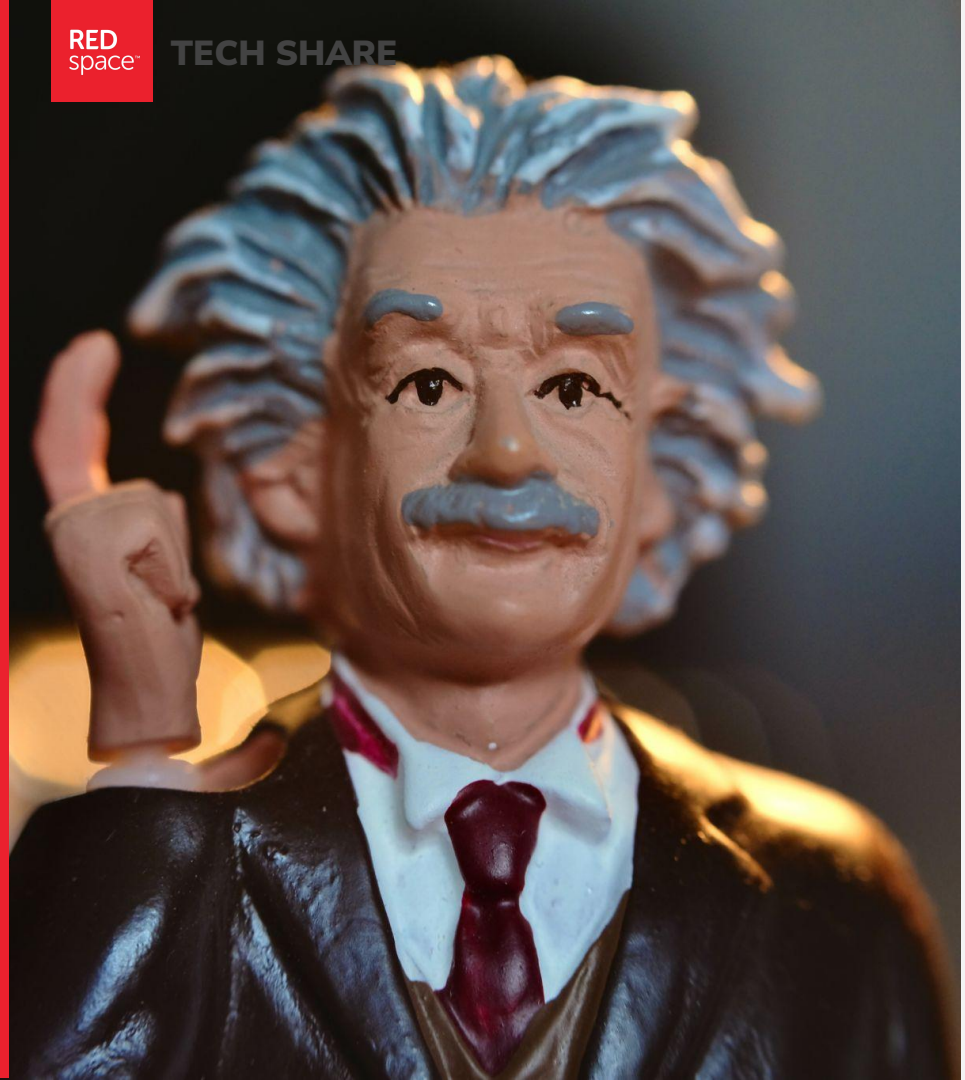
:is()

“takes a selector list as its argument, and selects any element that can be selected by one of the selectors in that list”

```css
.container:is(h3, .element) {
  color: red;
}
```

```html
<div class="container">
  <h3>Header Text</h3>
  <div class="element"> ... </div>
  <div class="el2"> ... </div>
  <div class="element"> ... </div>
</div>
```

MDN Link: https://developer.mozilla.org/en-US/docs/Web/CSS/:is



... wait a second...

:where() :is() this going?

"takes a selector list as its argument, and selects any element that can be selected by one of the selectors in that list"

```
.container:where(h3, .element) {
  color: red;
}
```

```
.container:is(h3, .element) {
  color: red;
}
```


Listing the damn selectors
:where( )
:is()



Key differences

Listing Selectors vs Modern Selectors

- More compact

- Forgiving

Forgiving Selectors

```
1 .container 3h,
2 .container .element {
3   color: red
4 }
```

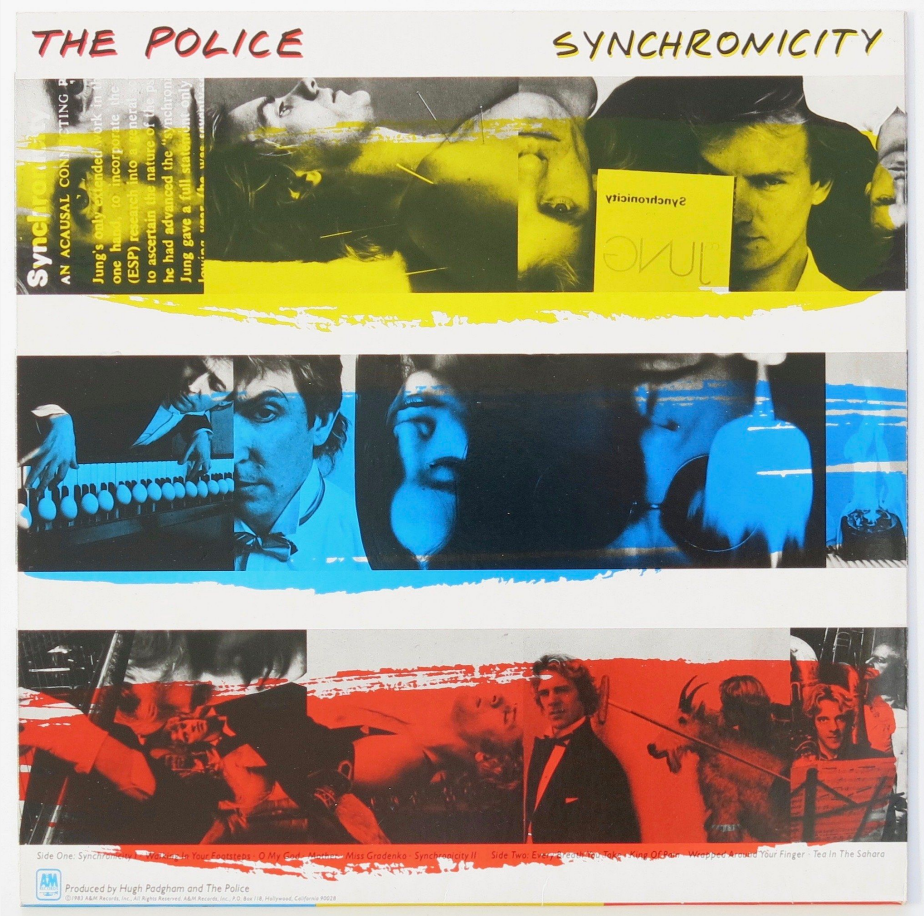
```
1 .container:where(3h, .element) {
2   color: red
3 }
```

```
1 <div class="container">
2   <h3>Header Text</h3>
3   <div class="element"> ... </div>
4   <div class="el2"> ... </div>
5   <div class="element"> ... </div>
6 </div>
```

RED space

TECH SHARE




THE POLICE
SYNCHRONICITY
Synchronicity
JUNG
Produced by Hugh Padgham and The Police

Specificity





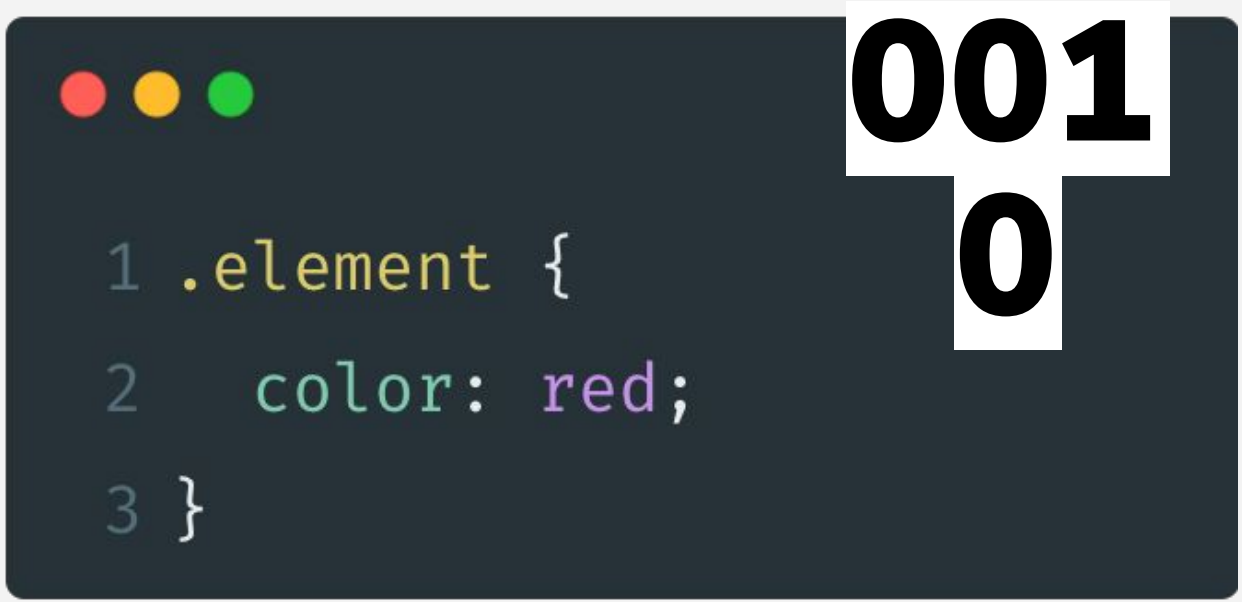
A quick refresher on specificity

Specificity

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.



Guess the Specificity



Guess the Specificity



Guess the Specificity



Guess the Specificity



Guess the Specificity



Specificity

```html
<div class="container" id="wrapper">
  <div class="element" id="fancy-element">
    Hello, World
  </div>
</div>
```

Specificity

```css
.container {
  color: red;
}

.container .
  color: blue;
}

#wrapper .el
  color: black;
}
```

0010

0020

0110

```html
<div class="container" id="wrapper">
  <div class="element" id="fancy-element">
    Hello, World
  </div>
</div>
```

Specificity

```
1 #wrapper {
2   color: red;
3 }
4
5 .container {
6   color: blue;
7 }
8
9 .element {
10  color: black;
11 }
```

0100

0011

0010

... on the parent element

```
1 <div class="container" id="wrapper">
2   <div class="element" id="fancy-element">
3     Hello, World
4   </div>
5 </div>
```

Specificity

```css
.container:where(h3, .element) {
  color: red;
}
```

Specificity of the **most** specific selector in the argument list, in this case, 10

```css
.container:is(h3, .element) {
  color: red;
}
```

Specificity of the **0**, regardless of the argument list

:where()

Browser compatibility

Report problems with this compatibility data on GitHub

	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
`:where()`	✓ 88	✓ 88	✓ 78	✓ 74	✓ 14	✓ 88	✓ 79	✓ 63	✓ 14	✓ 15.0	✓ 88
Support for forgiving selector list	✓ 88	✓ 88	✓ 82	✓ 74	✓ 14	✓ 88	✓ 82	✓ 63	✓ 14	✓ 15.0	✓ 88

:is()

Browser compatibility

Report problems with this compatibility data on GitHub

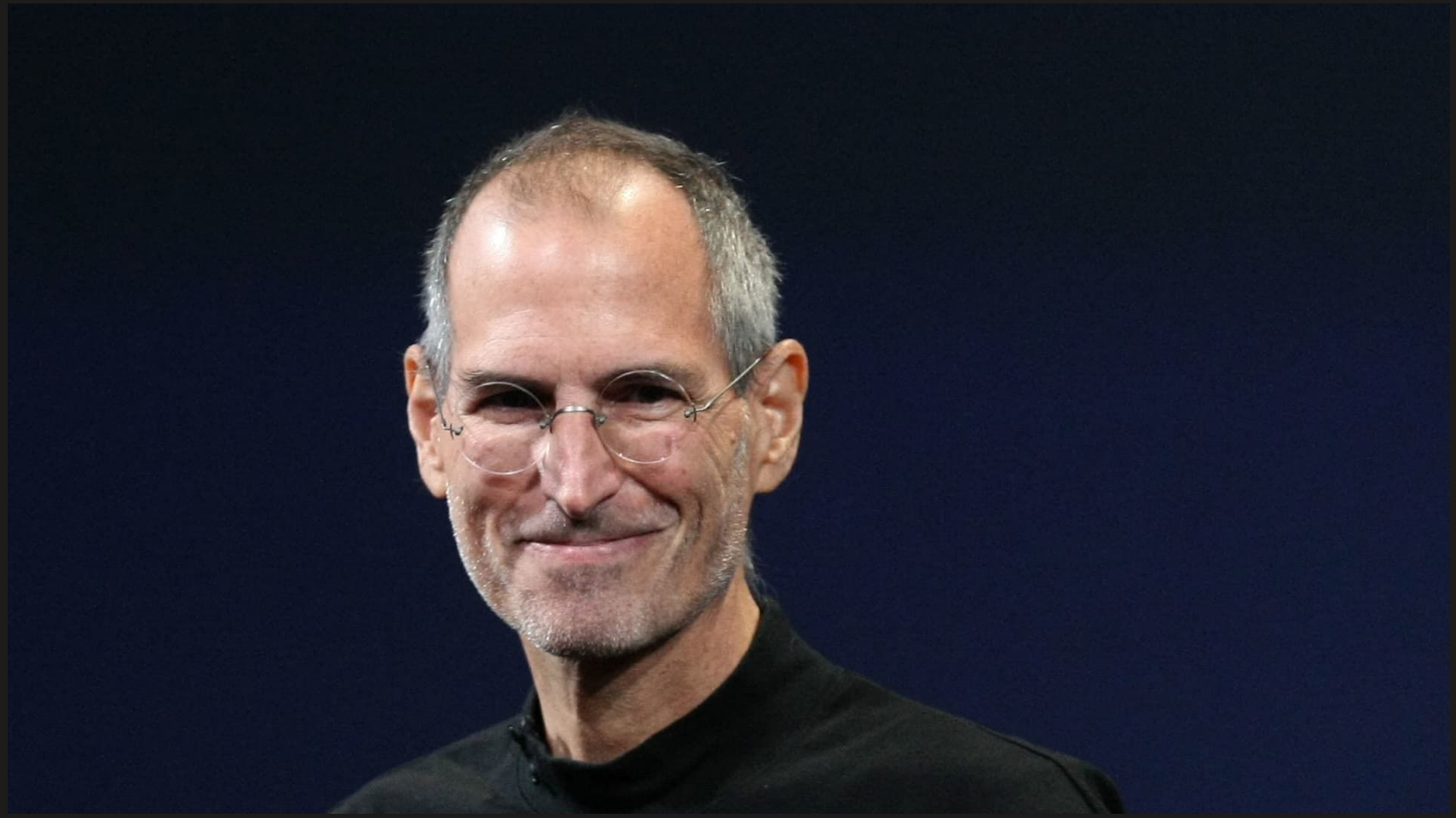
	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
`:is()`	✓ 88 …	✓ 88 …	✓ 78 …	✓ 74 …	✓ 14 …	✓ 88 …	✓ 79 …	✓ 63 …	✓ 14 …	✓ 15.0 …	✓ 88 …
Support for forgiving selector list	✓ 88	✓ 88	✓ 82	✓ 74	✓ 14	✓ 88	✓ 82	✓ 63	✓ 14	✓ 15.0	✓ 88



5/10

RED space

TECH SHARE



:has()

“represents an element if any of the relative selectors that are passed as an argument match at least one element when anchored against this element”

WE’RE TALKIN’ PARENT SELECTORS, BABY!!!!

:has()

the future has arrived!!

```css
article:has(h2) {
  color: red;
}
```

```html
<section>
  <article>
    <h1>Morning Times</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
  </article>
  <article>
    <h1>Morning Times</h1>
    <h2>Delivering you news every morning</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
  </article>
</section>
```

MDN Link: https://developer.mozilla.org/en-US/docs/Web/CSS/:has

:has()

the future has arrived!!

```css
article:has(h2) h1 {
  color: red;
  margin-bottom: 0;
}
```

```html
<section>
  <article>
    <h1>Morning Times</h1>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
  </article>
  <article>
    <h1>Morning Times</h1>
    <h2>Delivering you news every morning</h2>
    <p>Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.</p>
  </article>
</section>
```

RED space

TECH SHARE

... tip of the iceberg

I can :has() CSS?

```
body:has(video, audio) {
  background: red;
}
```

```
body:has(video):has(audio) {
  background: blue;
}
```

```
:is(h1, h2, h3):has(+ h2, + h3, + h4) {
  margin: 0 0 0.25rem 0;
}
```



RED space

TECH SHARE

:has()

Browser compatibility

Report problems with this compatibility data on GitHub

	Chrome	Edge	Firefox	Opera	Safari	Chrome Android	Firefox for Android	Opera Android	Safari on iOS	Samsung Internet	WebView Android
:has()	✓ 105	✓ 105	✗ 103	✓ 91	✓ 15.4	✓ 105	✗ No	✓ 72	✓ 15.4	✗ No	✓ 105



9/10



Go forth and style!