CSS ALMANAC → **SELECTORS** → **N** →

Sara Cope   Sep 6, 2011   Sep 10, 2020

The `:nth-child` selector allows you to select one or more elements based on their source order, according to a formula.

```css
/* Select the first list item */
li:nth-child(1) { }

/* Select the 5th list item */
li:nth-child(5) { }

/* Select every other list item starting with first */
li:nth-child(odd) { }

/* Select every 3rd list item starting with first */
li:nth-child(3n - 2) { }

/* Select every 3rd list item starting with 2nd */
li:nth-child(3n - 1) { }

/* Select every 3rd child item, as long as it has class "el" */
.el:nth-child(3n) { }
```

It is defined in the CSS Selectors Level 3 spec (http://www.w3.org/TR/selectors/) as a "structural pseudo-class", meaning it is used to style content based on its relationship with parent and sibling elements.

Suppose we are building a CSS grid, and want to remove the margin on every fourth grid module. Here's that HTML:

```html
<section class="grid">
  <article class="module">One</article>
  <article class="module">Two</article>
  <article class="module">Three</article>
  <article class="module">Four</article>
  <article class="module">Five</article>
</section>
```

Rather than adding a class to every fourth item (e.g. `.last`), we can use `:nth-child`:

```css
.module:nth-child(4n) {
  margin-right: 0;
}
```

The `:nth-child` selector takes an argument: this can be a single integer, the keywords `even`, `odd`, or a formula. If an integer is specified only one element is selected—but the keywords or a formula will iterate through all the children of the parent element and select matching elements — similar to navigating items in a JavaScript array. Keywords "even" and "odd" are straightforward (2, 4, 6, etc. or 1, 3, 5 respectively). The formula is constructed using the syntax `an+b`, where:

- "a" is an integer value
- "n" is the literal letter "n"
- "+" is an operator and may be either "+" or "-"
- "b" is an integer and is required if an operator is included in the formula

It is important to note that this formula is an equation, and iterates through each sibling element, determining which will be selected. The "n" part of the formula, if included, represents a set of increasing positive integers (just like iterating through an array). In our above example, we selected every fourth element with the formula 4n, which worked because every time an element was checked, "n" increased by one (4×0, 4×1, 4×2, 4×3, etc). If an element's order matches the result of the equation, it gets selected (4, 8, 12, etc). For a more in-depth explanation of the math involved, please read this article (https://css-tricks.com/how-nth-child-works/) .

To illustrate further, here are some examples of valid `:nth-child` selectors:

Embedded Pen Here

Luckily, you don't always have to do the math yourself—there are several `:nth-child` testers and generators out there:

- CSS-Tricks Tester (https://css-tricks.com/examples/nth-child-tester/#)
- Lea Verou's Tester (http://lea.verou.me/demos/nth.html)

(#aa-nth-childan-b-of-selector) :nth-child(an + b of <selector>)

There is a little-known filter that can be added to `:nth-child` according to the CSS Selectors (https://www.w3.org/TR/selectors-4/#the-nth-child-pseudo) specification: The ability to select the `:nth-child` of a subset of elements, using the `of` format. Suppose you have a list of mixed content: Some have the class `.video`, some have the class `.picture`, and you want to select the first 3 pictures. You could do so with the "of" filter like so:

```css
:nth-child(-n+3 of .picture) {
  /*
     Selects the first 3 elements
     applied not to ALL children but
     only to those matching .picture
  */
}
```

Note that this is distinct from appending a selector directly to the `:nth-child` selector:

```css
.picture:nth-child(-n+3) {
  /*
     Not the same!
     This applies to elements matching .picture
     which _also_ match :nth-child(-n+3)
  */
}
```

This can get a little confusing, so an example might help illustrate the difference:

Embedded Pen Here

Browser support for the "of" filter is very limited: As of this writing, only Safari supported the syntax. To check the status of your favorite browser, here are open issues related to `:nth-child(an+b of s)`:

- Firefox: Support for nth-child(An+B of sel) (https://bugzilla.mozilla.org/show_bug.cgi?id=854148)

- Chrome: Implement :nth-child(an+b of S) (https://bugs.chromium.org/p/chromium/issues/detail?id=304163&q=nth-child&colspec=ID%20Pri%20M%20Stars%20ReleaseBlock%20Component%20Status%20Owner%20Summary%20OS%20Modified)

↻ (#aa-points-of-interest) Points of Interest

- `:nth-child` iterates through elements starting from the top of the source order. The only difference between it and `:nth-last-child` (https://css-tricks.com/almanac/selectors/n/nth-last-child) is that the latter iterates through elements *starting from the bottom of the source order.*

- The syntax for selecting the *first* n number of elements is a bit counter-intuitive. You start with `-n`, plus the positive number of elements you want to select. For example, `li:nth-child(-n+3)` will select the first 3 `li` elements.

- The `:nth-child` selector is very similar to `:nth-of-type` (https://css-tricks.com/almanac/selectors/n/nth-of-type) but with one **critical difference:** it is less specific (https://css-tricks.com/the-difference-between-nth-child-and-nth-of-type/). In our example above they would produce the same result because we are iterating over only

`.module` elements, but if we were iterating over a more complex group of siblings, `:nth-child` would try to match all siblings, not only siblings of the same element type. This reveals the power of `:nth-child`—it can select any sibling element in an arrangement, *not only elements that are specified before the colon.*

↻ (#aa-related-properties) Related Properties

- nth-last-child (https://css-tricks.com/almanac/selectors/n/nth-last-child)
- nth-of-type (https://css-tricks.com/almanac/selectors/n/nth-of-type)
- nth-last-of-type (https://css-tricks.com/almanac/selectors/n/nth-last-of-type)
- first-of-type (https://css-tricks.com/almanac/selectors/f/first-of-type)
- last-of-type (https://css-tricks.com/almanac/selectors/l/last-of-type)

↻ (#aa-other-resources) Other Resources

- Useful :nth-child Recipes (https://css-tricks.com/useful-nth-child-recipies/)
- Mozilla Docs (https://developer.mozilla.org/en-US/docs/CSS/:nth-child)
- QuirksMode article (http://quirksmode.org/css/selectors/nthchild.html)
- Sitepoint article (http://reference.sitepoint.com/css/understandingnthchildexpressions)

↻ (#aa-browser-support) Browser Support

| Chrome: Any | Safari: 3.2+ | Firefox: Any | Opera: 9.5+ | IE: 9+ | Android: Any | iOS: Any |

:nth-child was introduced in CSS Selectors Module 3, which means old versions of browsers do not support it. However, modern browser support is impeccable, and the new pseudo-selectors are widely used in production environments. If you require older browser support, either polyfill for IE (https://css-tricks.com/how-to-use-the-webshims-polyfill/) , or use these selectors in non-critical ways á la progressive enhancement, which is recommended.

PROPERTIES

> A
> B
> C
> D
> E
> F
> G
> H
> I
> J
> L
> M
> O
> P
> Q
> R
> S
> T
> U
> V
> W
> Z

SELECTORS

> A
> B
> C
> D
> E
> F
> G
> H
> I
> L
> M
> N
> O
> P
> R
> S
> T
> U
> V
> W