✦ Bootcamps   🎁 Spaces   Sign Up   Log in

HTML   CSS   JAVASCRIPT   ◑   🌐   🔍

CSS Counters

‹ Previous   Next ›

1 Pizza

2 Hamburger

3 Hotdogs

CSS counters are "variables" maintained by CSS whose values can be incremented by CSS rules (to track how many times they are used). Counters let you adjust the appearance of content based on its placement in the document.

Automatic Numbering With Counters

CSS counters are like "variables". The variable values can be incremented by CSS rules (which will track how many times they are used).

To work with CSS counters we will use the following properties:

- `counter-reset` - Creates or resets a counter
- `counter-increment` - Increments a counter value
- `content` - Inserts generated content
- `counter()` or `counters()` function - Adds the value of a counter to an element

To use a CSS counter, it must first be created with `counter-reset` .

The following example creates a counter for the page (in the body selector), then increments the counter value for each <h2> element and adds "Section <*value of the*

Example

```css
body {
  counter-reset: section;
}

h2::before {
  counter-increment: section;
  content: "Section " counter(section) ": ";
}
```

Try it Yourself »
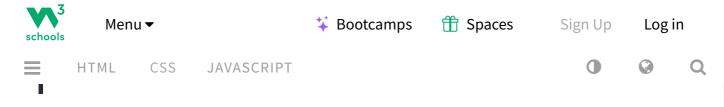
Nesting Counters

The following example creates one counter for the page (section) and one counter for each <h1> element (subsection). The "section" counter will be counted for each <h1> element with "Section *<value of the section counter>*.", and the "subsection" counter will be counted for each <h2> element with "*<value of the section counter>*.*<value of the subsection counter>*":

Example

```css
body {
  counter-reset: section;
}

h1 {
  counter-reset: subsection;
}

h1::before {
  counter-increment: section;
  content: "Section " counter(section) ". ";
}
```

✨ Bootcamps   🎁 Spaces   Sign Up   Log in

≡   HTML   CSS   JAVASCRIPT   ◐   🌐   🔍

■

Try it Yourself »

A counter can also be useful to make outlined lists because a new instance of a counter is automatically created in child elements. Here we use the `counters()` function to insert a string between different levels of nested counters:

Example

```
ol {
  counter-reset: section;
  list-style-type: none;
}

li::before {
  counter-increment: section;
  content: counters(section,".") " ";
}
```

Try it Yourself »

CSS Counter Properties

Property	Description
content	Used with the ::before and ::after pseudo-elements, to insert generated content
counter-increment	Increments one or more counter values
counter-reset	Creates or resets one or more counters
counter()	Returns the current value of the named counter

w3schools

Menu ▼

✦ Bootcamps

🎁 Spaces

Sign Up

Log in

HTML   CSS   JAVASCRIPT

Build your career today!

Get **lifelong** access to all our **courses** and **certifications**

Get Full Access

COLOR PICKER

Spaces

Upgrade

Newsletter

Get Certified

Report Error

Top Tutorials

Top References

Top Examples

Get Certified

terms of use, cookie and privacy policy.