

Selector	Example	Example description
<i>.class</i>	<i>.intro</i>	Selects all elements with class="intro"
<i>#id</i>	<i>#firstname</i>	Selects the element with id="firstname"
<i>*</i>	<i>*</i>	Selects all elements
<i>element</i>	<i>p</i>	Selects all <p> elements
<i>element,element</i>	<i>div, p</i>	Selects all <div> elements and all <p> elements
<i>element element</i>	<i>div p</i>	Selects all <p> elements inside <div> elements
<i>element>element</i>	<i>div > p</i>	Selects all <p> elements where the parent is a <div> element
<i>element+element</i>	<i>div + p</i>	Selects all <p> elements that are placed immediately after <div> elements
<i>element1~element2</i>	<i>p ~ ul</i>	Selects every element that are preceded by a <p> element
<i>[attribute]</i>	<i>[target]</i>	Selects all elements with a target attribute
<i>[attribute=value]</i>	<i>[target=_blank]</i>	Selects all elements with target="_blank"
<i>[attribute~=value]</i>	<i>[title~=flower]</i>	Selects all elements with a title attribute containing the word "flower"
<i>[attribute =value]</i>	<i>[lang =en]</i>	Selects all elements with a lang attribute value starting with "en"
<i>[attribute^=value]</i>	<i>a[href^="https"]</i>	Selects every <a> element whose href attribute value begins with "https"
<i>[attribute\$=value]</i>	<i>a[href\$=".pdf"]</i>	Selects every <a> element whose href attribute value ends with ".pdf"
<i>[attribute*=value]</i>	<i>a[href*="w3schools"]</i>	Selects every <a> element whose href attribute value contains the substring "w3schools"
<i>:active</i>	<i>a:active</i>	Selects the active link
<i>::after</i>	<i>p::after</i>	Insert something after the content of each <p> element
<i>::before</i>	<i>p::before</i>	Insert something before the content of each <p> element
<i>:checked</i>	<i>input:checked</i>	Selects every checked <input> element
<i>:disabled</i>	<i>input:disabled</i>	Selects every disabled <input> element
<i>:empty</i>	<i>p:empty</i>	Selects every <p> element that has no children (including text nodes)
<i>:enabled</i>	<i>input:enabled</i>	Selects every enabled <input> element
<i>:first-child</i>	<i>p:first-child</i>	Selects every <p> element that is the first child of its parent
<i>::first-letter</i>	<i>p::first-letter</i>	Selects the first letter of every <p> element
<i>::first-line</i>	<i>p::first-line</i>	Selects the first line of every <p> element
<i>:first-of-type</i>	<i>p:first-of-type</i>	Selects every <p> element that is the first <p> element of its parent

:focus	input:focus	Selects the input element which has focus
:hover	a:hover	Selects links on mouse over
:in-range	input:in-range	Selects input elements with a value within a specified range
:invalid	input:invalid	Selects all input elements with an invalid value
:lang(<i>language</i>)	p:lang(it)	Selects every <p> element with a lang attribute equal to "it" (Italian)
:last-child	p:last-child	Selects every <p> element that is the last child of its parent
:last-of-type	p:last-of-type	Selects every <p> element that is the last <p> element of its parent
:link	a:link	Selects all unvisited links
:not(<i>selector</i>)	:not(p)	Selects every element that is not a <p> element
:nth-child(<i>n</i>)	p:nth-child(2)	Selects every <p> element that is the second child of its parent
:nth-last-child(<i>n</i>)	p:nth-last-child(2)	Selects every <p> element that is the second child of its parent, counting from the last child
:nth-last-of-type(<i>n</i>)	p:nth-last-of-type(2)	Selects every <p> element that is the second <p> element of its parent, counting from the last child
:nth-of-type(<i>n</i>)	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent
:only-of-type	p:only-of-type	Selects every <p> element that is the only <p> element of its parent
:only-child	p:only-child	Selects every <p> element that is the only child of its parent
:optional	input:optional	Selects input elements with no "required" attribute
:out-of-range	input:out-of-range	Selects input elements with a value outside a specified range
:read-only	input:read-only	Selects input elements with the "readonly" attribute specified
:read-write	input:read-write	Selects input elements with the "readonly" attribute NOT specified
:required	input:required	Selects input elements with the "required" attribute specified
:root	:root	Selects the document's root element
::selection	::selection	Selects the portion of an element that is selected by a user
:target	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)
:valid	input:valid	Selects all input elements with a valid value
:visited	a:visited	Selects all visited links