

Lesson : More on CSS selector

Descendant Combinator (space)

- Selects elements that are descendants of another element, regardless of depth.
- Syntax: ancestor descendant { styles }.

Child Combinator (>)

- Selects elements that are direct children of another element.
- Syntax: parent > child { styles }.

Adjacent Sibling Combinator (+)

- Selects an element immediately following another element.
- Syntax: element1 + element2 { styles }.

General Sibling Combinator (~)

- Selects all elements that are siblings of a specified element.
- Syntax: element1 ~ element2 { styles }.

Attribute Selectors

Presence Attribute Selector ([attribute])

- Selects elements that have a specific attribute.
- Syntax: [attribute] { styles }.

Exact Value Attribute Selector ([attribute=value])

- Selects elements that have a specific attribute with a specific value.
- Syntax: [attribute=value] { styles }.

Value Prefix Attribute Selector ([attribute^=value])

- Selects elements with an attribute whose value begins with a specific string.
- Syntax: [attribute^=value] { styles }.

Value Suffix Attribute Selector ([attribute\$=value])

- Selects elements with an attribute whose value ends with a specific string.
- Syntax: [attribute\$=value] { styles }.

Value Space-separated Attribute Selector ([attribute~=value])

- Selects elements with an attribute value containing a specified word within a space-separated list.
- Syntax: [attribute~=value] { styles }.

Value Hyphen-separated Attribute Selector ([attribute|=value])

- Selects elements with an attribute value that is exactly or starts with the specified value followed by a hyphen.
- Syntax: [attribute|=value] { styles }.

Substring Attribute Selector ([attribute*=value])*

- Selects elements that have a specific attribute whose value contains a specific substring.
- Syntax: [attribute*=value] { styles }.

Pseudo-elements

::after

- Inserts content after the content of an element.
- Syntax: selector::after { styles }.

::before

- Inserts content before the content of an element.
- Syntax: selector::before { styles }.

::first-letter

- Styles the first letter of an element.

- Syntax: `selector::first-letter { styles }`.

`::first-line`

- Styles the first line of text in an element.
- Syntax: `selector::first-line { styles }`.

`::selection`

- Styles the portion of an element that is selected by the user.
- Syntax: `selector::selection { styles }`.

`::placeholder`

- Styles the placeholder text in input or textarea elements.
- Syntax: `selector::placeholder { styles }`.

`::marker`

- Styles the markers of list items.
- Syntax: `selector::marker { styles }`.

Pseudo-classes

- Selects an element when it is activated by the user.
- Syntax: `selector:active { styles }`.
- Selects elements when the user hovers over them with the cursor.
- Syntax: `selector:hover { styles }`.
- Selects elements that have focus.
- Syntax: `selector:focus { styles }`.
- Selects links that have been visited by the user.
- Syntax: `selector:visited { styles }`.
- Selects links that have not been visited.
- Syntax: `selector:link { styles }`.
- Selects the first child element of its parent.
- Syntax: `parent :first-child { styles }`.
- Selects the last child element of its parent.
- Syntax: `parent :last-child { styles }`.

`(n)`

- Selects the nth child of its parent.
- Syntax: `parent :nth-child(n) { styles }`.

`(n)`

- Selects the nth child of its parent, counting from the end.
- Syntax: `parent :nth-last-child(n) { styles }`.

`(n)`

- Selects the nth child of its parent with the same element type.
- Syntax: `parent :nth-of-type(n) { styles }`.

`(n)`

- Selects the nth child of its parent with the same element type, counting from the end.
- Syntax: `parent :nth-last-of-type(n) { styles }`.

`(selector)`

- Selects elements that do not match a specific selector.
- Syntax: `:not(selector) { styles }`.
- Selects the first child of its parent with the same element type.
- Syntax: `parent :first-of-type { styles }`.
- Selects the last child of its parent with the same element type.

- Syntax: `parent :last-of-type { styles }.`
- Selects elements that have no children or text nodes.
- Syntax: `selector:empty { styles }.`
- Selects input elements that are checked (e.g., checkboxes, radio buttons).
- Syntax: `input:checked { styles`