

CSS Selectors

CSS selectors target and select the HTML elements you want to style.

Specifically, CSS selectors allow you to select multiple elements at once.

They are helpful when you want to apply the same styles to more than one HTML element, because you will not repeat yourself by writing the same lines of code for different elements.

We can divide CSS selectors into five categories:

- Simple Selectors
 - CSS Element Selector
 - CSS Id Selector
 - CSS Class Selector
 - CSS Universal Selector
 - CSS Grouping Selector
- Combinator Selectors
 - Descendent Selector
 - Child Selector
 - Adjacent Sibling Selector
 - General Sibling Selector
- Pseudo-class Selectors
- Pseudo-elements Selectors
- Attribute Selectors

Simple Selectors

CSS Element Selector

The CSS type selector selects all HTML elements of the specified type.

```
p {  
  property: value;  
}
```

CSS Id Selector

The ID selector selects an HTML element based on the value of its ID attribute.

```
#my_id {  
  property: value;  
}
```

CSS Class Selector

The class selector matches and selects HTML elements based on the value of their given class. Specifically, it selects every single element in the document with that specific class name.

```
.my_class {  
  property: value;  
}
```

CSS Universal Selector

The universal selector (*) selects all HTML elements on the page.

```
* {  
  property: value;  
}
```

CSS Grouping Selector

The grouping selector selects all the HTML elements with the same style definitions.

```
h1, h2, p {  
  property: value;  
}
```

Combinator Selectors

Descendent Selector

The descendant selector matches all elements that are descendants of a specified element.

The following example selects all <p> elements inside <div> elements:

```
div p {  
  property: value;  
}
```

Child Selector

The child selector selects all elements that are the children of a specified element.

The following example selects all <p> elements that are children of a <div> element:

```
div > p {  
  property: value;  
}
```

Adjacent Sibling Selector

The adjacent sibling selector is used to select an element that is directly after another specific element.

The following example selects the first <p> element that are placed immediately after <div> elements:

```
div + p {  
  property: value;  
}
```

General Sibling Selector

The general sibling selector selects all elements that are next siblings of a specified element.

The following example selects all <p> elements that are next siblings of <div> elements:

```
div ~ p {  
  property: value;  
}
```

Pseudo-class Selectors

A pseudo-class is used to define a special state of an element.

```
selector:pseudo-class {  
  property: value;  
}
```

Examples:

```
/* unvisited link */  
a:link {  
  color: #FF0000;  
}  
  
/* visited link */  
a:visited {  
  color: #00FF00;  
}  
  
/* mouse over link */  
a:hover {  
  color: #FF00FF;  
}  
  
/* selected link */  
a:active {  
  color: #0000FF;  
}  
  
a:first-child {}  
a:last-child {}  
a:nth-child(even) {}  
a:nth-child(n) {}
```

Pseudo-elements Selectors

A CSS pseudo-element is used to style specified parts of an element.

```
selector::pseudo-element {  
  property: value;  
}
```

Examples:

```
p::before {}  
p::after {}  
p::first-letter {}  
p::first-line {}  
h1::selection {}
```

Attribute Selectors

The **[attribute]** selector is used to select elements with a specified attribute.

The following example selects all <a> elements with a target attribute:

```
a[target] {}  
a[target="_blank"] {}
```