

CSS3 selectors

- A CSS selector is a pattern to match the elements on a web page.
- The style rules associated with that selector will be applied to the elements that match the selector pattern.
- CSS selectors are used to target the HTML elements on our web pages that we want to style.
- There are a wide variety of CSS selectors available, allowing for fine-grained precision when selecting elements to style.
- A CSS selector is the first part of a CSS Rule. It is a pattern of elements and other terms that tell the browser which HTML elements should be selected to have the CSS property values inside the rule applied to them.
- The element or elements which are selected by the selector are referred to as the "subject of the selector".
- If we have more than one thing which uses the same CSS then the individual selectors can be combined into a "selector list" so that the rule is applied to all of the individual selectors.

Types of CSS selectors

Generally there are four types of selectors in css :-

a) Simple Selectors (Or, Basic Selectors)

- a.1) Element Type Selector
- a.2) Class Selector
- a.3) ID Selector
- a.4) Universal Selector(*)

b) Combinators Selectors

- b.1) Descendant Selector(space)
- b.2) Child Selector(>)
- b.3) Adjacent Sibling Selectors(+)
- b.4) General Sibling Selectors(~)

c) Attribute Selector

d) Pseudo-Class & Pseudo-Element

a) Simple Selectors

- A simple selector is either a type selector or universal selector followed immediately by zero or more attribute selectors, ID selectors in any order.
- The simple selector matches if all of its components match.

There are generally four types of simple selectors :-

1) Element Type Selector

An element type selector matches all instance of the element in the document with the corresponding element type name.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    p
    {
      font-size: 40px;
      color: blue;
    }
  </style>
</head>
<body>
  <p>Hello Html Content</p>
</body>
</html>
```

2) ID selector

The id selector selects the id attribute of an HTML element to select a specific element. An id is always unique within the page so it is chosen to select a single, unique element.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    #hello
    {
      font-size: 40px;
      color: blue;
    }
  </style>
</head>
<body>
  <p id="hello">Hello Para 1</p>
  <p>Hello Para 2</p>
</body>
</html>
```

3) Class selector

The class selectors can be used to select any HTML element that has a class attribute. All the elements having that class will be formatted according to the defined rule.

It is used with a period character . (full stop symbol) followed by the class name.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    .hello
    {
      font-size: 40px;
      color: blue;
    }
  </style>
</head>
<body>
  <p class="hello">Hello Para 1</p>
  <p>Hello Para 2</p>
</body>
</html>
```

4) Universal selector

The universal selector, denoted by an asterisk (*), matches every single element on the page.

This selector is often used to remove the default margins and paddings from the elements for quick testing purpose.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    *
    {
      margin: 0px;
      padding: 0px;
    }
  </style>
</head>
<body>
  <p class="hello">Hello Para 1</p>
  <p>Hello Para 2</p>
</body>
</html>
```

b) Combinators Selectors

- CSS combinators are explaining the relationship between two selectors.
- CSS selector can be a simple selector or a complex selector consisting of more than one selector connected using combinators.
- Combinators combine other selectors in a way that gives them a useful relationship to each other and the location of content in the document.

There are four types of combinators available in CSS :-

1) Descendant Selectors

We can use these selectors when we need to select an element that is the descendant of another element, for example, if we want to target only those anchors that are contained within an unordered list, rather than targeting all anchor elements.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    h1 p,span
    {
      color: blue;
      font-size: 40px;
    }
  </style>
</head>
<body>

  <h1>Hello Heading
    <p>Hello Paragraph Tag</p>
    <span>Hello Span Tag</span>
  </h1>

</body>
</html>
```

2) Child Selectors

A child selector is used to select only those elements that are the direct children of some element.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    ul > li
    {
      color:rgb(0, 255, 136);
      font-size: 40px;
    }
  </style>
</head>
<body>
  <ul>
    <li>Hello li 1</li>
    <li>Hello li 2</li>
    <li>Hello li 3</li>
  </ul>
</body>
</html>
```

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    ul > li,span
    {
      color:rgb(0, 255, 136);
      font-size: 40px;
    }
  </style>
</head>
<body>
  <ul>
    <li>Hello li 1</li>
    <li>Hello li 2</li>
    <li>Hello li 3</li>
    <span>Hello Span</span>
  </ul>
</body>
</html>
```



3) Adjacent Sibling Selectors

The adjacent sibling selectors can be used to select sibling elements (i.e. elements at the same level). This selector has the syntax like: E1 + E2, where E2 is the target of the selector.

This selector affect the last sibling not both.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    h1 + p
    {
      color:blue;
      font-size: 40px;
    }

    p + span
    {
      color: red;
      font-size: 20px;
    }
  </style>
</head>
<body>

  <h1>Hello Heading Tag</h1>
  <p>Hello Paragraph Tag</p>
  <span>Hello Span Tag</span>

</body>
</html>
```

4) General Sibling Selectors

The general sibling selector is similar to the adjacent sibling selector (E1 + E2), but it is less strict.

A general sibling selector is made up of two simple selectors separated by the tilde (~) character. It can be written like: E1 ~ E2, where E2 is the target of the selector.

Example

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Internal Stylesheets</title>
  <style>
    h1 ~ p
    {
      color:blue;
      font-size: 40px;
    }

    p ~ span
    {
      color: red;
      font-size: 20px;
    }
  </style>
</head>
<body>

  <h1>Hello Heading</h1>
  <p>Hello Paragraph Tag</p>
  <span>Hello Span Tag</span>

</body>
</html>
```

c) Attribute Selectors

The attribute selector is used to select the attribute according to the conditions. This selector selects all elements with a target attribute.

Below are the attribute selectors :-

Example -- [attribute]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align]
    {
      color: blue;
    }
  </style>
</head>
<body>

  <h1 align="center">Hello Heading 1</h1>
  <h1>Hello Heading 2</h1>

</body>
</html>
```

This selector selects all elements with attribute="attribute_value".

Example -- [attribute="attribute_value"]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align="center"]
    {
      color: red;
    }
  </style>
</head>
<body>
  <h1 align="center">Hello Heading 1</h1>
  <h1 align="left">Hello Heading 2</h1>

</body>
</html>
```

LEARN 2 EARN
LABS

Selects all the elements containing the word related to "attribute_value".

Example -- [attribute~="attribute_value"]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align~="center"]
    {
      color: red;
    }
  </style>
</head>
<body>
  <h1 align="center">Hello Heading 1</h1>
  <h1 align="left">Hello Heading 2</h1>

</body>
</html>
```

LEARN 2 EARN
LABS

Selects all elements whose starting value matches with "attribute_value".

Example -- [attribute,"attribute_value"]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align,"center"]
    {
      color: red;
    }
  </style>
</head>
<body>

  <h1 align="center">Hello Heading 1</h1>
  <h1 align="left">Hello Heading 2</h1>

</body>
</html>
```

LEARN 2 EARN
LABS

Selects every element whose "attribute_value" begins with "at".

Example -- [attribute^="attribute_value"]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align^="ce"]
    {
      color: red;
    }
  </style>
</head>
<body>
  <h1 align="center">Hello Heading 1</h1>
  <h1 align="left">Hello Heading 2</h1>

</body>
</html>
```

LEARN 2 EARN
LABS

Selects every element whose "attribute_value" ends with "ue".

Example -- [attribute\$="attribute_value"]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align$="er"]
    {
      color: red;
    }
  </style>
</head>
<body>
  <h1 align="center">Hello Heading 1</h1>
  <h1 align="left">Hello Heading 2</h1>

</body>
</html>
```

LEARN 2 EARN
LABS

Selects every element whose "attribute_value" contains the substring "attribute_value".

Example -- [attribute*="attribute_value"]

```
<!DOCTYPE html>
<html lang="en">
<head>
  <title>Document</title>
  <style>
    [align*="e"]
    {
      color: red;
    }
  </style>
</head>
<body>
  <h1 align="center">Hello Heading 1</h1>
  <h1 align="left">Hello Heading 2</h1>

</body>
</html>
```

LEARN 2 EARN
LABS