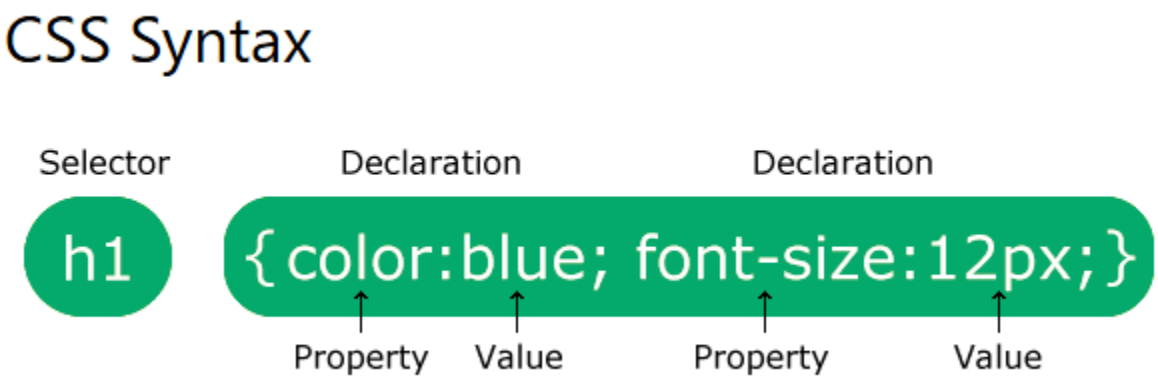
What is CSS?

CSS is the language we use to style a Web page.

- CSS stands for Cascading Style Sheets
- CSS describes how HTML elements are to be displayed on screen, paper, or in other media
- CSS saves a lot of work. It can control the layout of multiple web pages all at once
- External stylesheets are stored in CSS files



CSS Selectors

CSS selectors are used to "find" (or select) the HTML elements you want to style.

We can divide CSS selectors into five categories:

- Simple selectors (select elements based on name, id, class)

The CSS element Selector

The element selector selects HTML elements based on the element name.

Example

Here, all <p> elements on the page will be center-aligned, with a red text color:

```
<!DOCTYPE html>

<html>

<head>

<style>

p {

  text-align: center;

  color: red;

}

</style>

</head>

<body>


<p>Every paragraph will be affected by the style.</p>
```

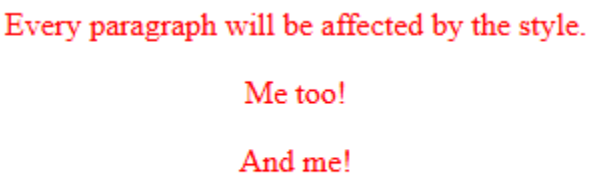
```
<p>Me too!</p>

<p>And me!</p>


</body>

</html>
```

Output:-



The CSS id Selector

The id selector uses the id attribute of an HTML element to select a specific element.

The id of an element is unique within a page, so the id selector is used to select one unique element!

To select an element with a specific id, write a hash (#) character, followed by the id of the element.

Example

The CSS rule below will be applied to the HTML element with id="para1":

```
<!DOCTYPE html>

<html>

<head>

<style>
```

```
#para1 {
  text-align: center;
  color: red;
}
</style>
</head>
<body>

<p id="para1">Hello World!</p>
<p>This paragraph is not affected by the style.</p>

</body>
</html>
```

Output:-

The CSS class Selector

The class selector selects HTML elements with a specific class attribute.

To select elements with a specific class, write a period (.) character, followed by the class name.

Example

In this example all HTML elements with class="center" will be red and center-aligned:

```
<html>

<head>

<style>

.center {

  text-align: center;

  color: red;

}

</style>

</head>

<body>



<h1 class="center">Red and center-aligned heading</h1>

<p class="center">Red and center-aligned paragraph.</p>



</body>

</html>
```

Output:-

CSS Combinators:-

A combinator is something that explains the relationship between the selectors.

A CSS selector can contain more than one simple selector. Between the simple selectors, we can include a combinator.

There are four different combinators in CSS:

- Descendant combinator (space)
- Child combinator (>)
- Next sibling combinator (+)
- Subsequent-sibling combinator (~)

Descendant Combinator:-

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

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

```
<!DOCTYPE html>

<html>

<head>

<style>

div p {

  background-color: yellow;

}

</style>

</head>

<body>


<h2>Descendant Selector</h2>


<p>The descendant selector matches all elements that are descendants of a specified element.</p>
```

```
<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <section><p>Paragraph 3 in the div.</p></section>
</div>

<p>Paragraph 4. Not in a div.</p>
<p>Paragraph 5. Not in a div.</p>

</body>
</html>
```

Child Combinator (>):-

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

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

```
<!DOCTYPE html>
<html>
<head>
<style>
div > p {
  background-color: yellow;
```

```
}
</style>
</head>
<body>

<h2>Child Selector</h2>

<p>The child selector (>) selects all elements that are the children of a specified element.</p>

<div>
  <p>Paragraph 1 in the div.</p>
  <p>Paragraph 2 in the div.</p>
  <section>
    <!-- not Child but Descendant -->
    <p>Paragraph 3 in the div (inside a section element).</p>
  </section>
  <p>Paragraph 4 in the div.</p>
</div>

<p>Paragraph 5. Not in a div.</p>
<p>Paragraph 6. Not in a div.</p>

</body>
</html>
```

Next Sibling Combinator (+)

The next sibling combinator is used to select an element that is directly after another specific element.

Sibling elements must have the same parent element, and "adjacent" means "immediately following".

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

```
<!DOCTYPE html>

<html>

<head>

<style>

div + p {

  background-color: yellow;

}

</style>

</head>

<body>



<h2>Adjacent Sibling Selector</h2>



<p>The + selector is used to select an element that is directly after another specific element.</p>

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



<div>

  <p>Paragraph 1 in the div.</p>
```

```
  <p>Paragraph 2 in the div.</p>
</div>

<p>Paragraph 3. After a div.</p>
<p>Paragraph 4. After a div.</p>

<div>
  <p>Paragraph 5 in the div.</p>
  <p>Paragraph 6 in the div.</p>
</div>

<p>Paragraph 7. After a div.</p>
<p>Paragraph 8. After a div.</p>

</body>
</html>
```

Subsequent-sibling Combinator (~)

The subsequent-sibling combinator selects all elements that are next siblings of a specified element.

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

```
<!DOCTYPE html>

<html>

<head>

<style>

div ~ p {

  background-color: yellow;

}

</style>

</head>

<body>


<h2>General Sibling Selector</h2>


<p>The general sibling selector (~) selects all elements that are next siblings of a specified element.</p>


<p>Paragraph 1.</p>


<div>

  <p>Paragraph 2.</p>

</div>
```

Paragraph 3.

<code>Some code.</code>

Paragraph 4.

</body>

</html>

CSS Pseudo-classes:-

What are Pseudo-classes?

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

For example, it can be used to:

- Style an element when a user moves the mouse over it
- Style visited and unvisited links differently
- Style an element when it gets focus
- Style valid/invalid/required/optional form elements

Syntax

The syntax of pseudo-classes:

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

Anchor Pseudo-classes

Links can be displayed in different ways:

Example

```
/* 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;
}
```

Example :-

<!DOCTYPE html>

<html>

<head>

<style>

/* unvisited link */

a:link {

  color: red;

```
}

/* visited link */
a:visited {
  color: green;
}

/* mouse over link */
a:hover {
  color: hotpink;
}

/* selected link */
a:active {
  color: blue;
}
</style>
</head>
<body>

<h2>Styling a link depending on state</h2>

<p><b><a href="#" target="_blank">This is a link</a></b></p>

<p><b>Note:</b> a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective.</p>

<p><b>Note:</b> a:active MUST come after a:hover in the CSS definition in order to be effective.</p>
```

```
</body>

</html>
```

Pseudo-classes and HTML Classes

Pseudo-classes can be combined with HTML classes:

When you hover over the link in the example, it will change color:

Example

```
a.highlight:hover {
  color: #ff0000;
}
```

```
<!DOCTYPE html>

<html>

<head>

<style>

a.highlight:hover {

 color: #ff0000;

 font-size: 22px;

}

</style>

</head>

<body>
```

<h2>Pseudo-classes and HTML Classes</h2>

When you hover over the first link below, it will change color and font size:

CSS Syntax

CSS Tutorial

</body>

</html>

Hover on

An example of using the `:hover` pseudo-class on a element:

Example

```
div:hover {
  background-color: blue;
}
```

<!DOCTYPE html>

<html>

<head>

<style>

div {

  background-color: green;

```
  color: white;
  padding: 25px;
  text-align: center;
}

div:hover {
  background-color: blue;
}
</style>
</head>
<body>

<p>Mouse over the div element below to change its background color:</p>

<div>Mouse Over Me</div>

</body>
</html>
```

Simple Tooltip Hover

Hover over a element to show a <p> element (like a tooltip):

Example

```
p {
  display: none;
  background-color: yellow;
```

```
  padding: 20px;
}

div:hover p {
  display: block;
}
```

```
<!DOCTYPE html>

<html>

<head>

<style>

p {

 display: none;

 background-color: yellow;

 padding: 20px;

}


div:hover p {

 display: block;

}

</style>

</head>

<body>


<div>Hover over this div element to show the p element

 <p>Tada! Here I am!</p>
```

```
</div>


</body>

</html>
```