

CSS Tutorial

CSS tutorial or CSS 3 tutorial provides basic and advanced concepts of CSS technology. Our CSS tutorial is developed for beginners and professionals. The major points of CSS are given below:

- CSS stands for Cascading Style Sheet.
- CSS is used to design HTML tags.
- CSS is a widely used language on the web.
- HTML, CSS and JavaScript are used for web designing. It helps the web designers to apply style on HTML tags.

CSS Example with CSS Editor

In this tutorial, you will get a lot of CSS examples, you can edit and run these examples with our online CSS editor tool.

```
1. <!DOCTYPE>
2. <html>
3. <head>
4. <style>
5. h1{
6. color:white;
7. background-color:red;
8. padding:5px;
9. }
10. p{
11. color:blue;
12. }
13. </style>
14. </head>
15. <body>
16. <h1>Write Your First CSS Example</h1>
17. <p>This is Paragraph.</p>
18. </body>
19. </html>
```

Output:

Write Your First CSS Example

This is Paragraph.

CSS 3 Tutorial

In this tutorial, we will learn CSS 3 properties to design box model, apply opacity, radius etc.

All CSS Properties

In this tutorial, you will get details of all CSS properties such as background, border, font, float, display, margin, opacity, padding, text-align, vertical-align, position, color etc.

Prerequisite

Before learning CSS, you must have the basic knowledge of HTML.

Audience

Our CSS tutorial is designed to help beginners and professionals both.

Problem

If you find any problem or mistake in our CSS tutorial, you can report to us. We assure, you will not find any problem in CSS tutorial.

What is CSS

CSS stands for Cascading Style Sheets. It is a style sheet language which is used to describe the look and formatting of a document written in markup language. It provides an additional feature to HTML. It is generally used with HTML to change the style of web pages and user interfaces. It can also be used with any kind of XML documents including plain XML, SVG and XUL.

CSS is used along with HTML and JavaScript in most websites to create user interfaces for web applications and user interfaces for many mobile applications.

What does CSS do

- You can add new looks to your old HTML documents.
- You can completely change the look of your website with only a few changes in CSS code.

Why use CSS

These are the three major benefits of CSS:

1) Solves a big problem

Before CSS, tags like font, color, background style, element alignments, border and size had to be repeated on every web page. This was a very long process. For example: If you are developing a large website where fonts and color information are added on every single page, it will become a long and expensive process. CSS was created to solve this problem. It was a W3C recommendation.

2) Saves a lot of time

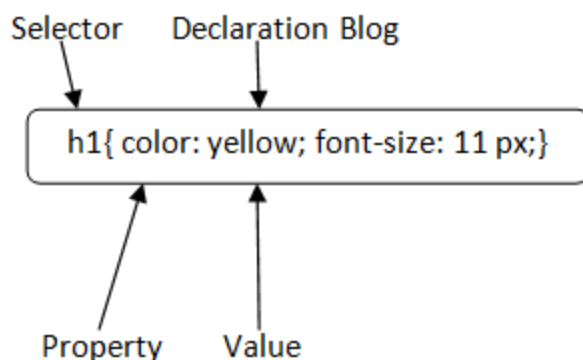
CSS style definitions are saved in external CSS files so it is possible to change the entire website by changing just one file.

3) Provide more attributes

CSS provides more detailed attributes than plain HTML to define the look and feel of the website.

CSS Syntax

A CSS rule set contains a selector and a declaration block.



Selector: Selector indicates the HTML element you want to style. It could be any tag like `<h1>`, `<title>` etc.

Declaration Block: The declaration block can contain one or more declarations separated by a semicolon. For the above example, there are two declarations:

1. `color: yellow;`
2. `font-size: 11 px;`

Each declaration contains a property name and value, separated by a colon.

Property: A Property is a type of attribute of HTML element. It could be color, border etc.

Value: Values are assigned to CSS properties. In the above example, value "yellow" is assigned to color property.

1. `Selector{Property1: value1; Property2: value2;;}`

CSS Selector

CSS selectors are used *to select the content you want to style*. Selectors are the part of CSS rule set. CSS selectors select HTML elements according to its id, class, type, attribute etc.

There are several different types of selectors in CSS.

1. CSS Element Selector
2. CSS Id Selector
3. CSS Class Selector
4. CSS Universal Selector
5. CSS Group Selector

1) CSS Element Selector

The element selector selects the HTML element by name.

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<style>`
5. `p{`
6. `text-align: center;`
7. `color: blue;`
8. `}`
9. `</style>`
10. `</head>`
11. `<body>`
12. `<p>This style will be applied on every paragraph.</p>`
13. `<p id="para1">Me too!</p>`
14. `<p>And me!</p>`
15. `</body>`
16. `</html>`

Output:

This style will be applied on every paragraph.

Me too!

And me!

2) CSS 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.

It is written with the hash character (#), followed by the id of the element.

Let's take an example with the id "para1".

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <style>
5. #para1 {
6.     text-align: center;
7.     color: blue;
8. }
9. </style>
10. </head>
11. <body>
12. <p id="para1">Hello Javatpoint.com</p>
13. <p>This paragraph will not be affected.</p>
14. </body>
15. </html>
```

Output:

Hello Javatpoint.com

This paragraph will not be affected.

3) CSS Class Selector

The class selector selects HTML elements with a specific class attribute. It is used with a period character . (full stop symbol) followed by the class name.

Note: A class name should not be started with a number.

Let's take an example with a class "center".

```
1. <!DOCTYPE html>
2. <html>
3. <head>
4. <style>
5. .center {
6.     text-align: center;
```

```

7.     color: blue;
8. }
9. </style>
10. </head>
11. <body>
12. <h1 class="center">This heading is blue and center-aligned.</h1>
13. <p class="center">This paragraph is blue and center-aligned.</p>
14. </body>
15. </html>

```

Output:

This heading is blue and center-aligned.

This paragraph is blue and center-aligned.

CSS Class Selector for specific element

If you want to specify that only one specific HTML element should be affected then you should use the element name with class selector.

Let's see an example.

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <style>
5. p.center {
6.     text-align: center;
7.     color: blue;
8. }
9. </style>
10. </head>
11. <body>
12. <h1 class="center">This heading is not affected</h1>
13. <p class="center">This paragraph is blue and center-aligned.</p>
14. </body>
15. </html>

```

Output:

This heading is not affected

This paragraph is blue and center-aligned.

4) CSS Universal Selector

The universal selector is used as a wildcard character. It selects all the elements on the pages.

1. `<!DOCTYPE html>`
2. `<html>`
3. `<head>`
4. `<style>`
5. `* {`
6. `color: green;`
7. `font-size: 20px;`
8. `}`
9. `</style>`
10. `</head>`
11. `<body>`
12. `<h2>This is heading</h2>`
13. `<p>This style will be applied on every paragraph.</p>`
14. `<p id="para1">Me too!</p>`
15. `<p>And me!</p>`
16. `</body>`
17. `</html>`

Output:

This is heading

This style will be applied on every paragraph.

Me too!

And me!

5) CSS Group Selector

The grouping selector is used to select all the elements with the same style definitions.

Grouping selector is used to minimize the code. Commas are used to separate each selector in grouping.

Let's see the CSS code without group selector.

1. `h1 {`
2. `text-align: center;`

```

3.    color: blue;
4.  }
5.  h2 {
6.    text-align: center;
7.    color: blue;
8.  }
9.  p {
10.   text-align: center;
11.   color: blue;
12. }

```

As you can see, you need to define CSS properties for all the elements. It can be grouped in following ways:

```

1.  h1,h2,p {
2.    text-align: center;
3.    color: blue;
4.  }

```

Let's see the full example of CSS group selector.

```

1.  <!DOCTYPE html>
2.  <html>
3.  <head>
4.  <style>
5.  h1, h2, p {
6.    text-align: center;
7.    color: blue;
8.  }
9.  </style>
10. </head>
11. <body>
12. <h1>Hello Javatpoint.com</h1>
13. <h2>Hello Javatpoint.com (In smaller font)</h2>
14. <p>This is a paragraph.</p>
15. </body>
16. </html>

```

Output:

Hello Javatpoint.com

Hello Javatpoint.com (In smaller font)

This is a paragraph.

How to add CSS

CSS is added to HTML pages to format the document according to information in the style sheet. There are three ways to insert CSS in HTML documents.

1. Inline CSS
2. Internal CSS
3. External CSS

1) Inline CSS

Inline CSS is used to apply CSS on a single line or element.

For example:

1. `<p style="color:blue">Hello CSS</p>`

For more visit here: [Inline CSS](#)

2) Internal CSS

Internal CSS is used to apply CSS on a single document or page. It can affect all the elements of the page. It is written inside the style tag within head section of html.

For example:

1. `<style>`
2. `p{color:blue}`
3. `</style>`

For more visit here: [Internal CSS](#)

3) External CSS

External CSS is used to apply CSS on multiple pages or all pages. Here, we write all the CSS code in a css file. Its extension must be .css for example style.css.

For example:

1. `p{color:blue}`

You need to link this style.css file to your html pages like this:

1. `<link rel="stylesheet" type="text/css" href="style.css">`

The link tag must be used inside head section of html.

Inline CSS

We can apply CSS in a single element by inline CSS technique.

The inline CSS is also a method to insert style sheets in HTML document. This method mitigates some advantages of style sheets so it is advised to use this method sparingly.

If you want to use inline CSS, you should use the style attribute to the relevant tag.

Syntax:

1. `<htmltag style="cssproperty1:value; cssproperty2:value;"> </htmltag>`

Example:

1. `<h2 style="color:red;margin-left:40px;">`Inline CSS is applied on this heading.`</h2>`
2. `<p>`This paragraph is not affected.`</p>`

Test it Now

Output:

Inline CSS is applied on this heading.

This paragraph is not affected.

Disadvantages of Inline CSS

- You cannot use quotations within inline CSS. If you use quotations the browser will interpret this as an end of your style value.
- These styles cannot be reused anywhere else.
- These styles are tough to be edited because they are not stored at a single place.
- It is not possible to style pseudo-codes and pseudo-classes with inline CSS.
- Inline CSS does not provide browser cache advantages.

Internal CSS

The internal style sheet is used to add a unique style for a single document. It is defined in `<head>` section of the HTML page inside the `<style>` tag.

Example:

```

1. <!DOCTYPE html>
2. <html>
3. <head>
4. <style>
5. body {
6.     background-color: linen;
7. }
8. h1 {
9.     color: red;
10.    margin-left: 80px;
11. }
12. </style>
13. </head>
14. <body>
15. <h1>The internal style sheet is applied on this heading.</h1>
16. <p>This paragraph will not be affected.</p>
17. </body>
18. </html>

```

Output:

The internal style sheet is applied on this heading.

This paragraph will not be affected.

External CSS

The external style sheet is generally used when you want to make changes on multiple pages. It is ideal for this condition because it facilitates you to change the look of the entire web site by changing just one file.

It uses the <link> tag on every pages and the <link> tag should be put inside the head section.

Example:

```

1. <head>
2. <link rel="stylesheet" type="text/css" href="mystyle.css">
3. </head>

```

The external style sheet may be written in any text editor but must be saved with a .css extension. This file should not contain HTML elements.

Let's take an example of a style sheet file named "mystyle.css".

File: mystyle.css

1. body {
2. background-color: lightblue;
3. }
4. h1 {
5. color: navy;
6. margin-left: 20px;
7. }

Note: You should not use a space between the property value and the unit. For example: It should be margin-left:20px not margin-left:20 px.