**Cascading Style Sheets (CSS)**

1. **CSS Introduction**

CSS is a language that describes the style of an HTML document.CSS describes how HTML elements should be displayed. World Wide Web Consortium (W3C) has created CSS.This tutorial will teach you CSS from basic to advanced.

**What is CSS?**

* 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

## Why Use CSS?

CSS is used to define styles for your web pages, including the design, layout and variations in display for different devices and screen sizes.

***CSS Sample example***

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-color: lightblue;

}

h1 {

color: white;

text-align: center;

}

p {

font-family: verdana;

font-size: 20px;

}

</style>

</head>

<body>

<h1>My First CSS Example</h1>

<p>This is a paragraph.</p>

</body>

</html>

1. **CSS Syntax**

A CSS rule-set consists of a selector and a declaration block:



The selector points to the HTML element you want to style.

The declaration block contains one or more declarations separated by semicolons.

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

A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces.

In the following example all <p> elements will be center-aligned, with a red text color:

***Example :***

<!DOCTYPE html>

<html>

<head>

<style>

p {

color: red;

text-align: center;

} </style>

</head>

<body>

<p>Hello World!</p>

<p>These paragraphs are styled with CSS.</p>

</body>

</html>

## CSS Selectors

CSS selectors are used to "find" (or select) HTML elements based on their element name, id, class, attribute, and more.

## The element Selector

The element selector selects elements based on the element name.

You can select all <p> elements on a page like this (in this case, all <p> elements will be center-aligned, with a red text color):

**Example:**

<!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 id="para1">Me too!</p>

<p>And me!</p>

</body>

</html>

## The id Selector

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

The id of an element should be 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.

The style 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>

## The class Selector

The class selector selects elements with a specific class attribute.

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

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

<!DOCTYPE html>

<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>

You can also specify that only specific HTML elements should be affected by a class.

In the example below, only <p> elements with class="center" will be center-aligned:

<!DOCTYPE html>

<html>

<head>

<style>

p.center {

text-align: center;

color: red;

}

</style>

</head>

<body>

<h1 class="center">This heading will not be affected</h1>

<p class="center">This paragraph will be red and center-aligned.</p>

</body>

</html>

HTML elements can also refer to more than one class.

In the example below, the <p> element will be styled according to class="center" and to class="large":

<!DOCTYPE html>

<html>

<head>

<style>

p.center {

text-align: center;

color: red;

}

p.large {

font-size: 300%;

}

</style>

</head>

<body>

<h1 class="center">This heading will not be affected</h1>

<p class="center">This paragraph will be red and center-aligned.</p>

<p class="center large">This paragraph will be red, center-aligned, and in a large font-size.</p>

</body>

</html>

## Grouping Selectors

If you have elements with the same style definitions, like this:

<!DOCTYPE html>

<html>

<head>

<style>

h1, h2, p {

text-align: center;

color: red;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<h2>Smaller heading!</h2>

<p>This is a paragraph.</p>

</body>

</html>

## CSS Comments

Comments are used to explain the code, and may help when you edit the source code at a later date.

Comments are ignored by browsers.

A CSS comment starts with /\* and ends with \*/. Comments can also span multiple lines:

<!DOCTYPE html>

<html>

<head>

<style>

p {

color: red;

/\* This is a single-line comment \*/

text-align: center;

}

/\* This is

a multi-line

comment \*/

</style>

</head>

<body>

<p>Hello World!</p>

<p>This paragraph is styled with CSS.</p>

<p>CSS comments are not shown in the output.</p>

</body>

</html>

# **CSS How To...**

## Three Ways to Insert CSS

There are three ways of inserting a style sheet:

* External style sheet
* Internal style sheet
* Inline style

## External Style Sheet

With an external style sheet, you can change the look of an entire website by changing just one file!

Each page must include a reference to the external style sheet file inside the <link> element. The <link> element goes inside the <head> section:

<!DOCTYPE html>

<html>

<head>

/\* rel – Relation, \*/

<link rel="stylesheet" type="text/css" href="mystyle.css">

</head>

<body>

<h1>This is a heading</h1>

<p>This is a paragraph.</p>

</body>

</html>

## Internal Style Sheet

An internal style sheet may be used if one single page has a unique style.

Internal styles are defined within the <style> element, inside the <head> section of an HTML page:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-color: linen;

}

h1 {

color: maroon;

margin-left: 40px;

}

</style>

</head>

<body>

<h1>This is a heading</h1>

<p>This is a paragraph.</p>

</body>

</html>

## Inline Styles

An inline style may be used to apply a unique style for a single element.

To use inline styles, add the style attribute to the relevant element. The style attribute can contain any CSS property.

The example below shows how to change the color and the left margin of a <h1> element:

<!DOCTYPE html>

<html>

<body>

<h1 style="color:blue;margin-left:30px;">This is a heading</h1>

<p>This is a paragraph.</p>

<h1>This is for testing</h1>

</body>

</html>

Cascading Order

What style will be used when there is more than one style specified for an HTML element?

Generally speaking we can say that all the styles will "cascade" into a new "virtual" style sheet by the following rules, where number one has the highest priority:

1. Inline style (inside an HTML element)
2. External and internal style sheets (in the head section)
3. Browser default

So, an inline style (inside a specific HTML element) has the highest priority, which means that it will override a style defined inside the <head> tag, or in an external style sheet, or a browser default value.

<!DOCTYPE html>

<html>

<head>

<link rel="stylesheet" type="text/css" href="mystyle.css">

<style>

body {background-color: linen;}

</style>

</head>

<body style="background-color: lightcyan">

<h1>Multiple Styles Will Cascade into One</h1>

<p>In this example, the background color is set inline, in an internal stylesheet, and in an external stylesheet.</p>

<p>Try experimenting by removing styles to see how the cascading stylesheets work. (try removing the inline first, then the internal, then the external)</p>

</body>

</html>

# **CSS Colors**

Colors in CSS are most often specified by:

* a valid color name - like "red"
* an RGB value - like "rgb(255, 0, 0)"
* a HEX value - like "#ff0000"

## Color Names

Colors set by using color names:

<!DOCTYPE html>

<html>

<body>

<h2>Color Names Examples</h2>

<p>Note: You will learn more about the background-color and the color property later in our tutorial.</p>

<h2 style="background-color:red">

Red background-color

</h2>

<h2 style="background-color:green">

Green background-color

</h2>

<h2 style="background-color:blue;color:white">

Blue background-color and white text color

</h2>

<h2 style="background-color:orange">

Orange background-color

</h2>

<h2 style="background-color:yellow">

Yellow background-color

</h2>

<h2 style="background-color:cyan">

Cyan background-color

</h2>

<h2 style="background-color:black;color:white">

Black background-color and white text color

</h2>

</body>

</html>

## RGB (Red, Green, Blue)

<!DOCTYPE html>

<html>

<body>

<h2>RGB Color Examples</h2>

<h2 style="background-color:rgb(255, 0, 0)">

Background-color set by using rgb(255, 0, 0)

</h2>

<h2 style="background-color:rgb(0, 255, 0)">

Background-color set by using rgb(0, 255, 0)

</h2>

<h2 style="background-color:rgb(0, 0, 255)">

Background-color set by using rgb(0, 0, 255)

</h2>

<h2 style="background-color:rgb(255, 165, 0)">

Background-color set by using rgb(255, 165, 0)

</h2>

<h2 style="background-color:rgb(255, 255, 0)">

Background-color set by using rgb(255, 255, 0)

</h2>

<h2 style="background-color:rgb(0, 255, 255)">

Background-color set by using rgb(0, 255, 255)

</h2>

</body>

</html>

## Hexadecimal Colors

<!DOCTYPE html>

<html>

<body>

<h2>HEX Color Examples</h2>

<h2 style="background-color:#FF0000">

Background-color set by using #FF0000

</h2>

<h2 style="background-color:#00FF00">

Background-color set by using #00FF00

</h2>

<h2 style="background-color:#0000FF">

Background-color set by using #0000FF

</h2>

<h2 style="background-color:#FFA500">

Background-color set by using #FFA500

</h2>

<h2 style="background-color:#FFFF00">

Background-color set by using #FFFF00

</h2>

<h2 style="background-color:#00FFFF">

Background-color set by using #00FFFF

</h2>

</body>

</html>

# **CSS Backgrounds**

The CSS background properties are used to define the background effects for elements.

CSS background properties:

* background-color
* background-image
* background-repeat
* background-attachment
* background-position

## Background Color

The background-color property specifies the background color of an element.

The background color of a page is set like this:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-color: lightblue;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>This page has a light blue background color!</p>

</body>

</html>

## Background Image

The background-image property specifies an image to use as the background of an element.

By default, the image is repeated so it covers the entire element.

The background image for a page can be set like this:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("paper.gif");

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>This page has an image as the background!</p>

</body>

</html>

## Background Image - Repeat Horizontally or Vertically

By default, the background-image property repeats an image both horizontally and vertically.

Some images should be repeated only horizontally or vertically, or they will look strange, like this:

If the image above is repeated only horizontally (background-repeat: repeat-x;), the background will look better:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("gradient\_bg.png");

background-repeat: repeat-x;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>Here, a backgound image is repeated only horizontally!</p>

</body>

</html>

To repeat an image vertically, set background-repeat: repeat-y;

## Background Image - Set position and no-repeat

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("img\_tree.png");

background-repeat: no-repeat;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>W3Schools background image example.</p>

<p>The background image is only showing once, but it is disturbing the reader!</p>

</body>

</html>

In the example above, the background image is shown in the same place as the text. We want to change the position of the image, so that it does not disturb the text too much.

The position of the image is specified by the background-position property:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("img\_tree.png");

background-repeat: no-repeat;

background-position: right top;

margin-right: 200px;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>W3Schools background no-repeat, set position example.</p>

<p>Now the background image is only shown once, and positioned away from the text.</p>

<p>In this example we have also added a margin on the right side, so the background image will never disturb the text.</p>

</body>

</html>

## Background Image - Fixed position

To specify that the background image should be fixed (will not scroll with the rest of the page), use the background-attachment property:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("img\_tree.png");

background-repeat: no-repeat;

background-position: right top;

margin-right: 200px;

background-attachment: fixed;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>The background-image is fixed. Try to scroll down the page.</p>

<p>If you do not see any scrollbars, try to resize the browser window.</p>

</body>

</html>

## Background - Shorthand property

To shorten the code, it is also possible to specify all the background properties in one single property. This is called a shorthand property.

The shorthand property for background is background:

<!DOCTYPE html>

<html>

<head>

<style>

body {

background: #ffffff url("img\_tree.png") no-repeat right top;

margin-right: 200px;

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>Now the background image is only shown once, and it is also positioned away from the text.</p>

<p>In this example we have also added a margin on the right side, so that the background image will not disturb the text.</p>

</body>

</html>

1. **CSS**Borders

## CSS Border Properties

The CSS border properties allow you to specify the style, width, and color of an element's border.

## Border Style

The border-style property specifies what kind of border to display.

The following values are allowed:

* dotted - Defines a dotted border
* dashed - Defines a dashed border
* solid - Defines a solid border
* double - Defines a double border
* groove - Defines a 3D grooved border. The effect depends on the border-color value
* ridge - Defines a 3D ridged border. The effect depends on the border-color value
* inset - Defines a 3D inset border. The effect depends on the border-color value
* outset - Defines a 3D outset border. The effect depends on the border-color value
* none - Defines no border
* hidden - Defines a hidden border

The border-style property can have from one to four values (for the top border, right border, bottom border, and the left border).

## Example

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p.dotted {border-style: dotted;}

## p.dashed {border-style: dashed;}

## p.solid {border-style: solid;}

## p.double {border-style: double;}

## p.groove {border-style: groove;}

## p.ridge {border-style: ridge;}

## p.inset {border-style: inset;}

## p.outset {border-style: outset;}

## p.none {border-style: none;}

## p.hidden {border-style: hidden;}

## p.mix {border-style: dotted dashed solid double;}

## </style>

## </head>

## <body>

## <h2>The border-style Property</h2>

## <p>This property specifies what kind of border to display:</p>

## <p class="dotted">A dotted border.</p>

## <p class="dashed">A dashed border.</p>

## <p class="solid">A solid border.</p>

## <p class="double">A double border.</p>

## <p class="groove">A groove border.</p>

## <p class="ridge">A ridge border.</p>

## <p class="inset">An inset border.</p>

## <p class="outset">An outset border.</p>

## <p class="none">No border.</p>

## <p class="hidden">A hidden border.</p>

## <p class="mix">A mixed border.</p>

## </body>

## </html>

## Border Width

The border-width property specifies the width of the four borders.

The width can be set as a specific size (in px, pt, cm, em, etc) or by using one of the three pre-defined values: thin, medium, or thick.

The border-width property can have from one to four values (for the top border, right border, bottom border, and the left border).

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p.one {

## border-style: solid;

## border-width: 5px;

## }

## p.two {

## border-style: solid;

## border-width: medium;

## }

## p.three {

## border-style: dotted;

## border-width: 2px;

## }

## p.four {

## border-style: dotted;

## border-width: thick;

## }

## p.five {

## border-style: double;

## border-width: 15px;

## }

## p.six {

## border-style: double;

## border-width: thick;

## }

## p.seven {

## border-style: solid;

## border-width: 2px 10px 4px 20px;

## }

## </style>

## </head>

## <body>

## <h2>The border-width Property</h2>

## <p>This property specifies the width of the four borders:</p>

## <p class="one">Some text.</p>

## <p class="two">Some text.</p>

## <p class="three">Some text.</p>

## <p class="four">Some text.</p>

## <p class="five">Some text.</p>

## <p class="six">Some text.</p>

## <p class="seven">Some text.</p>

## <p><b>Note:</b> The "border-width" property does not work if it is used alone.

## Always specify the "border-style" property to set the borders first.</p>

## </body>

## </html>

Border Color

The border-color property is used to set the color of the four borders.

The color can be set by:

* name - specify a color name, like "red"
* Hex - specify a hex value, like "#ff0000"
* RGB - specify a RGB value, like "rgb(255,0,0)"
* transparent

The border-color property can have from one to four values (for the top border, right border, bottom border, and the left border).

If border-color is not set, it inherits the color of the element.

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p.one {

## border-style: solid;

## border-color: red;

## }

## p.two {

## border-style: solid;

## border-color: green;

## }

## p.three {

## border-style: solid;

## border-color: red green blue yellow;

## }

## </style>

## </head>

## <body>

## <h2>The border-color Property</h2>

## <p>This property specifies the color of the four borders:</p>

## <p class="one">A solid red border</p>

## <p class="two">A solid green border</p>

## <p class="three">A solid multicolor border</p>

## <p><b>Note:</b> The "border-color" property does not work if it is used alone. Use the "border-style" property to set the borders first.</p>

## </body>

## </html>

## Border - Individual Sides

From the examples above you have seen that it is possible to specify a different border for each side.

In CSS, there are also properties for specifying each of the borders (top, right, bottom, and left):

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p {

## border-top-style: dotted;

## border-right-style: solid;

## border-bottom-style: dotted;

## border-left-style: solid;

## }

## </style>

## </head>

## <body>

## <p>2 different border styles.</p>

## </body>

## </html>

## The example above gives the same result as this:

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p {

## border-style: dotted solid;

## }

## </style>

## </head>

## <body>

## <p>2 different border styles.</p>

## </body>

## </html>

So, here is how it works:

If the border-style property has four values:

* **border-style: dotted solid double dashed;**
  + top border is dotted
  + right border is solid
  + bottom border is double
  + left border is dashed

If the border-style property has three values:

* **border-style: dotted solid double;**
  + top border is dotted
  + right and left borders are solid
  + bottom border is double

If the border-style property has two values:

* **border-style: dotted solid;**
  + top and bottom borders are dotted
  + right and left borders are solid

If the border-style property has one value:

* **border-style: dotted;**
  + all four borders are dotted

The border-style property is used in the example above. However, it also works with border-width and border-color.

Border - Shorthand Property

As you can see from the examples above, there are many properties to consider when dealing with borders.

To shorten the code, it is also possible to specify all the individual border properties in one property.

The border property is a shorthand property for the following individual border properties:

* border-width
* border-style (required)
* border-color

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p {

## border: 5px solid red;

## }

## </style>

## </head>

## <body>

## <h2>The border Property</h2>

## <p>This property is a shorthand property for border-width, border-style, and border-color.</p>

## </body>

## </html>

You can also specify all the individual border properties for just one side:

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p {

## border-left: 6px solid red;

## background-color: lightgrey;

## }

## </style>

## </head>

## <body>

## <h2>The border-left Property</h2>

## <p>This property is a shorthand property for border-left-width, border-left-style, and border-left-color.</p>

## </body>

## </html>

## Rounded border

## <!DOCTYPE html>

## <html>

## <head>

## <style>

## p {

## border-bottom: 6px solid red;

## background-color: lightgrey;

## }

## </style>

## </head>

## <body>

## <h2>The border-bottom Property</h2>

## <p>This property is a shorthand property for border-bottom-width, border-bottom-style, and border-bottom-color.</p>

## </body>

## </html>

<!DOCTYPE html>

<html>

<head>

<style>

p.normal {

border: 2px solid red;

}

p.round1 {

border: 2px solid red;

border-radius: 5px;

}

p.round2 {

border: 2px solid red;

border-radius: 8px;

}

p.round3 {

border: 2px solid red;

border-radius: 12px;

}

</style>

</head>

<body>

<h2>The border-radius Property</h2>

<p>This property is used to add rounded borders to an element:</p>

<p class="normal">Normal border</p>

<p class="round1">Round border</p>

<p class="round2">Rounder border</p>

<p class="round3">Roundest border</p>

<p><b>Note:</b> The "border-radius" property is not supported in IE8 and earlier versions.</p>

</body>

</html>

**CSS**Margins

The CSS margin properties are used to generate space around elements.

The margin properties set the size of the white space outside the border.

With CSS, you have full control over the margins. There are CSS properties for setting the margin for each side of an element (top, right, bottom, and left).

## Margin - Individual Sides

CSS has properties for specifying the margin for each side of an element:

* margin-top
* margin-right
* margin-bottom
* margin-left

All the margin properties can have the following values:

* auto - the browser calculates the margin
* length - specifies a margin in px, pt, cm, etc.
* % - specifies a margin in % of the width of the containing element
* inherit - specifies that the margin should be inherited from the parent element

**Tip:** Negative values are allowed.

The following example sets different margins for all four sides of a <p> element:

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

margin-top: 100px;

margin-bottom: 100px;

margin-right: 150px;

margin-left: 80px;

background-color: lightblue;

}

</style>

</head>

<body>

<h2>Using individual margin properties</h2>

<div>This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.</div>

</body>

</html>

## Margin - Shorthand Property

To shorten the code, it is possible to specify all the margin properties in one property.

The margin property is a shorthand property for the following individual margin properties:

* margin-top
* margin-right
* margin-bottom
* margin-left

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

margin: 100px 150px 100px 80px;

background-color: lightblue;

}

</style>

</head>

<body>

<h2>Using the margin shorthand property</h2>

<div>This div element has a top margin of 100px, a right margin of 150px, a bottom margin of 100px, and a left margin of 80px.</div>

</body>

</html>

So, here is how it works:

If the margin property has four values:

* **margin: 25px 50px 75px 100px;**
  + top margin is 25px
  + right margin is 50px
  + bottom margin is 75px
  + left margin is 100px

If the margin property has three values:

* **margin: 25px 50px 75px;**
  + top margin is 25px
  + right and left margins are 50px
  + bottom margin is 75px

If the margin property has two values:

* **margin: 25px 50px;**
  + top and bottom margins are 25px
  + right and left margins are 50px

If the margin property has one value:

* **margin: 25px;**
  + all four margins are 25px

## The auto Value

You can set the margin property to auto to horizontally center the element within its container.

The element will then take up the specified width, and the remaining space will be split equally between the left and right margins:

<!DOCTYPE html>

<html>

<head>

<style>

div {

width:300px;

margin: auto;

border: 1px solid red;

}

</style>

</head>

<body>

<h2>Use of the auto Value</h2>

<p>You can set the margin property to auto to horizontally center the element within its container.

The element will then take up the specified width, and the remaining space will be split equally between the left and right margins:</p>

<div>

This div will be centered because it has margin: auto;

</div>

</body>

</html>

## The inherit Value

This example lets the left margin be inherited from the parent element:

<!DOCTYPE html>

<html>

<head>

<style>

div.container {

border: 1px solid red;

margin-left: 100px;

}

p.one {

margin-left: inherit;

}

</style>

</head>

<body>

<h2>Use of the inherit value</h2>

<p>Let the left margin be inherited from the parent element:</p>

<div class="container">

<p class="one">This is a paragraph with an inherited left margin (from the div element).</p>

</div>

</body>

</html>

## Margin Collapse

Top and bottom margins of elements are sometimes collapsed into a single margin that is equal to the largest of the two margins.

This does not happen on left and right margins! Only top and bottom margins!

Look at the following example:

<!DOCTYPE html>

<html>

<head>

<style>

h1 {

margin: 0 0 50px 0;

}

h2 {

margin: 20px 0 0 0;

}

</style>

</head>

<body>

<p>In this example the h1 element has a bottom margin of 50px and the h2 element has a top margin of 20px. Then, the vertical margin between h1 and h2 should have been 70px (50px + 20px). However, due to margin collapse, the actual margin ends up being 50px.</p>

<h1>Heading 1</h1>

<h2>Heading 2</h2>

</body>

</html>

In the example above, the <h1> element has a bottom margin of 50px. The <h2> element has a top margin set to 20px.

Common sense would seem to suggest that the vertical margin between the <h1> and the <h2> would be a total of 70px (50px + 20px). But due to margin collapse, the actual margin ends up being 50px.

## All CSS Margin Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| [margin](https://www.w3schools.com/cssref/pr_margin.asp) | A shorthand property for setting the margin properties in one declaration |
| [margin-bottom](https://www.w3schools.com/cssref/pr_margin-bottom.asp) | Sets the bottom margin of an element |
| [margin-left](https://www.w3schools.com/cssref/pr_margin-left.asp) | Sets the left margin of an element |
| [margin-right](https://www.w3schools.com/cssref/pr_margin-right.asp) | Sets the right margin of an element |
| [margin-top](https://www.w3schools.com/cssref/pr_margin-top.asp) | Sets the top margin of an element |

# **CSS Padding**

The CSS padding properties are used to generate space around content.

The padding clears an area around the content (inside the border) of an element.

With CSS, you have full control over the padding. There are CSS properties for setting the padding for each side of an element (top, right, bottom, and left).

## Padding - Individual Sides

CSS has properties for specifying the padding for each side of an element:

* padding-top
* padding-right
* padding-bottom
* padding-left

All the padding properties can have the following values:

* length - specifies a padding in px, pt, cm, etc.
* % - specifies a padding in % of the width of the containing element
* inherit - specifies that the padding should be inherited from the parent element

**Note**:

* **in**: inches — 1in is equal to 2.54cm.
* **cm**: centimeters
* **mm**: millimeters
* **pt**: points — the points used by CSS are equal to 1/72nd of 1in.
* **pc**: picas — 1pc is equal to 12pt.
* **px**: pixel units — 1px is equal to 0.75pt.

The following example sets different padding for all four sides of a <p> element:

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

background-color: lightblue;

padding-top: 50px;

padding-right: 30px;

padding-bottom: 50px;

padding-left: 80px;

}

</style>

</head>

<body>

<h2>Using individual padding properties</h2>

<div>This div element has a top padding of 50px, a right padding of 30px, a bottom padding of 50px, and a left padding of 80px.</div>

</body>

</html>

## Padding - Shorthand Property

To shorten the code, it is possible to specify all the padding properties in one property.

The padding property is a shorthand property for the following individual padding properties:

* padding-top
* padding-right
* padding-bottom
* padding-left

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

background-color: lightblue;

padding: 50px 30px 50px 80px;

}

</style>

</head>

<body>

<h2>Using the padding shorthand property</h2>

<div>This div element has a top padding of 50px, a right padding of 30px, a bottom padding of 50px, and a left padding of 80px.</div>

</body>

</html>

So, here is how it works:

If the padding property has four values:

* **padding: 25px 50px 75px 100px;**
  + top padding is 25px
  + right padding is 50px
  + bottom padding is 75px
  + left padding is 100px

If the padding property has three values:

* **padding: 25px 50px 75px;**
  + top padding is 25px
  + right and left paddings are 50px
  + bottom padding is 75px

If the padding property has two values:

* **padding: 25px 50px;**
  + top and bottom paddings are 25px
  + right and left paddings are 50px

If the padding property has one value:

* **padding: 25px;**
  + all four paddings are 25px

<!DOCTYPE html>

<html>

<head>

<style>

div {

border: 1px solid black;

background-color: lightblue;

}

div.ex1 {

padding: 25px 50px 75px 100px;

}

div.ex2 {

padding: 25px 50px 75px;

}

div.ex3 {

padding: 25px 50px;

}

div.ex4 {

padding: 25px;

}

</style>

</head>

<body>

<h2>Using the padding shorthand property</h2>

<div class="ex1">This div element has a top padding of 25px, a right padding of 50px, a bottom padding of 75px and a left padding of 100px.</div><br>

<div class="ex2">This div element has a top padding of 25px, a left and right padding of 50px, and a bottom padding of 75px.</div><br>

<div class="ex3">This div element has a top and bottom padding of 25px, and a left and right padding of 50px.</div><br>

<div class="ex4">This div element has a top, right, bottom and left paddding of 25px.</div>

</body>

</html>

## All CSS Padding Properties

|  |  |
| --- | --- |
| **Property** | **Description** |
| [padding](https://www.w3schools.com/cssref/pr_padding.asp) | A shorthand property for setting all the padding properties in one declaration |
| [padding-bottom](https://www.w3schools.com/cssref/pr_padding-bottom.asp) | Sets the bottom padding of an element |
| [padding-left](https://www.w3schools.com/cssref/pr_padding-left.asp) | Sets the left padding of an element |
| [padding-right](https://www.w3schools.com/cssref/pr_padding-right.asp) | Sets the right padding of an element |
| [padding-top](https://www.w3schools.com/cssref/pr_padding-top.asp) | Sets the top padding of an element |

# **CSS Height and Width**

## Setting height and width

The height and width properties are used to set the height and width of an element.

The height and width can be set to auto (this is default. Means that the browser calculates the height and width), or be specified in length values, like px, cm, etc., or in percent (%) of the containing block.

<!DOCTYPE html>

<html>

<head>

<style>

div {

height: 200px;

width: 50%;

background-color: powderblue;

}

</style>

</head>

<body>

<h2>Set the height and width of an element</h2>

<p>This div element has a height of 200px and a width of 50%:</p>

<div></div>

</body>

</html>

<!DOCTYPE html>

<html>

<head>

<style>

div {

height: 100px;

width: 500px;

background-color: powderblue;

}

</style>

</head>

<body>

<h2>Set the height and width of an element</h2>

<p>This div element has a height of 100px and a width of 500px:</p>

<div></div>

</body>

</html>

## Setting max-width

The max-width property is used to set the maximum width of an element.

The max-width can be specified in length values, like px, cm, etc., or in percent (%) of the containing block, or set to none (this is default. Means that there is no maximum width).

The problem with the <div> above occurs when the browser window is smaller than the width of the element (500px). The browser then adds a horizontal scrollbar to the page.

Using max-width instead, in this situation, will improve the browser's handling of small windows.

**Tip:** Drag the browser window to smaller than 500px wide, to see the difference between the two divs!

**Note:** The value of the max-width property overrides width.

<!DOCTYPE html>

<html>

<head>

<style>

div {

max-width: 500px;

height: 100px;

background-color: powderblue;

}

</style>

</head>

<body>

<h2>Set the max-width of an element</h2>

<p>This div element has a height of 100px and a max-width of 500px:</p>

<div></div>

<p>Resize the browser window to see the effect.</p>

</body>

</html>