1. TUTORIAL ON CSS

Working with css Example Progs

CSS is a language that describes the style of an HTML document.

CSS describes how HTML elements should be displayed.

Example Prog 1 : Internal css

<!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 Prog</h1>

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

</body>

</html>

**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 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 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 Prog all <p> elements will be center-aligned, with a red text color:

Example Prog 2 :

p {  
    color: red;  
    text-align: center;  
}

**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):

Ex3 :

<!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":

Example Prog 4:

<!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>

**Note:** An id name cannot start with a number!

**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 Prog below, all HTML elements with class="center" will be red and center-aligned:

Example Prog 5:

<!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 Prog below, only <p> elements with class="center" will be center-aligned:

Example Prog 6:

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

Example Prog 7:

<!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:

h1 {  
    text-align: center;  
    color: red;  
}  
  
h2 {  
    text-align: center;  
    color: red;  
}  
  
p {  
    text-align: center;  
    color: red;  
}

It will be better to group the selectors, to minimize the code.

To group selectors, separate each selector with a comma.

In the Example Prog below we have grouped the selectors from the code above:

Example Prog 8 :

<!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:

Example Prog 9 :

<!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>

1. 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:

Example Prog 10 :

<!DOCTYPE html>

<html>

<head>

<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>

An external style sheet can be written in any text editor. The file should not contain any html tags. The style sheet file must be saved with a .css extension.

Here is how the "myStyle.css" looks:

body {  
    background-color: lightblue;  
}  
  
h1 {  
    color: navy;  
    margin-left: 20px;  
}

Note: Do not add a space between the property value and the unit (such as margin-left: 20 px;). The correct way is: margin-left: 20px;

**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:

Example Prog 11 :

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

Example Prog 12 :

<!DOCTYPE html>

<html>

<body>

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

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

</body>

</html>

**Multiple Style Sheets :**

If some properties have been defined for the same selector (element) in different style sheets, the value from the last read style sheet will be used.

Example Prog :

Assume that an external style sheet has the following style for the <h1> element:

h1 {

color: navy;

}

then, assume that an internal style sheet also has the following style for the <h1> element:

h1 {

color: orange;

}

If the internal style is defined after the link to the external style sheet, the <h1> elements will be "orange":

Example Prog 13 :

<!DOCTYPE html>

<html>

<head>

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

<style>

h1 {

color: orange;

}

</style>

</head>

<body>

<h1>This is a heading</h1>

<p>The style of this document is a combination of an external stylesheet, and internal style</p>

</body>

</html>

However, if the internal style is defined before the link to the external style sheet, the <h1> elements will be "navy":

Example Prog 14 :

<!DOCTYPE html>

<html>

<head>

<style>

h1 {

color: orange;

}

</style>

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

</head>

<body>

<h1>This is a heading</h1>

<p>The style of this document is a combination of an external stylesheet, and internal style</p>

</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.

Example Prog 14 :

<!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 Prog, 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>

1. 3.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:

Example Prog 15 :

<!DOCTYPE html>

<html>

<body>

<h2>Color Names Example Progs</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>

**Note:** Color names are case-insensitive: "Red" is the same as "red" or "RED".

HTML and CSS supports 140 standard colors.

|  |  |
| --- | --- |
| AliceBlue #F0F8FF  AntiqueWhite #FAEBD7  Aqua #00FFFF  Aquamarine #7FFFD4  Azure #F0FFFF  Beige #F5F5DC  Bisque #FFE4C4  Black #000000  BlanchedAlmond #FFEBCD  Blue #0000FF  BlueViolet #8A2BE2  Brown #A52A2A  BurlyWood #DEB887  CadetBlue #5F9EA0  Chartreuse #7FFF00  Chocolate #D2691E  Coral #FF7F50  CornflowerBlue #6495ED  Cornsilk #FFF8DC  Crimson #DC143C  Cyan #00FFFF  DarkBlue #00008B  DarkCyan #008B8B  DarkGoldenRod #B8860B  DarkGray #A9A9A9  DarkGrey #A9A9A9  DarkGreen #006400  DarkKhaki #BDB76B  DarkMagenta #8B008B  DarkOliveGreen #556B2F  DarkOrange #FF8C00  DarkOrchid #9932CC  DarkRed #8B0000  DarkSalmon #E9967A  DarkSeaGreen #8FBC8F  DarkSlateBlue #483D8B  DarkSlateGray #2F4F4F  DarkSlateGrey #2F4F4F  DarkTurquoise #00CED1  DarkViolet #9400D3  DeepPink #FF1493  DeepSkyBlue #00BFFF  DimGray #696969  DimGrey #696969  DodgerBlue #1E90FF  FireBrick #B22222  FloralWhite #FFFAF0  ForestGreen #228B22  Fuchsia #FF00FF  Gainsboro #DCDCDC  GhostWhite #F8F8FF  Gold #FFD700  GoldenRod #DAA520  Gray #808080  Grey #808080  Green #008000  GreenYellow #ADFF2F  HoneyDew #F0FFF0  HotPink #FF69B4  IndianRed #CD5C5C  Indigo #4B0082  Ivory #FFFFF0  Khaki #F0E68C  Lavender #E6E6FA  LavenderBlush #FFF0F5  LawnGreen #7CFC00  LemonChiffon #FFFACD  LightBlue #ADD8E6  LightCoral #F08080  LightCyan #E0FFFF  LightGoldenRodYellow #FAFAD2  LightGray #D3D3D3  LightGrey #D3D3D3  LightGreen #90EE90 | LightPink #FFB6C1  LightSalmon #FFA07A  LightSeaGreen #20B2AA  LightSkyBlue #87CEFA  LightSlateGray #778899  LightSlateGrey #778899  LightSteelBlue #B0C4DE  LightYellow #FFFFE0  Lime #00FF00  LimeGreen #32CD32  Linen #FAF0E6    Magenta #FF00FF  Maroon #800000  MediumAquaMarine #66CDAA  MediumBlue #0000CD  MediumOrchid #BA55D3  MediumPurple #9370DB  MediumSeaGreen #3CB371  MediumSlateBlue #7B68EE  MediumSpringGreen #00FA9A  MediumTurquoise #48D1CC  MediumVioletRed #C71585  MidnightBlue #191970  MintCream #F5FFFA  MistyRose #FFE4E1  Moccasin #FFE4B5  NavajoWhite #FFDEAD  Navy #000080  OldLace #FDF5E6  Olive #808000  OliveDrab #6B8E23  Orange #FFA500  OrangeRed #FF4500  Orchid #DA70D6  PaleGoldenRod #EEE8AA  PaleGreen #98FB98  PaleTurquoise #AFEEEE  PaleVioletRed #DB7093  PapayaWhip #FFEFD5  PeachPuff #FFDAB9  Peru #CD853F  Pink #FFC0CB  Plum #DDA0DD  PowderBlue #B0E0E6  Purple #800080  RebeccaPurple #663399  Red #FF0000  RosyBrown #BC8F8F  RoyalBlue #4169E1  SaddleBrown #8B4513  Salmon #FA8072  SandyBrown #F4A460  SeaGreen #2E8B57  SeaShell #FFF5EE  Sienna #A0522D  Silver #C0C0C0  SkyBlue #87CEEB  SlateBlue #6A5ACD  SlateGray #708090  SlateGrey #708090  Snow #FFFAFA  SpringGreen #00FF7F  SteelBlue #4682B4  Tan #D2B48C  Teal #008080  Thistle #D8BFD8  Tomato #FF6347  Turquoise #40E0D0  Violet #EE82EE  Wheat #F5DEB3  White #FFFFFF  WhiteSmoke #F5F5F5  Yellow #FFFF00    YellowGreen #9ACD32 |

**RGB (Red, Green, Blue) :**

RGB color values can be specified using this formula: rgb(red, green, blue).

Each parameter (red, green, blue) defines the intensity of the color between 0 and 255.

For Example Prog, rgb(255,0,0) is displayed as red, because red is set to its highest value (255) and the others are set to 0. Experiment by mixing the RGB values below:

Example Prog 16 :

<!DOCTYPE html>

<html>

<body>

<h2>RGB Color Example Progs</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>

Shades of grey are often defined using equal values for all the 3 light sources:

Example Prog 17 :

<!DOCTYPE html>

<html>

<body>

<h2>RGB Color Example Progs</h2>

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

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

</h2>

<h2 style="background-color:rgb(128, 128, 128);color:white">

Background-color set by using rgb(128,128,128)

</h2>

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

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

</h2>

</body>

</html>

**Hexadecimal Colors :**

RGB values can also be specified using hexadecimal color values in the form: #RRGGBB, where RR (red), GG (green) and BB (blue) are hexadecimal values between 00 and FF (same as decimal 0-255).

For Example Prog, #FF0000 is displayed as red, because red is set to its highest value (FF) and the others are set to the lowest value (00).

Note: HEX values are case-insensitive: "#ff0000" is the same as "FF0000".

Example Prog 18:

<!DOCTYPE html>

<html>

<body>

<h2>HEX Color Example Progs</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>

Shades of grey are often defined using equal values for all the 3 light sources:

Example Prog 18:

<!DOCTYPE html>

<html>

<body>

<h2>HEX Color Example Progs</h2>

<h2 style="background-color:#000000;color:white">

Background-color set by using #000000

</h2>

<h2 style="background-color:#808080;color:white">

Background-color set by using #808080

</h2>

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

Background-color set by using #FFFFFF

</h2>

</body>

</html>

1. Background Colors

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:

Example Prog 19 :

<!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>

With CSS, a color is most often specified by:

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

In the Example Prog below, the <h1>, <p>, and <div> elements have different background colors:

Example Prog 20 :

<!DOCTYPE html>

<html>

<head>

<style>

h1 {

background-color: green;

}

div {

background-color: lightblue;

}

p {

background-color: yellow;

}

</style>

</head>

<body>

<h1>CSS background-color Example Prog!</h1>

<div>

This is a text inside a div element.

<p>This paragraph has its own background color.</p>

We are still in the div element.

</div>

</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:

Example Prog 21 :

<!DOCTYPE html>

<html>

<head>

<style>

body {

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

}

</style>

</head>

<body>

<h1>Hello World!</h1>

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

</body>

</html>

Below is an Example Prog of a bad combination of text and background image. The text is hardly readable:

Example Prog 22 :

<!DOCTYPE html>

<html>

<head>

<style>

body {

background-image: url("bgdesert.jpg");

}

</style>

</head>

<body>

<h1>Hello World!</h1>

<p>This text is not easy to read on this background image.</p>

</body>

</html>

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

Example Prog 23 :

<!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>

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

**Background Image - Set position and no-repeat :**

Showing the background image only once is also specified by the background-repeat property:

Example Prog 24 :

<!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 Prog.</p>

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

</body>

</html>

In the Example Prog 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 :

Example Prog 25 :

<!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 Prog.</p>

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

<p>In this Example Prog 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:

Example Prog 26:

<!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>

Example Prog 27 :

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

</body>

</html>

When using the shorthand property the order of the property values is:

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

It does not matter if one of the property values is missing, as long as the other ones are in this order.

|  |  |
| --- | --- |
| **Property** | **Description** |
| Background | Sets all the background properties in one declaration |
| background-attachment | Sets whether a background image is fixed or scrolls with the rest of the page |
| background-color | Sets the background color of an element |
| background-image | Sets the background image for an element |
| background-position | Sets the starting position of a background image |
| background-repeat | Sets how a background image will be repeated |

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 Prog :

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;}

Example Prog 28 :

<!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>

**Note:** None of the OTHER CSS border properties described below will have ANY effect unless the border-style property is set!

**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).

Example Prog 29 :

<!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.

Example Prog 30 :

<!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 Example Progs above you have seen that it is possible to specify a different border for each side.

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

Example Prog 31 :

<!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 Prog above gives the same result as this:

Example Prog 32 :

<!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 Prog above. However, it also works with border-width and border-color.

**Border - Shorthand Property :**

As you can see from the Example Progs 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

Example Prog :

p {  
    border: 5px solid red;  
}

Example Prog 32 :

<!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:

**Left Border**

p {  
    border-left: 6px solid red;  
    background-color: lightgrey;  
}

Example Prog 33 :

<!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>

**Bottom Border**

p {  
    border-bottom: 6px solid red;  
    background-color: lightgrey;  
}

Example Prog 34 :

<!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>

**Rounded Borders :**

The border-radius property is used to add rounded borders to an element:

Example Prog 35 :

<!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>

Note: The border-radius property is not supported in IE8 and earlier versions.

All CSS Border Properties

|  |  |
| --- | --- |
| Property | Description |
| Border | Sets all the border properties in one declaration |
| border-bottom | Sets all the bottom border properties in one declaration |
| border-bottom-color | Sets the color of the bottom border |
| border-bottom-style | Sets the style of the bottom border |
| border-bottom-width | Sets the width of the bottom border |
| border-color | Sets the color of the four borders |
| border-left | Sets all the left border properties in one declaration |
| border-left-color | Sets the color of the left border |
| border-left-style | Sets the style of the left border |
| border-left-width | Sets the width of the left border |
| border-radius | Sets all the four border-\*-radius properties for rounded corners |
| border-right | Sets all the right border properties in one declaration |
| border-right-color | Sets the color of the right border |
| border-right-style | Sets the style of the right border |
| border-right-width | Sets the width of the right border |
| border-style | Sets the style of the four borders |
| border-top | Sets all the top border properties in one declaration |
| border-top-color | Sets the color of the top border |
| border-top-style | Sets the style of the top border |
| border-top-width | Sets the width of the top border |
| border-width | Sets the width of the four borders |