**What You Should Already Know**

Before you continue you should have a basic understanding of the following:

* HTML / XHTML

If you want to study these subjects first, find the tutorials on our [Home page](http://w3schools.com/default.asp).

**What is CSS?**

* **CSS** stands for **C**ascading **S**tyle **S**heets
* Styles define **how to display** HTML elements
* Styles were added to HTML 4.0 **to solve a problem**
* **External Style Sheets** can save a lot of work
* External Style Sheets are stored in **CSS files**

**CSS Demo**

An HTML document can be displayed with different styles: [See how it works](http://w3schools.com/css/demo_default.htm)

**Styles Solved a Big Problem**

HTML was never intended to contain tags for formatting a document.

HTML was intended to define the content of a document, like:

<h1>This is a heading</h1>

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

When tags like <font>, and color attributes were added to the HTML 3.2 specification, it started a nightmare for web developers. Development of large web sites, where fonts and color information were added to every single page, became a long and expensive process.

To solve this problem, the World Wide Web Consortium (W3C) created CSS.

In HTML 4.0, all formatting could be removed from the HTML document, and stored in a separate CSS file.

All browsers support CSS today.

**CSS Saves a Lot of Work!**

CSS defines HOW HTML elements are to be displayed.

Styles are normally saved in external .css files. External style sheets enable you to change the appearance and layout of all the pages in a Web site, just by editing one single file!

**CSS Syntax**

A CSS rule has two main parts: a selector, and one or more declarations:

The selector is normally the HTML element you want to style.

Each declaration consists of a property and a value.

The property is the style attribute you want to change. Each property has a value.

**CSS Example**

A CSS declaration always ends with a semicolon, and declaration groups are surrounded by curly brackets:

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

To make the CSS more readable, you can put one declaration on each line, like this:

**Example**

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

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_syntax1)

**CSS Comments**

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

A CSS comment begins with "/\*", and ends with "\*/", like this:

/\*This is a comment\*/  
p  
{  
text-align:center;  
/\*This is another comment\*/  
color:black;  
font-family:arial;  
}

**he id and class Selectors**

In addition to setting a style for a HTML element, CSS allows you to specify your own selectors called "id" and "class".

**The id Selector**

The id selector is used to specify a style for a single, unique element.

The id selector uses the id attribute of the HTML element, and is defined with a "#".

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

**Example**

#para1  
{  
text-align:center;  
color:red;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_syntax_id)

Do **NOT** start an ID name with a number! It will not work in Mozilla/Firefox.

**The class Selector**

The class selector is used to specify a style for a group of elements. Unlike the id selector, the class selector is most often used on several elements.

This allows you to set a particular style for many HTML elements with the same class.

The class selector uses the HTML class attribute, and is defined with a "."

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

**Example**

.center {text-align:center;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_syntax_class)

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

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

**Example**

p.center {text-align:center;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_syntax_element_class)

Do **NOT** start a class name with a number! This is only supported in Internet Explorer.

## 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

An external style sheet is ideal when the style is applied to many pages. With an external style sheet, you can change the look of an entire Web site by changing one file. Each page must link to the style sheet using the <link> tag. The <link> tag goes inside the head section:

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

An external style sheet can be written in any text editor. The file should not contain any html tags. Your style sheet should be saved with a .css extension. An example of a style sheet file is shown below:

hr {color:sienna;}  
p {margin-left:20px;}  
body {background-image:url("images/back40.gif");}

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 should be used when a single document has a unique style. You define internal styles in the head section of an HTML page, by using the <style> tag, like this:

<head>  
<style type="text/css">  
hr {color:sienna;}  
p {margin-left:20px;}  
body {background-image:url("images/back40.gif");}  
</style>  
</head>

## Inline Styles

An inline style loses many of the advantages of style sheets by mixing content with presentation. Use this method sparingly!

To use inline styles you use the style attribute in the relevant tag. The style attribute can contain any CSS property. The example shows how to change the color and the left margin of a paragraph:

<p style="color:sienna;margin-left:20px">This is a paragraph.</p>

## Multiple Style Sheets

If some properties have been set for the same selector in different style sheets, the values will be inherited from the more specific style sheet.

For example, an external style sheet has these properties for the h3 selector:

h3  
{  
color:red;  
text-align:left;  
font-size:8pt;  
}

And an internal style sheet has these properties for the h3 selector:

h3  
{  
text-align:right;  
font-size:20pt;  
}

If the page with the internal style sheet also links to the external style sheet the properties for h3 will be:

color:red;  
text-align:right;  
font-size:20pt;

The color is inherited from the external style sheet and the text-alignment and the font-size is replaced by the internal style sheet.

## Multiple Styles Will Cascade into One

Styles can be specified:

* inside an HTML element
* inside the head section of an HTML page
* in an external CSS file

**Tip:** Even multiple external style sheets can be referenced inside a single HTML document.

### 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 four has the highest priority:

1. Browser default
2. External style sheet
3. Internal style sheet (in the head section)
4. Inline style (inside an HTML element)

So, an inline style (inside an 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 in a browser (a default value).

**Note:** If the link to the external style sheet is placed after the internal style sheet in HTML <head>, the external style sheet will override the internal style sheet!

**Background Color**

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

The background color of a page is defined in the body selector:

**Example**

body {background-color:#b0c4de;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-color_body)

With CSS, a color is most often specified by:

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

Look at [CSS Color Values](http://w3schools.com/cssref/css_colors_legal.asp) for a complete list of possible color values.

In the example below, the h1, p, and div elements have different background colors:

**Example**

h1 {background-color:#6495ed;}  
p {background-color:#e0ffff;}  
div {background-color:#b0c4de;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-color_elements)

**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**

body {background-image:url('paper.gif');}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-image)

Below is an example of a bad combination of text and background image. The text is almost not readable:

**Example**

body {background-image:url('bgdesert.jpg');}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-image_bad)

**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:

**Example**

body  
{  
background-image:url('gradient2.png');  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-image_gradient1)

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

**Example**

body  
{  
background-image:url('gradient2.png');  
background-repeat:repeat-x;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-image_gradient2)

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

When using a background image, use an image that does not disturb the text.

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

**Example**

body  
{  
background-image:url('img\_tree.png');  
background-repeat:no-repeat;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-image_norepeat)

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:

**Example**

body  
{  
background-image:url('img\_tree.png');  
background-repeat:no-repeat;  
background-position:right top;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background-image_position)

**Background - Shorthand property**

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

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

The shorthand property for background is simply "background":

**Example**

body {background:#ffffff url('img\_tree.png') no-repeat right top;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_background_shorthand)

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 ones that are present are in this order.

This example uses more advanced CSS. Take a look: [Advanced example](http://w3schools.com/css/tryit.asp?filename=trycss_background_shorthand2)

**More Examples**

[How to set a fixed background image](http://w3schools.com/css/tryit.asp?filename=trycss_background-attachment)  
This example demonstrates how to set a fixed background image. The image will not scroll with the rest of the page.

**All CSS Background Properties**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [background](http://w3schools.com/cssref/css3_pr_background.asp) | Sets all the background properties in one declaration |
| [background-attachment](http://w3schools.com/cssref/pr_background-attachment.asp) | Sets whether a background image is fixed or scrolls with the rest of the page |
| [background-color](http://w3schools.com/cssref/pr_background-color.asp) | Sets the background color of an element |
| [background-image](http://w3schools.com/cssref/pr_background-image.asp) | Sets the background image for an element |
| [background-position](http://w3schools.com/cssref/pr_background-position.asp) | Sets the starting position of a background image |
| [background-repeat](http://w3schools.com/cssref/pr_background-repeat.asp) | Sets how a background image will be repeated |

**Text Color**

The color property is used to set the color of the text.

With CSS, a color is most often specified by:

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

Look at [CSS Color Values](http://w3schools.com/cssref/css_colors_legal.asp) for a complete list of possible color values.

The default color for a page is defined in the body selector.

**Example**

body {color:blue;}  
h1 {color:#00ff00;}  
h2 {color:rgb(255,0,0);}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_color)

For W3C compliant CSS: If you define the color property, you must also define the background-color property.

**Text Alignment**

The text-align property is used to set the horizontal alignment of a text.

Text can be centered, or aligned to the left or right, or justified.

When text-align is set to "justify", each line is stretched so that every line has equal width, and the left and right margins are straight (like in magazines and newspapers).

**Example**

h1 {text-align:center;}  
p.date {text-align:right;}  
p.main {text-align:justify;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_text-align_all)

**Text Decoration**

The text-decoration property is used to set or remove decorations from text.

The text-decoration property is mostly used to remove underlines from links for design purposes:

**Example**

a {text-decoration:none;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_text-decoration_link)

It can also be used to decorate text:

**Example**

h1 {text-decoration:overline;}  
h2 {text-decoration:line-through;}  
h3 {text-decoration:underline;}  
h4 {text-decoration:blink;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_text-decoration)

It is not recommended to underline text that is not a link, as this often confuses users.

**Text Transformation**

The text-transform property is used to specify uppercase and lowercase letters in a text.

It can be used to turn everything into uppercase or lowercase letters, or capitalize the first letter of each word.

**Example**

p.uppercase {text-transform:uppercase;}  
p.lowercase {text-transform:lowercase;}  
p.capitalize {text-transform:capitalize;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_text-transform)

**Text Indentation**

The text-indentation property is used to specify the indentation of the first line of a text.

**Example**

p {text-indent:50px;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_text-indent)

**More Examples**

[Specify the space between characters](http://w3schools.com/css/tryit.asp?filename=trycss_letter-spacing)  
This example demonstrates how to increase or decrease the space between characters.

[Specify the space between lines](http://w3schools.com/css/tryit.asp?filename=trycss_line-height)  
This example demonstrates how to specify the space between the lines in a paragraph.

[Set the text direction of an element](http://w3schools.com/css/tryit.asp?filename=trycss_text_direction)  
This example demonstrates how to change the text direction of an element.

[Increase the white space between words](http://w3schools.com/css/tryit.asp?filename=trycss_text_word-spacing)  
This example demonstrates how to increase the white space between words in a paragraph.

[Disable text wrapping inside an element](http://w3schools.com/css/tryit.asp?filename=trycss_text_white-space)  
This example demonstrates how to disable text wrapping inside an element.

[Vertical alignment of an image](http://w3schools.com/css/tryit.asp?filename=trycss_vertical-align)  
This example demonstrates how to set the vertical align of an image in a text.

**All CSS Text Properties**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [color](http://w3schools.com/cssref/pr_text_color.asp) | Sets the color of text |
| [direction](http://w3schools.com/cssref/pr_text_direction.asp) | Specifies the text direction/writing direction |
| [letter-spacing](http://w3schools.com/cssref/pr_text_letter-spacing.asp) | Increases or decreases the space between characters in a text |
| [line-height](http://w3schools.com/cssref/pr_dim_line-height.asp) | Sets the line height |
| [text-align](http://w3schools.com/cssref/pr_text_text-align.asp) | Specifies the horizontal alignment of text |
| [text-decoration](http://w3schools.com/cssref/pr_text_text-decoration.asp) | Specifies the decoration added to text |
| [text-indent](http://w3schools.com/cssref/pr_text_text-indent.asp) | Specifies the indentation of the first line in a text-block |
| text-shadow | Specifies the shadow effect added to text |
| [text-transform](http://w3schools.com/cssref/pr_text_text-transform.asp) | Controls the capitalization of text |
| unicode-bidi |  |
| [vertical-align](http://w3schools.com/cssref/pr_pos_vertical-align.asp) | Sets the vertical alignment of an element |
| [white-space](http://w3schools.com/cssref/pr_text_white-space.asp) | Specifies how white-space inside an element is handled |
| [word-spacing](http://w3schools.com/cssref/pr_text_word-spacing.asp) | Increases or decreases the space between words in a text |

**Difference Between Serif and Sans-serif Fonts**

On computer screens, sans-serif fonts are considered easier to read than serif fonts.

**CSS Font Families**

In CSS, there are two types of font family names:

* **generic family** - a group of font families with a similar look (like "Serif" or "Monospace")
* **font family** - a specific font family (like "Times New Roman" or "Arial")

|  |  |  |
| --- | --- | --- |
| **Generic family** | **Font family** | **Description** |
| Serif | Times New Roman Georgia | Serif fonts have small lines at the ends on some characters |
| Sans-serif | Arial Verdana | "Sans" means without - these fonts do not have the lines at the ends of characters |
| Monospace | Courier New Lucida Console | All monospace characters have the same width |

**Font Family**

The font family of a text is set with the font-family property.

The font-family property should hold several font names as a "fallback" system. If the browser does not support the first font, it tries the next font.

Start with the font you want, and end with a generic family, to let the browser pick a similar font in the generic family, if no other fonts are available.

**Note**: If the name of a font family is more than one word, it must be in quotation marks, like font-family: "Times New Roman".

More than one font family is specified in a comma-separated list:

**Example**

p{font-family:"Times New Roman", Times, serif;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_font-family)

For more commonly used font combinations, look at our [Web Safe Font Combinations](http://w3schools.com/cssref/css_websafe_fonts.asp).

**Font Style**

The font-style property is mostly used to specify italic text.

This property has three values:

* normal - The text is shown normally
* italic - The text is shown in italics
* oblique - The text is "leaning" (oblique is very similar to italic, but less supported)

**Example**

p.normal {font-style:normal;}  
p.italic {font-style:italic;}  
p.oblique {font-style:oblique;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_font-style)

**Font Size**

The font-size property sets the size of the text.

Being able to manage the text size is important in web design. However, you should not use font size adjustments to make paragraphs look like headings, or headings look like paragraphs.

Always use the proper HTML tags, like <h1> - <h6> for headings and <p> for paragraphs.

The font-size value can be an absolute, or relative size.

Absolute size:

* Sets the text to a specified size
* Does not allow a user to change the text size in all browsers (bad for accessibility reasons)
* Absolute size is useful when the physical size of the output is known

Relative size:

* Sets the size relative to surrounding elements
* Allows a user to change the text size in browsers

If you do not specify a font size, the default size for normal text, like paragraphs, is 16px (16px=1em).

**Set Font Size With Pixels**

Setting the text size with pixels gives you full control over the text size:

**Example**

h1 {font-size:40px;}  
h2 {font-size:30px;}  
p {font-size:14px;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_font-size_px)

The example above allows Internet Explorer 9, Firefox, Chrome, Opera, and Safari to resize the text.

**Note:** The example above does not work in IE, prior version 9.

The text can be resized in all browsers using the zoom tool (however, this resizes the entire page, not just the text).

**Set Font Size With Em**

To avoid the resizing problem with older versions of Internet Explorer, many developers use em instead of pixels.

The em size unit is recommended by the W3C.

1em is equal to the current font size. The default text size in browsers is 16px. So, the default size of 1em is 16px.

The size can be calculated from pixels to em using this formula: *pixels*/16=*em*

**Example**

h1 {font-size:2.5em;} /\* 40px/16=2.5em \*/  
h2 {font-size:1.875em;} /\* 30px/16=1.875em \*/  
p {font-size:0.875em;} /\* 14px/16=0.875em \*/

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_font-size_em)

In the example above, the text size in em is the same as the previous example in pixels. However, with the em size, it is possible to adjust the text size in all browsers.

Unfortunately, there is still a problem with older versions of IE. The text becomes larger than it should when made larger, and smaller than it should when made smaller.

**Use a Combination of Percent and Em**

The solution that works in all browsers, is to set a default font-size in percent for the <body> element:

**Example**

body {font-size:100%;}  
h1 {font-size:2.5em;}  
h2 {font-size:1.875em;}  
p {font-size:0.875em;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_font-size_percent_em)

Our code now works great! It shows the same text size in all browsers, and allows all browsers to zoom or resize the text!

**More Examples**

[Set the boldness of the font](http://w3schools.com/css/tryit.asp?filename=trycss_font-weight)  
This example demonstrates how to set the boldness of a font.

[Set the variant of the font](http://w3schools.com/css/tryit.asp?filename=trycss_font-variant)  
This example demonstrates how to set the variant of a font.

[All the font properties in one declaration](http://w3schools.com/css/tryit.asp?filename=trycss_font)  
This example demonstrates how to use the shorthand property for setting all of the font properties in one declaration.

**All CSS Font Properties**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [font](http://w3schools.com/cssref/pr_font_font.asp) | Sets all the font properties in one declaration |
| [font-family](http://w3schools.com/cssref/pr_font_font-family.asp) | Specifies the font family for text |
| [font-size](http://w3schools.com/cssref/pr_font_font-size.asp) | Specifies the font size of text |
| [font-style](http://w3schools.com/cssref/pr_font_font-style.asp) | Specifies the font style for text |
| [font-variant](http://w3schools.com/cssref/pr_font_font-variant.asp) | Specifies whether or not a text should be displayed in a small-caps font |
| [font-weight](http://w3schools.com/cssref/pr_font_weight.asp) | Specifies the weight of a font |

Links can be styled with any CSS property (e.g. color, font-family, background, etc.).

Special for links are that they can be styled differently depending on what state they are in.

The four links states are:

* a:link - a normal, unvisited link
* a:visited - a link the user has visited
* a:hover - a link when the user mouses over it
* a:active - a link the moment it is clicked

**Example**

a:link {color:#FF0000;}      /\* unvisited link \*/  
a:visited {color:#00FF00;}  /\* visited link \*/  
a:hover {color:#FF00FF;}  /\* mouse over link \*/  
a:active {color:#0000FF;}  /\* selected link \*/

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_link)

When setting the style for several link states, there are some order rules:

* a:hover MUST come after a:link and a:visited
* a:active MUST come after a:hover

**Common Link Styles**

In the example above the link changes color depending on what state it is in.

Lets go through some of the other common ways to style links:

**Text Decoration**

The text-decoration property is mostly used to remove underlines from links:

**Example**

a:link {text-decoration:none;}  
a:visited {text-decoration:none;}  
a:hover {text-decoration:underline;}  
a:active {text-decoration:underline;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_link_decoration)

**Background Color**

The background-color property specifies the background color for links:

**Example**

a:link {background-color:#B2FF99;}  
a:visited {background-color:#FFFF85;}  
a:hover {background-color:#FF704D;}  
a:active {background-color:#FF704D;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_link_background)

**More Examples**

[Add different styles to hyperlinks](http://w3schools.com/css/tryit.asp?filename=trycss_link2)  
This example demonstrates how to add other styles to hyperlinks.

[Advanced - Create link boxes](http://w3schools.com/css/tryit.asp?filename=trycss_link_advanced)  
This example demonstrates a more advanced example where we combine several CSS properties to display links as boxes.

**List**

In HTML, there are two types of lists:

* unordered lists - the list items are marked with bullets
* ordered lists - the list items are marked with numbers or letters

With CSS, lists can be styled further, and images can be used as the list item marker.

**Different List Item Markers**

The type of list item marker is specified with the list-style-type property:

**Example**

ul.a {list-style-type: circle;}  
ul.b {list-style-type: square;}  
  
ol.c {list-style-type: upper-roman;}  
ol.d {list-style-type: lower-alpha;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_list-style-type_ex)

Some of the values are for unordered lists, and some for ordered lists.

**An Image as The List Item Marker**

To specify an image as the list item marker, use the list-style-image property:

**Example**

ul  
{  
list-style-image: url('sqpurple.gif');  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_list-style-image)

The example above does not display equally in all browsers. IE and Opera will display the image-marker a little bit higher than Firefox, Chrome, and Safari.

If you want the image-marker to be placed equally in all browsers, a crossbrowser solution is explained below.

**Crossbrowser Solution**

The following example displays the image-marker equally in all browsers:

**Example**

ul  
{  
list-style-type: none;  
padding: 0px;  
margin: 0px;  
}  
ul li  
{  
background-image: url(sqpurple.gif);  
background-repeat: no-repeat;  
background-position: 0px 5px;   
padding-left: 14px;   
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_list-style-image_crossbrow)

Example explained:

* For ul:
  + Set the list-style-type to none to remove the list item marker
  + Set both padding and margin to 0px (for cross-browser compatibility)
* For all li in ul:
  + Set the URL of the image, and show it only once (no-repeat)
  + Position the image where you want it (left 0px and down 5px)
  + Position the text in the list with padding-left

**List - Shorthand property**

It is also possible to specify all the list properties in one, single property. This is called a shorthand property.

The shorthand property used for lists, is the list-style property:

**Example**

ul  
{  
list-style: square url("sqpurple.gif");  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_list-style)

When using the shorthand property, the order of the values are:

* list-style-type
* list-style-position (for a description, see the CSS properties table below)
* list-style-image

It does not matter if one of the values above are missing, as long as the rest are in the specified order.

**More Examples**

[All the different list-item markers for lists](http://w3schools.com/css/tryit.asp?filename=trycss_list-style-type_all)  
This example demonstrates all the different list-item markers in CSS.

**All CSS List Properties**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [list-style](http://w3schools.com/cssref/pr_list-style.asp) | Sets all the properties for a list in one declaration |
| [list-style-image](http://w3schools.com/cssref/pr_list-style-image.asp) | Specifies an image as the list-item marker |
| [list-style-position](http://w3schools.com/cssref/pr_list-style-position.asp) | Specifies if the list-item markers should appear inside or outside the content flow |
| [list-style-type](http://w3schools.com/cssref/pr_list-style-type.asp) | Specifies the type of list-item marker |

**Table Borders**

To specify table borders in CSS, use the border property.

The example below specifies a black border for table, th, and td elements:

**Example**

table, th, td  
{  
border: 1px solid black;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_border)

Notice that the table in the example above has double borders. This is because both the table and the th/td elements have separate borders.

To display a single border for the table, use the border-collapse property.

**Collapse Borders**

The border-collapse property sets whether the table borders are collapsed into a single border or separated:

**Example**

table  
{  
border-collapse:collapse;  
}  
table,th, td  
{  
border: 1px solid black;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_border-collapse)

**Table Width and Height**

Width and height of a table is defined by the width and height properties.

The example below sets the width of the table to 100%, and the height of the th elements to 50px:

**Example**

table   
{  
width:100%;  
}  
th  
{  
height:50px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_width)

**Table Text Alignment**

The text in a table is aligned with the text-align and vertical-align properties.

The text-align property sets the horizontal alignment, like left, right, or center:

**Example**

td  
{  
text-align:right;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_align)

The vertical-align property sets the vertical alignment, like top, bottom, or middle:

**Example**

td  
{  
height:50px;  
vertical-align:bottom;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_vertical-align)

**Table Padding**

To control the space between the border and content in a table, use the padding property on td and th elements:

**Example**

td  
{  
padding:15px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_padding)

**Table Color**

The example below specifies the color of the borders, and the text and background color of th elements:

**Example**

table, td, th  
{  
border:1px solid green;  
}  
th  
{  
background-color:green;  
color:white;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_table_color)

**More Examples**

[Make a fancy table](http://w3schools.com/css/tryit.asp?filename=trycss_table_fancy)  
This example demonstrates how to create a fancy table.

[Set the position of the table caption](http://w3schools.com/css/tryit.asp?filename=trycss_table_caption-side)  
This example demonstrates how to position the table caption.

**The CSS Box Model**

All HTML elements can be considered as boxes. In CSS, the term "box model" is used when talking about design and layout.

The CSS box model is essentially a box that wraps around HTML elements, and it consists of: margins, borders, padding, and the actual content.

The box model allows us to place a border around elements and space elements in relation to other elements.

The image below illustrates the box model:

Explanation of the different parts:

* **Margin** - Clears an area around the border. The margin does not have a background color, it is completely transparent
* **Border** - A border that goes around the padding and content. The border is affected by the background color of the box
* **Padding** - Clears an area around the content. The padding is affected by the background color of the box
* **Content** - The content of the box, where text and images appear

In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.

**Width and Height of an Element**

**Important:** When you set the width and height properties of an element with CSS, you just set the width and height of the **content area**. To calculate the full size of an element, you must also add the padding, borders and margins.

The total width of the element in the example below is 300px:

width:250px;  
padding:10px;  
border:5px solid gray;  
margin:10px;

Let's do the math:  
250px (width)  
+ 20px (left and right padding)  
+ 10px (left and right border)  
+ 20px (left and right margin)  
= 300px

Assume that you had only 250px of space. Let's make an element with a total width of 250px:

**Example**

width:220px;  
padding:10px;  
border:5px solid gray;  
margin:0px;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_boxmodel_width)

The total width of an element should be calculated like this:

Total element width = width + left padding + right padding + left border + right border + left margin + right margin

The total height of an element should be calculated like this:

Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

**Browsers Compatibility Issue**

The example above does not display properly in IE8 and earlier versions.

IE8 and earlier versions includes padding and border in the width, if a **DOCTYPE is NOT declared**.

To fix this problem, just add a DOCTYPE to the HTML page:

**Example**

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">  
<html>  
<head>  
<style type="text/css">  
div.ex  
{  
width:220px;  
padding:10px;  
border:5px solid gray;  
margin:0px;  
}  
</style>  
</head>

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_boxmodel_width_doctype)

**Border Style**

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

None of the border properties will have ANY effect unless the **border-style** property is set!

**border-style values:**

none: Defines no border

dotted: Defines a dotted border

dashed: Defines a dashed border

solid: Defines a solid border

double: Defines two borders. The width of the two borders are the same as the border-width value

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

Try it yourself: [Set the style of the border](http://w3schools.com/css/tryit.asp?filename=trycss_border-style)

**Border Width**

The border-width property is used to set the width of the border.

The width is set in pixels, or by using one of the three pre-defined values: thin, medium, or thick.

**Note:** The "border-width" property does not work if it is used alone. Use the "border-style" property to set the borders first.

**Example**

p.one  
{  
border-style:solid;  
border-width:5px;  
}  
p.two  
{  
border-style:solid;  
border-width:medium;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_border-width)

**Border Color**

The border-color property is used to set the color of the border. The color can be set by:

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

You can also set the border color to "transparent".

**Note:** The "border-color" property does not work if it is used alone. Use the "border-style" property to set the borders first.

**Example**

p.one  
{  
border-style:solid;  
border-color:red;  
}  
p.two  
{  
border-style:solid;  
border-color:#98bf21;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_border-color1)

**Border - Individual sides**

In CSS it is possible to specify different borders for different sides:

**Example**

p  
{  
border-top-style:dotted;  
border-right-style:solid;  
border-bottom-style:dotted;  
border-left-style:solid;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_border-side)

The example above can also be set with a single property:

**Example**

border-style:dotted solid;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_border-side2)

The border-style property can have from one to four values.

* **border-style:dotted solid double dashed;** 
  + top border is dotted
  + right border is solid
  + bottom border is double
  + left border is dashed
* **border-style:dotted solid double;**
  + top border is dotted
  + right and left borders are solid
  + bottom border is double
* **border-style:dotted solid;**
  + top and bottom borders are dotted
  + right and left borders are solid
* **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. This is called a shorthand property.

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

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

**Example**

border:5px solid red;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_border)

**More Examples**

[All the top border properties in one declaration](http://w3schools.com/css/tryit.asp?filename=trycss_border-top)  
This example demonstrates a shorthand property for setting all of the properties for the top border in one declaration.

[Set the style of the bottom border](http://w3schools.com/css/tryit.asp?filename=trycss_border-bottom-style)  
This example demonstrates how to set the style of the bottom border.

[Set the width of the left border](http://w3schools.com/css/tryit.asp?filename=trycss_border-left-width)  
This example demonstrates how to set the width of the left border.

[Set the color of the four borders](http://w3schools.com/css/tryit.asp?filename=trycss_border-color)  
This example demonstrates how to set the color of the four borders. It can have from one to four colors.

[Set the color of the right border](http://w3schools.com/css/tryit.asp?filename=trycss_border-right-color)  
This example demonstrates how to set the color of the right border.

**All CSS Border Properties**

|  |  |
| --- | --- |
| **Property** | **Description** |
| [border](http://w3schools.com/cssref/pr_border.asp) | Sets all the border properties in one declaration |
| [border-bottom](http://w3schools.com/cssref/pr_border-bottom.asp) | Sets all the bottom border properties in one declaration |
| [border-bottom-color](http://w3schools.com/cssref/pr_border-bottom_color.asp) | Sets the color of the bottom border |
| [border-bottom-style](http://w3schools.com/cssref/pr_border-bottom_style.asp) | Sets the style of the bottom border |
| [border-bottom-width](http://w3schools.com/cssref/pr_border-bottom_width.asp) | Sets the width of the bottom border |
| [border-color](http://w3schools.com/cssref/pr_border-color.asp) | Sets the color of the four borders |
| [border-left](http://w3schools.com/cssref/pr_border-left.asp) | Sets all the left border properties in one declaration |
| [border-left-color](http://w3schools.com/cssref/pr_border-left_color.asp) | Sets the color of the left border |
| [border-left-style](http://w3schools.com/cssref/pr_border-left_style.asp) | Sets the style of the left border |
| [border-left-width](http://w3schools.com/cssref/pr_border-left_width.asp) | Sets the width of the left border |
| [border-right](http://w3schools.com/cssref/pr_border-right.asp) | Sets all the right border properties in one declaration |
| [border-right-color](http://w3schools.com/cssref/pr_border-right_color.asp) | Sets the color of the right border |
| [border-right-style](http://w3schools.com/cssref/pr_border-right_style.asp) | Sets the style of the right border |
| [border-right-width](http://w3schools.com/cssref/pr_border-right_width.asp) | Sets the width of the right border |
| [border-style](http://w3schools.com/cssref/pr_border-style.asp) | Sets the style of the four borders |
| [border-top](http://w3schools.com/cssref/pr_border-top.asp) | Sets all the top border properties in one declaration |
| [border-top-color](http://w3schools.com/cssref/pr_border-top_color.asp) | Sets the color of the top border |
| [border-top-style](http://w3schools.com/cssref/pr_border-top_style.asp) | Sets the style of the top border |
| [border-top-width](http://w3schools.com/cssref/pr_border-top_width.asp) | Sets the width of the top border |
| [border-width](http://w3schools.com/cssref/pr_border-width.asp) | Sets the width of the four borders |

**CSS Outline**

An outline is a line that is drawn around elements (outside the borders) to make the element "stand out".

However, the outline property is different from the border property.

The outline is not a part of an element's dimensions; the element's total width and height is not affected by the width of the outline.

**All CSS Outline Properties**

The number in the "CSS" column indicates in which CSS version the property is defined (CSS1 or CSS2).

|  |  |  |  |
| --- | --- | --- | --- |
| **Property** | **Description** | **Values** | **CSS** |
| [outline](http://w3schools.com/cssref/pr_outline.asp) | Sets all the outline properties in one declaration | *outline-color outline-style outline-width* inherit | 2 |
| [outline-color](http://w3schools.com/cssref/pr_outline-color.asp) | Sets the color of an outline | *color\_name hex\_number rgb\_number* invert inherit | 2 |
| [outline-style](http://w3schools.com/cssref/pr_outline-style.asp) | Sets the style of an outline | none dotted dashed solid double groove ridge inset outset inherit | 2 |
| [outline-width](http://w3schools.com/cssref/pr_outline-width.asp) | Sets the width of an outline | thin medium thick *length* inherit |  |

**Margin**

The margin clears an area around an element (outside the border). The margin does not have a background color, and is completely transparent.

The top, right, bottom, and left margin can be changed independently using separate properties. A shorthand margin property can also be used, to change all margins at once.

**Possible Values**

|  |  |
| --- | --- |
| **Value** | **Description** |
| auto | The browser calculates a margin |
| *length* | Specifies a margin in px, pt, cm, etc. Default value is 0px |
| *%* | Specifies a margin in percent of the width of the containing element |
| inherit | Specifies that the margin should be inherited from the parent element |

It is possible to use negative values, to overlap content.

**Margin - Individual sides**

In CSS, it is possible to specify different margins for different sides:

**Example**

margin-top:100px;  
margin-bottom:100px;  
margin-right:50px;  
margin-left:50px;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_margin_sides)

**Margin - Shorthand property**

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

The shorthand property for all the margin properties is "margin":

**Example**

margin:100px 50px;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_margin_shorthand)

The margin property can have from one to four values.

* **margin:25px 50px 75px 100px;** 
  + top margin is 25px
  + right margin is 50px
  + bottom margin is 75px
  + left margin is 100px
* **margin:25px 50px 75px;**
  + top margin is 25px
  + right and left margins are 50px
  + bottom margin is 75px
* **margin:25px 50px;**
  + top and bottom margins are 25px
  + right and left margins are 50px
* **margin:25px;**
  + all four margins are 25px

**More Examples**

[Set the top margin of a text using a cm value](http://w3schools.com/css/tryit.asp?filename=trycss_margin-top)  
This example demonstrates how to set the top margin of a text using a cm value.

[Set the bottom margin of a text using a percent value](http://w3schools.com/css/tryit.asp?filename=trycss_margin-bottom_percent)  
This example demonstrates how to set the bottom margin in percent, relative to the width of the containing element.

**All CSS Margin Properties**

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

**Padding**

The padding clears an area around the content (inside the border) of an element. The padding is affected by the background color of the element.

The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property can also be used, to change all paddings at once.

**Possible Values**

|  |  |
| --- | --- |
| **Value** | **Description** |
| *length* | Defines a fixed padding (in pixels, pt, em, etc.) |
| *%* | Defines a padding in % of the containing element |

**Padding - Individual sides**

In CSS, it is possible to specify different padding for different sides:

**Example**

padding-top:25px;  
padding-bottom:25px;  
padding-right:50px;  
padding-left:50px;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_padding_sides)

**Padding - Shorthand property**

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

The shorthand property for all the padding properties is "padding":

**Example**

padding:25px 50px;

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_padding_shorthand)

The padding property can have from one to four values.

* **padding:25px 50px 75px 100px;** 
  + top padding is 25px
  + right padding is 50px
  + bottom padding is 75px
  + left padding is 100px
* **padding:25px 50px 75px;**
  + top padding is 25px
  + right and left paddings are 50px
  + bottom padding is 75px
* **padding:25px 50px;**
  + top and bottom paddings are 25px
  + right and left paddings are 50px
* **padding:25px;**
  + all four paddings are 25px

**More Examples**

[All the padding properties in one declaration](http://w3schools.com/css/tryit.asp?filename=trycss_padding)  
This example demonstrates a shorthand property for setting all of the padding properties in one declaration, can have from one to four values.

[Set the left padding](http://w3schools.com/css/tryit.asp?filename=trycss_padding-left)  
This example demonstrates how to set the left padding of a p element.

[Set the right padding](http://w3schools.com/css/tryit.asp?filename=trycss_padding-right)  
This example demonstrates how to set the right padding of a p element.

[Set the top padding](http://w3schools.com/css/tryit.asp?filename=trycss_padding-top)  
This example demonstrates how to set the top padding of a p element.

[Set the bottom padding](http://w3schools.com/css/tryit.asp?filename=trycss_padding-bottom)  
This example demonstrates how to set the bottom padding of a p element.

**All CSS Padding Properties**

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

**Grouping Selectors**

In style sheets there are often elements with the same style.

h1  
{  
color:green;  
}  
h2  
{  
color:green;  
}  
p  
{  
color:green;  
}

To minimize the code, you can group selectors.

Separate each selector with a comma.

In the example below we have grouped the selectors from the code above:

**Example**

h1,h2,p  
{  
color:green;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_grouping)

**Nesting Selectors**

It is possible to apply a style for a selector within a selector.

In the example below, one style is specified for all p elements, one style is specified for all elements with class="marked", and a third style is specified only for p elements within elements with class="marked":

**Example**

p  
{  
color:blue;  
text-align:center;  
}  
.marked  
{  
background-color:red;  
}  
.marked p  
{  
color:white;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_nesting)

**Hiding an Element - display:none or visibility:hidden**

Hiding an element can be done by setting the display property to "none" or the visibility property to "hidden". However, notice that these two methods produce different results:

visibility:hidden hides an element, but it will still take up the same space as before. The element will be hidden, but still affect the layout.

**Example**

h1.hidden {visibility:hidden;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_visibility_hidden)

display:none hides an element, and it will not take up any space. The element will be hidden, and the page will be displayed as if the element is not there:

**Example**

h1.hidden {display:none;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_display_none)

**CSS Display - Block and Inline Elements**

A block element is an element that takes up the full width available, and has a line break before and after it.

Examples of block elements:

* <h1>
* <p>
* <div>

An inline element only takes up as much width as necessary, and does not force line breaks.

Examples of inline elements:

* <span>
* <a>

**Changing How an Element is Displayed**

Changing an inline element to a block element, or vice versa, can be useful for making the page look a specific way, and still follow web standards.

The following example displays list items as inline elements:

**Example**

li {display:inline;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_display_inline_list)

The following example displays span elements as block elements:

**Example**

span {display:block;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_display_block2)

**Note:** Changing the display type of an element changes only how the element is displayed, NOT what kind of element it is. For example: An inline element set to display:block is not allowed to have a block element nested inside of it.

**More Examples**

[How to display an element as an inline element.](http://w3schools.com/css/tryit.asp?filename=trycss_display)  
This example demonstrates how to display an element as an inline element.

[How to display an element as a block element](http://w3schools.com/css/tryit.asp?filename=trycss_display_block)  
This example demonstrates how to display an element as a block element.

[How to make a table element collapse](http://w3schools.com/css/tryit.asp?filename=trycss_visibility_collapse)  
This example demonstrates how to make a table element collapse.

**Positioning**

The CSS positioning properties allow you to position an element. It can also place an element behind another, and specify what should happen when an element's content is too big.

Elements can be positioned using the top, bottom, left, and right properties. However, these properties will not work unless the position property is set first. They also work differently depending on the positioning method.

There are four different positioning methods.

**Static Positioning**

HTML elements are positioned static by default. A static positioned element is always positioned according to the normal flow of the page.

Static positioned elements are not affected by the top, bottom, left, and right properties.

**Fixed Positioning**

An element with fixed position is positioned relative to the browser window.

It will not move even if the window is scrolled:

**Example**

p.pos\_fixed  
{  
position:fixed;  
top:30px;  
right:5px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_position_fixed)

**Note:** IE7 and IE8 support the fixed value only if a !DOCTYPE is specified.

Fixed positioned elements are removed from the normal flow. The document and other elements behave like the fixed positioned element does not exist.

Fixed positioned elements can overlap other elements.

**Relative Positioning**

A relative positioned element is positioned relative to its normal position.

**Example**

h2.pos\_left  
{  
position:relative;  
left:-20px;  
}  
h2.pos\_right  
{  
position:relative;  
left:20px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_position_relative)

The content of relatively positioned elements can be moved and overlap other elements, but the reserved space for the element is still preserved in the normal flow.

**Example**

h2.pos\_top  
{  
position:relative;  
top:-50px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_position_relative2)

Relatively positioned elements are often used as container blocks for absolutely positioned elements.

**Absolute Positioning**

An absolute position element is positioned relative to the first parent element that has a position other than static. If no such element is found, the containing block is <html>:

**Example**

h2  
{  
position:absolute;  
left:100px;  
top:150px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_position_absolute)

Absolutely positioned elements are removed from the normal flow. The document and other elements behave like the absolutely positioned element does not exist.

Absolutely positioned elements can overlap other elements.

**Overlapping Elements**

When elements are positioned outside the normal flow, they can overlap other elements.

The z-index property specifies the stack order of an element (which element should be placed in front of, or behind, the others).

An element can have a positive or negative stack order:

**Example**

img  
{  
position:absolute;  
left:0px;  
top:0px;  
z-index:-1;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_zindex)

An element with greater stack order is always in front of an element with a lower stack order.

**Note:** If two positioned elements overlap, without a z-index specified, the element positioned last in the HTML code will be shown on top.

**More Examples**

[Set the shape of an element](http://w3schools.com/css/tryit.asp?filename=trycss_clip)  
This example demonstrates how to set the shape of an element. The element is clipped into this shape, and displayed.

[How to show overflow in an element using scroll](http://w3schools.com/css/tryit.asp?filename=trycss_overflow)  
This example demonstrates how to set the overflow property to create a scroll bar when an element's content is too big to fit in a specified area.

[How to set the browser to automatically handle overflow](http://w3schools.com/css/tryit.asp?filename=trycss_pos_overflow_auto)  
This example demonstrates how to set the browser to automatically handle overflow.

[Change the cursor](http://w3schools.com/css/tryit.asp?filename=trycss_cursor)  
This example demonstrates how to change the cursor.

**All CSS Positioning Properties**

The number in the "CSS" column indicates in which CSS version the property is defined (CSS1 or CSS2).

|  |  |  |  |
| --- | --- | --- | --- |
| **Property** | **Description** | **Values** | **CSS** |
| [bottom](http://w3schools.com/cssref/pr_pos_bottom.asp) | Sets the bottom margin edge for a positioned box | auto *length %* inherit | 2 |
| [clip](http://w3schools.com/cssref/pr_pos_clip.asp) | Clips an absolutely positioned element | *shape* auto inherit | 2 |
| [cursor](http://w3schools.com/cssref/pr_class_cursor.asp) | Specifies the type of cursor to be displayed | *url* auto crosshair default pointer move e-resize ne-resize nw-resize n-resize se-resize sw-resize s-resize w-resize text wait help | 2 |
| [left](http://w3schools.com/cssref/pr_pos_left.asp) | Sets the left margin edge for a positioned box | auto *length %* inherit | 2 |
| [overflow](http://w3schools.com/cssref/pr_pos_overflow.asp) | Specifies what happens if content overflows an element's box | auto hidden scroll visible inherit | 2 |
| [position](http://w3schools.com/cssref/pr_class_position.asp) | Specifies the type of positioning for an element | absolute fixed relative static inherit | 2 |
| [right](http://w3schools.com/cssref/pr_pos_right.asp) | Sets the right margin edge for a positioned box | auto *length %* inherit | 2 |
| [top](http://w3schools.com/cssref/pr_pos_top.asp) | Sets the top margin edge for a positioned box | auto *length %* inherit | 2 |
| [z-index](http://w3schools.com/cssref/pr_pos_z-index.asp) | Sets the stack order of an element | *number* auto inherit | 2 |

**How Elements Float**

Elements are floated horizontally, this means that an element can only be floated left or right, not up or down.

A floated element will move as far to the left or right as it can. Usually this means all the way to the left or right of the containing element.

The elements after the floating element will flow around it.

The elements before the floating element will not be affected.

If an image is floated to the right, a following text flows around it, to the left:

**Example**

img  
{  
float:right;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_float)

**Floating Elements Next to Each Other**

If you place several floating elements after each other, they will float next to each other if there is room.

Here we have made an image gallery using the float property:

**Example**

.thumbnail   
{  
float:left;  
width:110px;  
height:90px;  
margin:5px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_float_elements)

**Turning off Float - Using Clear**

Elements after the floating element will flow around it. To avoid this, use the clear property.

The clear property specifies which sides of an element other floating elements are not allowed.

Add a text line into the image gallery, using the clear property:

**Example**

.text\_line  
{  
clear:both;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_float_clear)

**More Examples**

[An image with border and margins that floats to the right in a paragraph](http://w3schools.com/css/tryit.asp?filename=trycss_float2)  
Let an image float to the right in a paragraph. Add border and margins to the image.

[An image with a caption that floats to the right](http://w3schools.com/css/tryit.asp?filename=trycss_float3)  
Let an image with a caption float to the right.

[Let the first letter of a paragraph float to the left](http://w3schools.com/css/tryit.asp?filename=trycss_float4)  
Let the first letter of a paragraph float to the left and style the letter.

[Creating a horizontal menu](http://w3schools.com/css/tryit.asp?filename=trycss_float5)  
Use float with a list of hyperlinks to create a horizontal menu.

[Creating a homepage without tables](http://w3schools.com/css/tryit.asp?filename=trycss_float6)  
Use float to create a homepage with a header, footer, left content and main content.

**All CSS Float Properties**

The number in the "CSS" column indicates in which CSS version the property is defined (CSS1 or CSS2).

|  |  |  |  |
| --- | --- | --- | --- |
| **Property** | **Description** | **Values** | **CSS** |
| [clear](http://w3schools.com/cssref/pr_class_clear.asp) | Specifies which sides of an element where other floating elements are not allowed | left right both none inherit | 1 |
| [float](http://w3schools.com/cssref/pr_class_float.asp) | Specifies whether or not a box should float | left right none inherit | 1 |

**Aligning Block Elements**

A block element is an element that takes up the full width available, and has a line break before and after it.

Examples of block elements:

* <h1>
* <p>
* <div>

For aligning text, see the [CSS Text](http://w3schools.com/css/css_text.asp) chapter.

In this chapter we will show you how to horizontally align block elements for layout purposes.

**Center Aligning Using the margin Property**

Block elements can be aligned by setting the left and right margins to "auto".

**Note:** Using margin:auto will not work in IE8 and earlier, **unless a !DOCTYPE is declared.**

Setting the left and right margins to auto specifies that they should split the available margin equally. The result is a centered element:

**Example**

.center  
{  
margin-left:auto;  
margin-right:auto;  
width:70%;  
background-color:#b0e0e6;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_align_container)

**Tip:** Aligning has no effect if the width is 100%.

**Note:** In IE5 there is a margin handling bug for block elements. To make the example above work in IE5, add some extra code. [Try it yourself](http://w3schools.com/css/tryit.asp?filename=trycss_align_container_ie5)

**Left and Right Aligning Using the position Property**

One method of aligning elements is to use absolute positioning:

**Example**

.right  
{  
position:absolute;  
right:0px;  
width:300px;  
background-color:#b0e0e6;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_align_pos)

**Note:** Absolute positioned elements are removed from the normal flow, and can overlap elements.

**Crossbrowser Compatibility Issues**

When aligning elements like this, it is always a good idea to predefine margin and padding for the <body> element. This is to avoid visual differences in different browsers.

There is a problem with IE8 and earlier, when using the position property. If a container element (in our case <div class="container">) has a specified width, and the !DOCTYPE declaration is missing, IE8 and earlier versions will add a 17px margin on the right side. This seems to be space reserved for a scrollbar. Always set the !DOCTYPE declaration when using the position property:

**Example**

body  
{  
margin:0;  
padding:0;  
}  
.container  
{  
position:relative;  
width:100%;  
}  
.right  
{  
position:absolute;  
right:0px;  
width:300px;  
background-color:#b0e0e6;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_align_pos_crossbrowser)

**Left and Right Aligning Using the float Property**

One method of aligning elements is to use the float property:

**Example**

.right  
{  
float:right;  
width:300px;  
background-color:#b0e0e6;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_align_float)

**Crossbrowser Compatibility Issues**

When aligning elements like this, it is always a good idea to predefine margin and padding for the <body> element. This is to avoid visual differences in different browsers.

There is a problem with IE8 and earlier when using the float property. If the !DOCTYPE declaration is missing, IE8 and earlier versions will add a 17px margin on the right side. This seems to be space reserved for a scrollbar. Always set the !DOCTYPE declaration when using the float property:

**Example**

body  
{  
margin:0;  
padding:0;  
}  
.right  
{  
float:right;  
width:300px;  
background-color:#b0e0e6;  
}

**Syntax**

The syntax of pseudo-classes:

selector:pseudo-class {property:value;}

CSS classes can also be used with pseudo-classes:

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

**Anchor Pseudo-classes**

Links can be displayed in different ways in a CSS-supporting browser:

**Example**

a:link {color:#FF0000;}      /\* unvisited link \*/  
a:visited {color:#00FF00;}  /\* visited link \*/  
a:hover {color:#FF00FF;}  /\* mouse over link \*/  
a:active {color:#0000FF;}  /\* selected link \*/

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_link)

**Note:** a:hover MUST come after a:link and a:visited in the CSS definition in order to be effective!!

**Note:** a:active MUST come after a:hover in the CSS definition in order to be effective!!

**Note:** Pseudo-class names are not case-sensitive.

**Pseudo-classes and CSS Classes**

Pseudo-classes can be combined with CSS classes:

a.red:visited {color:#FF0000;}  
  
<a class="red" href="css\_syntax.asp">CSS Syntax</a>

If the link in the example above has been visited, it will be displayed in red.

**CSS - The :first-child Pseudo-class**

The :first-child pseudo-class matches a specified element that is the first child of another element.

**Note:** For :first-child to work in IE8 and earlier, a [<!DOCTYPE>](http://w3schools.com/tags/tag_doctype.asp) must be declared.

**Match the first <p> element**

In the following example, the selector matches any <p> element that is the first child of any element:

**Example**

<html>  
<head>  
<style type="text/css">  
p:first-child  
{  
color:blue;  
}   
</style>  
</head>  
  
<body>  
<p>I am a strong man.</p>  
<p>I am a strong man.</p>  
</body>  
</html>

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_first-child1)

**Match the first <i> element in all <p> elements**

In the following example, the selector matches the first <i> element in all <p> elements:

**Example**

<html>  
<head>  
<style type="text/css">  
p > i:first-child  
{  
color:blue;  
}   
</style>  
</head>  
  
<body>  
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>  
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>  
</body>  
</html>

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_first-child2)

**Match all <i> elements in all first child <p> elements**

In the following example, the selector matches all <i> elements in <p> elements that are the first child of another element:

**Example**

<html>  
<head>  
<style type="text/css">  
p:first-child i  
{  
color:blue;  
}   
</style>  
</head>  
  
<body>  
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>  
<p>I am a <i>strong</i> man. I am a <i>strong</i> man.</p>  
</body>  
</html>

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_first-child3)

**CSS - The :lang Pseudo-class**

The :lang pseudo-class allows you to define special rules for different languages.

**Note:** IE8 supports the :lang pseudo-class only if a [<!DOCTYPE>](http://w3schools.com/tags/tag_doctype.asp) is specified.

In the example below, the :lang class defines the quotation marks for q elements with lang="no":

**Example**

<html>  
<head>  
<style type="text/css">  
q:lang(no) {quotes: "~" "~";}  
</style>  
</head>  
  
<body>  
<p>Some text <q lang="no">A quote in a paragraph</q> Some text.</p>  
</body>  
</html>

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_lang)

**More Examples**

[Add different styles to hyperlinks](http://w3schools.com/css/tryit.asp?filename=trycss_link2)  
This example demonstrates how to add other styles to hyperlinks.

[Use of :focus](http://w3schools.com/css/tryit.asp?filename=trycss_link_focus)  
This example demonstrates how to use the :focus pseudo-class.

**All CSS Pseudo Classes/Elements**

|  |  |  |
| --- | --- | --- |
| **Selector** | **Example** | **Example description** |
| [:link](http://w3schools.com/cssref/sel_link.asp) | a:link | Selects all unvisited links |
| [:visited](http://w3schools.com/cssref/sel_visited.asp) | a:visited | Selects all visited links |
| [:active](http://w3schools.com/cssref/sel_active.asp) | a:active | Selects the active link |
| [:hover](http://w3schools.com/cssref/sel_hover.asp) | a:hover | Selects links on mouse over |
| [:focus](http://w3schools.com/cssref/sel_focus.asp) | input:focus | Selects the input element which has focus |
| [:first-letter](http://w3schools.com/cssref/sel_firstletter.asp) | p:first-letter | Selects the first letter of every <p> element |
| [:first-line](http://w3schools.com/cssref/sel_firstline.asp) | p:first-line | Selects the first line of every <p> element |
| [:first-child](http://w3schools.com/cssref/sel_firstchild.asp) | p:first-child | Selects every <p> elements that is the first child of its parent |
| [:before](http://w3schools.com/cssref/sel_before.asp) | p:before | Insert content before every <p> element |
| [:after](http://w3schools.com/cssref/sel_after.asp) | p:after | Insert content after every <p> element |
| [:lang(*language*)](http://w3schools.com/cssref/sel_lang.asp) | p:lang(it) | Selects every <p> element with a lang attribute value starting with "it" |

**Syntax**

The syntax of pseudo-elements:

selector:pseudo-element {property:value;}

CSS classes can also be used with pseudo-elements:

selector.class:pseudo-element {property:value;}

**The :first-line Pseudo-element**

The "first-line" pseudo-element is used to add a special style to the first line of a text.

In the following example the browser formats the first line of text in a p element according to the style in the "first-line" pseudo-element (where the browser breaks the line, depends on the size of the browser window):

**Example**

p:first-line   
{  
color:#ff0000;  
font-variant:small-caps;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_firstline)

**Note:** The "first-line" pseudo-element can only be used with block-level elements.

**Note:** The following properties apply to the "first-line" pseudo-element:

* font properties
* color properties
* background properties
* word-spacing
* letter-spacing
* text-decoration
* vertical-align
* text-transform
* line-height
* clear

**The :first-letter Pseudo-element**

The "first-letter" pseudo-element is used to add a special style to the first letter of a text:

**Example**

p:first-letter   
{  
color:#ff0000;  
font-size:xx-large;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_firstletter)

**Note:** The "first-letter" pseudo-element can only be used with block-level elements.

**Note:** The following properties apply to the "first-letter" pseudo- element:

* font properties
* color properties
* background properties
* margin properties
* padding properties
* border properties
* text-decoration
* vertical-align (only if "float" is "none")
* text-transform
* line-height
* float
* clear

**Pseudo-elements and CSS Classes**

Pseudo-elements can be combined with CSS classes:

p.article:first-letter {color:#ff0000;}  
  
<p class="article">A paragraph in an article</p>

The example above will display the first letter of all paragraphs with class="article", in red.

**Multiple Pseudo-elements**

Several pseudo-elements can also be combined.

In the following example, the first letter of a paragraph will be red, in an xx-large font size. The rest of the first line will be blue, and in small-caps. The rest of the paragraph will be the default font size and color:

**Example**

p:first-letter  
{  
color:#ff0000;  
font-size:xx-large;  
}  
p:first-line   
{  
color:#0000ff;  
font-variant:small-caps;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_firstline_letter)

**CSS - The :before Pseudo-element**

The ":before" pseudo-element can be used to insert some content before the content of an element.

The following example inserts an image before each <h1> element:

**Example**

h1:before   
{  
content:url(smiley.gif);  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_before)

**CSS - The :after Pseudo-element**

The ":after" pseudo-element can be used to insert some content after the content of an element.

The following example inserts an image after each <h1> element:

**Example**

h1:after  
{  
content:url(smiley.gif);  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_after)

**All CSS Pseudo Classes/Elements**

|  |  |  |
| --- | --- | --- |
| **Selector** | **Example** | **Example description** |
| [:link](http://w3schools.com/cssref/sel_link.asp) | a:link | Selects all unvisited links |
| [:visited](http://w3schools.com/cssref/sel_visited.asp) | a:visited | Selects all visited links |
| [:active](http://w3schools.com/cssref/sel_active.asp) | a:active | Selects the active link |
| [:hover](http://w3schools.com/cssref/sel_hover.asp) | a:hover | Selects links on mouse over |
| [:focus](http://w3schools.com/cssref/sel_focus.asp) | input:focus | Selects the input element which has focus |
| [:first-letter](http://w3schools.com/cssref/sel_firstletter.asp) | p:first-letter | Selects the first letter of every <p> element |
| [:first-line](http://w3schools.com/cssref/sel_firstline.asp) | p:first-line | Selects the first line of every <p> element |
| [:first-child](http://w3schools.com/cssref/sel_firstchild.asp) | p:first-child | Selects every <p> elements that is the first child of its parent |
| [:before](http://w3schools.com/cssref/sel_before.asp) | p:before | Insert content before every <p> element |
| [:after](http://w3schools.com/cssref/sel_after.asp) | p:after | Insert content after every <p> element |
| [:lang(*language*)](http://w3schools.com/cssref/sel_lang.asp) | p:lang(it) | Selects every <p> element with a lang attribute value starting with "it" |

**Navigation Bars**

Having easy-to-use navigation is important for any web site.

With CSS you can transform boring HTML menus into good-looking navigation bars.

**Navigation Bar = List of Links**

A navigation bar needs standard HTML as a base.

In our examples we will build the navigation bar from a standard HTML list.

A navigation bar is basically a list of links, so using the <ul> and <li> elements makes perfect sense:

**Example**

<ul>  
<li><a href="default.asp">Home</a></li>  
<li><a href="news.asp">News</a></li>  
<li><a href="contact.asp">Contact</a></li>  
<li><a href="about.asp">About</a></li>  
</ul>

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_basic_html)

Now let's remove the bullets and the margins and padding from the list:

**Example**

ul  
{  
list-style-type:none;  
margin:0;  
padding:0;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_basic_css)

Example explained:

* list-style-type:none - Removes the bullets. A navigation bar does not need list markers
* Setting margins and padding to 0 to remove browser default settings

The code in the example above is the standard code used in both vertical, and horizontal navigation bars.

**Vertical Navigation Bar**

To build a vertical navigation bar we only need to style the <a> elements, in addition to the code above:

**Example**

a  
{  
display:block;  
width:60px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_vertical)

Example explained:

* display:block - Displaying the links as block elements makes the whole link area clickable (not just the text), and it allows us to specify the width
* width:60px - Block elements take up the full width available by default. We want to specify a 60 px width

**Tip:** Also take a look at our [fully styled vertical navigation bar example](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_vertical_advanced).

**Note:** Always specify the width for <a> elements in a vertical navigation bar. If you omit the width, IE6 can produce unexpected results.

**Horizontal Navigation Bar**

There are two ways to create a horizontal navigation bar. Using **inline** or **floating** list items.

Both methods work fine, but if you want the links to be the same size, you have to use the floating method.

**Inline List Items**

One way to build a horizontal navigation bar is to specify the <li> elements as inline, in addition to the "standard" code above:

**Example**

li  
{  
display:inline;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_horizontal)

Example explained:

* display:inline; - By default, <li> elements are block elements. Here, we remove the line breaks before and after each list item, to display them on one line

**Tip:** Also take a look at our [fully styled horizontal navigation bar example](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_horizontal_advanced).

**Floating List Items**

In the example above the links have different widths.

For all the links to have an equal width, float the <li> elements and specify a width for the <a> elements:

**Example**

li  
{  
float:left;  
}  
a  
{  
display:block;  
width:60px;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_horizontal_float)

Example explained:

* float:left - use float to get block elements to slide next to each other
* display:block - Displaying the links as block elements makes the whole link area clickable (not just the text), and it allows us to specify the width
* width:60px - Since block elements take up the full width available, they cannot float next to each other. We specify the width of the links to 60px

**Tip:** Also take a look at our [fully styled horizontal navigation bar example](http://w3schools.com/css/tryit.asp?filename=trycss_navbar_horizontal_float_advanced).

**Image Gallery**

The following image gallery is created with CSS:

**Example**

<html>  
<head>  
<style type="text/css">  
div.img  
  {  
  margin:2px;  
  border:1px solid #0000ff;  
  height:auto;  
  width:auto;  
  float:left;  
  text-align:center;  
  }  
div.img img  
  {  
  display:inline;  
  margin:3px;  
  border:1px solid #ffffff;  
  }  
div.img a:hover img  
  {  
  border:1px solid #0000ff;  
  }  
div.desc  
  {  
  text-align:center;  
  font-weight:normal;  
  width:120px;  
  margin:2px;  
  }  
</style>  
</head>  
<body>  
  
<div class="img">  
  <a target="\_blank" href="klematis\_big.htm">  
  <img src="klematis\_small.jpg" alt="Klematis" width="110" height="90" />  
  </a>  
  <div class="desc">Add a description of the image here</div>  
</div>  
<div class="img">  
  <a target="\_blank" href="klematis2\_big.htm">  
  <img src="klematis2\_small.jpg" alt="Klematis" width="110" height="90" />  
  </a>  
  <div class="desc">Add a description of the image here</div>  
</div>  
<div class="img">  
  <a target="\_blank" href="klematis3\_big.htm">  
  <img src="klematis3\_small.jpg" alt="Klematis" width="110" height="90" />  
  </a>  
  <div class="desc">Add a description of the image here</div>  
</div>  
<div class="img">  
  <a target="\_blank" href="klematis4\_big.htm">  
  <img src="klematis4\_small.jpg" alt="Klematis" width="110" height="90" />  
  </a>  
  <div class="desc">Add a description of the image here</div>  
</div>  
  
</body>  
</html>

**Example 1 - Creating a Transparent Image**

The CSS3 property for transparency is **opacity**.

First we will show you how to create a transparent image with CSS.

Regular image:

The same image with transparency:

Look at the following CSS:

img  
{  
opacity:0.4;  
filter:alpha(opacity=40); /\* For IE8 and earlier \*/  
}

IE9, Firefox, Chrome, Opera, and Safari use the property **opacity** for transparency. The opacity property can take a value from 0.0 - 1.0. A lower value makes the element more transparent.

IE8 and earlier use **filter:alpha(opacity=x)**. The x can take a value from 0 - 100. A lower value makes the element more transparent.

**Example 2 - Image Transparency - Hover Effect**

Mouse over the images:

The CSS looks like this:

img  
{  
opacity:0.4;  
filter:alpha(opacity=40); /\* For IE8 and earlier \*/  
}  
img:hover  
{  
opacity:1.0;  
filter:alpha(opacity=100); /\* For IE8 and earlier \*/  
}

The first CSS block is similar to the code in Example 1. In addition, we have added what should happen when a user hover over one of the images. In this case we want the image to NOT be transparent when the user hover over it.

The CSS for this is: **opacity=1**.

IE8 and earlier: **filter:alpha(opacity=100)**.

When the mouse pointer moves away from the image, the image will be transparent again.

**Example 3 - Text in Transparent Box**

This is some text that is placed in the transparent box. This is some text that is placed in the transparent box. This is some text that is placed in the transparent box. This is some text that is placed in the transparent box. This is some text that is placed in the transparent box.

The source code looks like this:

<html>  
<head>  
<style type="text/css">  
div.background  
  {  
  width:500px;  
  height:250px;  
  background:url(klematis.jpg) repeat;  
  border:2px solid black;  
  }  
div.transbox  
  {  
  width:400px;  
  height:180px;  
  margin:30px 50px;  
  background-color:#ffffff;  
  border:1px solid black;  
  opacity:0.6;  
  filter:alpha(opacity=60); /\* For IE8 and earlier \*/  
  }  
div.transbox p  
  {  
  margin:30px 40px;  
  font-weight:bold;  
  color:#000000;  
  }  
</style>  
</head>  
  
<body>  
  
<div class="background">  
<div class="transbox">  
<p>This is some text that is placed in the transparent box.  
This is some text that is placed in the transparent box.  
This is some text that is placed in the transparent box.  
This is some text that is placed in the transparent box.  
This is some text that is placed in the transparent box.  
</p>  
</div>  
</div>  
  
</body>  
</html>

First, we create a div element (class="background") with a fixed height and width, a background image, and a border. Then we create a smaller div (class="transbox") inside the first div element. The "transbox" div have a fixed width, a background color, and a border - and it is transparent. Inside the transparent div, we add some text inside a p element.

**Image Sprites**

An image sprite is a collection of images put into a single image.

A web page with many images can take a long time to load and generates multiple server requests.

Using image sprites will reduce the number of server requests and save bandwidth.

**Image Sprites - Simple Example**

Instead of using three separate images, we use this single image ("img\_navsprites.gif"):

With CSS, we can show just the part of the image we need.

In the following example the CSS specifies which part of the "img\_navsprites.gif" image to show:

**Example**

img.home  
{  
width:46px;  
height:44px;  
background:url(img\_navsprites.gif) 0 0;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_sprites_img)

**Example explained:**

* <img class="home" src="img\_trans.gif" /> - Only defines a small transparent image because the src attribute cannot be empty. The displayed image will be the background image we specify in CSS
* width:46px;height:44px; - Defines the portion of the image we want to use
* background:url(img\_navsprites.gif) 0 0; - Defines the background image and its position (left 0px, top 0px)

This is the easiest way to use image sprites, now we want to expand it by using links and hover effects.

**Image Sprites - Create a Navigation List**

We want to use the sprite image ("img\_navsprites.gif") to create a navigation list.

We will use an HTML list, because it can be a link and also supports a background image:

**Example**

#navlist{position:relative;}  
#navlist li{margin:0;padding:0;list-style:none;position:absolute;top:0;}  
#navlist li, #navlist a{height:44px;display:block;}  
  
#home{left:0px;width:46px;}  
#home{background:url('img\_navsprites.gif') 0 0;}  
  
#prev{left:63px;width:43px;}  
#prev{background:url('img\_navsprites.gif') -47px 0;}  
  
#next{left:129px;width:43px;}  
#next{background:url('img\_navsprites.gif') -91px 0;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_sprites_nav)

**Example explained:**

* #navlist{position:relative;} - position is set to relative to allow absolute positioning inside it
* #navlist li{margin:0;padding:0;list-style:none;position:absolute;top:0;} - margin and padding is set to 0, list-style is removed, and all list items are absolute positioned
* #navlist li, #navlist a{height:44px;display:block;} - the height of all the images are 44px

Now start to position and style for each specific part:

* #home{left:0px;width:46px;} - Positioned all the way to the left, and the width of the image is 46px
* #home{background:url(img\_navsprites.gif) 0 0;} - Defines the background image and its position (left 0px, top 0px)
* #prev{left:63px;width:43px;} - Positioned 63px to the right (#home width 46px + some extra space between items), and the width is 43px.
* #prev{background:url('img\_navsprites.gif') -47px 0;} - Defines the background image 47px to the right (#home width 46px + 1px line divider)
* #next{left:129px;width:43px;}- Positioned 129px to the right (start of #prev is 63px + #prev width 43px + extra space), and the width is 43px.
* #next{background:url('img\_navsprites.gif') no-repeat -91px 0;} - Defines the background image 91px to the right (#home width 46px + 1px line divider + #prev width 43px + 1px line divider )

**Image Sprites - Hover Effect**

Now we want to add a hover effect to our navigation list.

Our new image ("img\_navsprites\_hover.gif") contains three navigation images and three images to use for hover effects:

Because this is one single image, and not six separate files, there will be **no loading delay** when a user hovers over the image.

We only add three lines of code to add the hover effect:

**Example**

#home a:hover{background: url('img\_navsprites\_hover.gif') 0 -45px;}  
#prev a:hover{background: url('img\_navsprites\_hover.gif') -47px -45px;}  
#next a:hover{background: url('img\_navsprites\_hover.gif') -91px -45px;}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_sprites_hover_nav)

Example explained:

* Since the list item contains a link, we can use the :hover pseudo-class
* #home a:hover{background: transparent url(img\_navsprites\_hover.gif) 0 -45px;} - For all three hover images we specify the same background position,  only 45px further down
* **Media Types**
* Some CSS properties are only designed for a certain media. For example the "voice-family" property is designed for aural user agents. Some other properties can be used for different media types. For example, the "font-size" property can be used for both screen and print media, but perhaps with different values. A document usually needs a larger font-size on a screen than on paper, and sans-serif fonts are easier to read on the screen, while serif fonts are easier to read on paper.
* **The @media Rule**
* The @media rule allows different style rules for different media in the same style sheet.
* The style in the example below tells the browser to display a 14 pixels Verdana font on the screen. But if the page is printed, it will be in a 10 pixels Times font. Notice that the font-weight is set to bold, both on screen and on paper:
* <html>  
  <head>  
  <style>  
  @media screen  
    {  
    p.test {font-family:verdana,sans-serif;font-size:14px;}  
    }  
  @media print  
    {  
    p.test {font-family:times,serif;font-size:10px;}  
    }  
  @media screen,print  
    {  
    p.test {font-weight:bold;}  
    }  
  </style>  
  </head>  
    
  <body>  
  ....  
  </body>  
  </html>
* **See it yourself !** If you are using Mozilla/Firefox or IE5+ and print this page, you will see that the paragraph under "Media Types" will be displayed in another font, and have a smaller font size than the rest of the text.
* **Different Media Types**
* **Note:** The media type names are not case-sensitive.

|  |  |
| --- | --- |
| **Media Type** | **Description** |
| all | Used for all media type devices |
| aural | Used for speech and sound synthesizers |
| braille | Used for braille tactile feedback devices |
| embossed | Used for paged braille printers |
| handheld | Used for small or handheld devices |
| print | Used for printers |
| projection | Used for projected presentations, like slides |
| screen | Used for computer screens |
| tty | Used for media using a fixed-pitch character grid, like teletypes and terminals |
| tv | Used for television-type devices |

**Style HTML Elements With Specific Attributes**

It is possible to style HTML elements that have specific attributes, not just class and id.

**Note:** IE7 and IE8 support attribute selectors only if a !DOCTYPE is specified. Attribute selection is **NOT** supported in IE6 and lower.

**Attribute Selector**

The example below styles all elements with a title attribute:

**Example**

[title]  
{  
color:blue;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_attselector_att)

**Attribute and Value Selector**

The example below styles all elements with title="W3Schools":

**Example**

[title=W3Schools]  
{  
border:5px solid green;  
}

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_attselector_value)

**Attribute and Value Selector - Multiple Values**

The example below styles all elements with a title attribute that contains a specified value. This works even if the attribute has space separated values:

**Example**

[title~=hello] { color:blue; }

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_attselector_valuelist_space)

The example below styles all elements with a lang attribute that contains a specified value. This works even if the attribute has hyphen ( - ) separated values:

**Example**

[lang|=en] { color:blue; }

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_attselector_valuelist_hyphen)

**Styling Forms**

The attribute selectors are particularly useful for styling forms without class or ID:

**Example**

input[type="text"]  
{  
width:150px;  
display:block;  
margin-bottom:10px;  
background-color:yellow;  
}  
input[type="button"]  
{  
width:120px;  
margin-left:35px;  
display:block;  
}

**CSS Summary**

This tutorial has taught you how to create style sheets to control the style and layout of multiple web sites at once.

You have learned how to use CSS to add backgrounds, format text, add and format borders, and specify padding and margins of elements.

You have also learned how to position an element, control the visibility and size of an element, set the shape of an element, place an element behind another, and to add special effects to some selectors, like links.

For more information on CSS, please take a look at our [CSS examples](http://w3schools.com/css/css_examples.asp) and our [CSS reference](http://w3schools.com/cssref/default.asp).

**Now You Know CSS, What's Next?**

The next step is to learn JavaScript.

**JavaScript**

JavaScript can make your web site more dynamic.

A static web site is nice when you just want to show flat content, but a dynamic web site can react to events and allow user interaction.

JavaScript is the most popular scripting language on the internet and it works with all major browsers.

If you want to learn more about JavaScript, please visit our [JavaScript tutorial](http://w3schools.com/js/default.asp).

[Try it yourself »](http://w3schools.com/css/tryit.asp?filename=trycss_attselector_form)